Quét để xem sơ đồ trực tuyến

COMPUTER SCIENCE

Nhập môn Công nghệ Thông tin: Từ con số 0 đến nền tảng số vững chắc

Lộ trình nhập môn CNTT dành cho người chưa có nền tảng: hiểu máy tính, hệ điều hành, Internet, thuật toán, Python, dữ liệu, phát triển phần mềm, an toàn thông tin và AI. Học qua ví dụ, thực hành có minh chứng, kiểm thử và dự án cuối; sau khóa có thể chọn nhánh chuyên sâu phù hợp.

Cơ bản · 10 giai đoạn · 50 chủ đề · Cập nhật 04/11/2026

Tài liệu nguồn chính: https://csed.acm.org/knowledge-areas/

MỤC TIÊU

- Phân biệt CNTT, Khoa học máy tính và các vị trí nghề; lựa chọn hướng học dựa trên sản phẩm và sở thích
- Giải thích cơ chế cơ bản của máy tính, hệ điều hành, Internet, Web, HTTP và lưu trữ dữ liệu
- Tự phân rã bài toán, viết thuật toán đơn giản, hiện thực bằng Python và kiểm thử đúng/sai
- Hiểu JSON, CSV, cơ sở dữ liệu quan hệ, SQL cơ bản và nguyên tắc chất lượng dữ liệu
- Sử dụng Git, terminal, trình duyệt DevTools và quy trình xây dựng phần mềm mức nhập môn
- Thực hành quyền riêng tư, bảo mật tài khoản, kiểm chứng dữ liệu và sử dụng AI có trách nhiệm
- Hoàn thành một sản phẩm nhỏ có README, dữ liệu mẫu, bộ ca thử, lịch sử Git và bản demo
- Chọn được lộ trình học tiếp theo (Web, Mobile, AI, Data, Security, Cloud...) dựa trên năng lực đã thể hiện

CÁC GIAI ĐOẠN

Giai đoạn 1: Hiểu ngành Công nghệ thông tin

Định hướng trước khi chọn công cụ: hiểu ngành, sản phẩm số, nghề nghiệp và nguyên tắc học có trách nhiệm. (5 chủ đề)

Giai đoạn 2: Máy tính và hệ điều hành

Làm chủ kiến thức kỹ thuật số tối thiểu: phần cứng, dữ liệu số, hệ điều hành, tệp và terminal. (5 chủ đề)

Giai đoạn 3: Internet, Web và an toàn trực tuyến

Tìm hiểu Internet và Web như một hệ thống; đồng thời rèn thói quen an toàn số. (5 chủ đề)

Giai đoạn 4: Tư duy thuật toán và lập trình

Luyện tư duy thuật toán độc lập với ngôn ngữ lập trình. (5 chủ đề)

Giai đoạn 5: Thực hành lập trình với Python

Chuyển thuật toán thành chương trình Python nhỏ có thể chạy, kiểm thử và sửa lỗi. (5 chủ đề)

Giai đoạn 6: Dữ liệu và cơ sở dữ liệu
Hiểu dữ liệu có cấu trúc, mô hình quan hệ và truy vấn để không chỉ biết code mà còn biết dữ liệu đi đâu. (5 chủ đề)

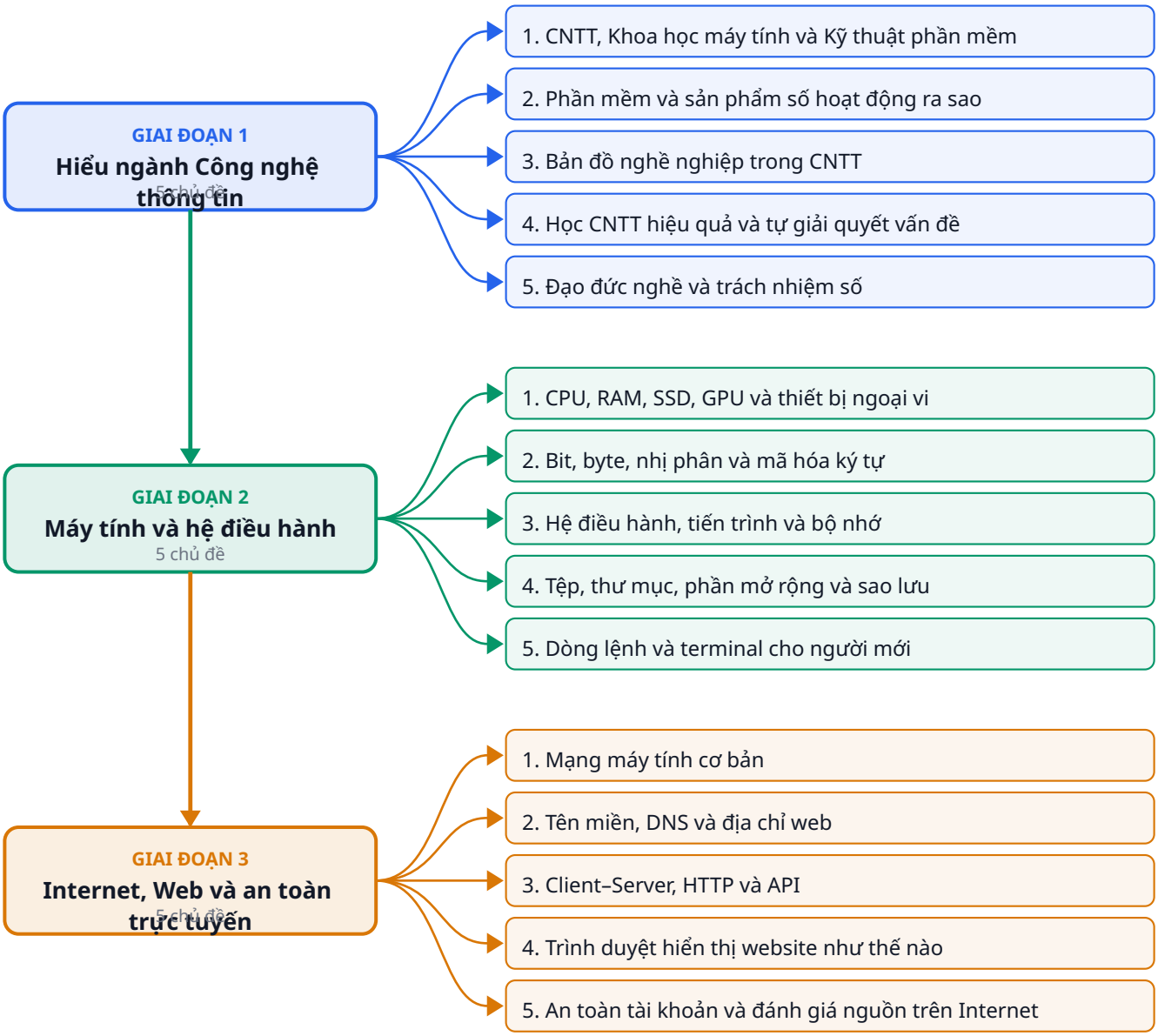
Giai đoạn 7: Xây dựng phần mềm và cộng tác
Tập trung vào quy trình phát triển sản phẩm: yêu cầu, thiết kế, Git, kiểm thử, triển khai. (5 chủ đề)

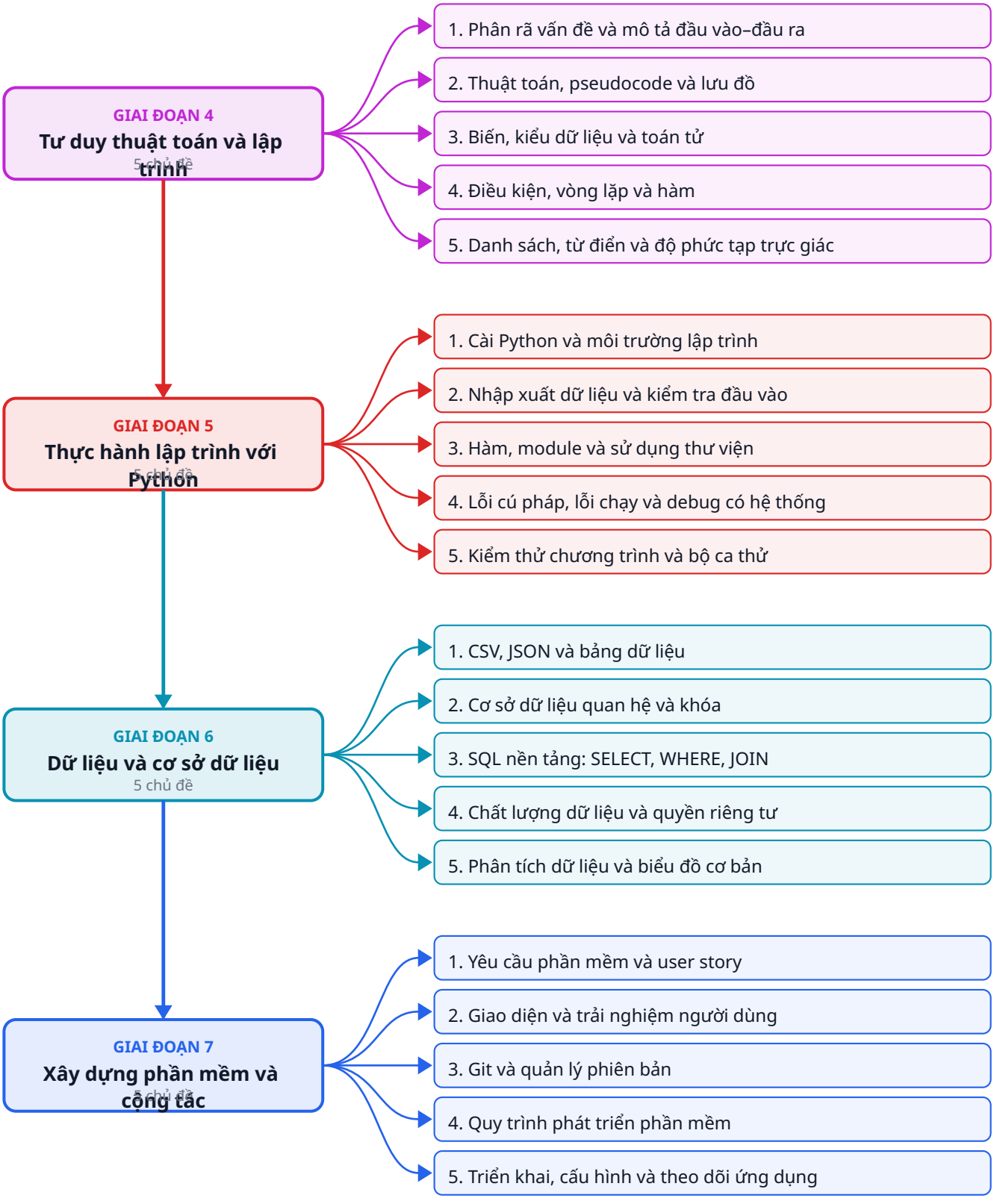
Giai đoạn 8: Hệ thống, bảo mật và điện toán đám mây
Hiểu hệ thống và rủi ro ở mức nhập môn, không biến thành khóa chuyên sâu về bảo mật. (5 chủ đề)

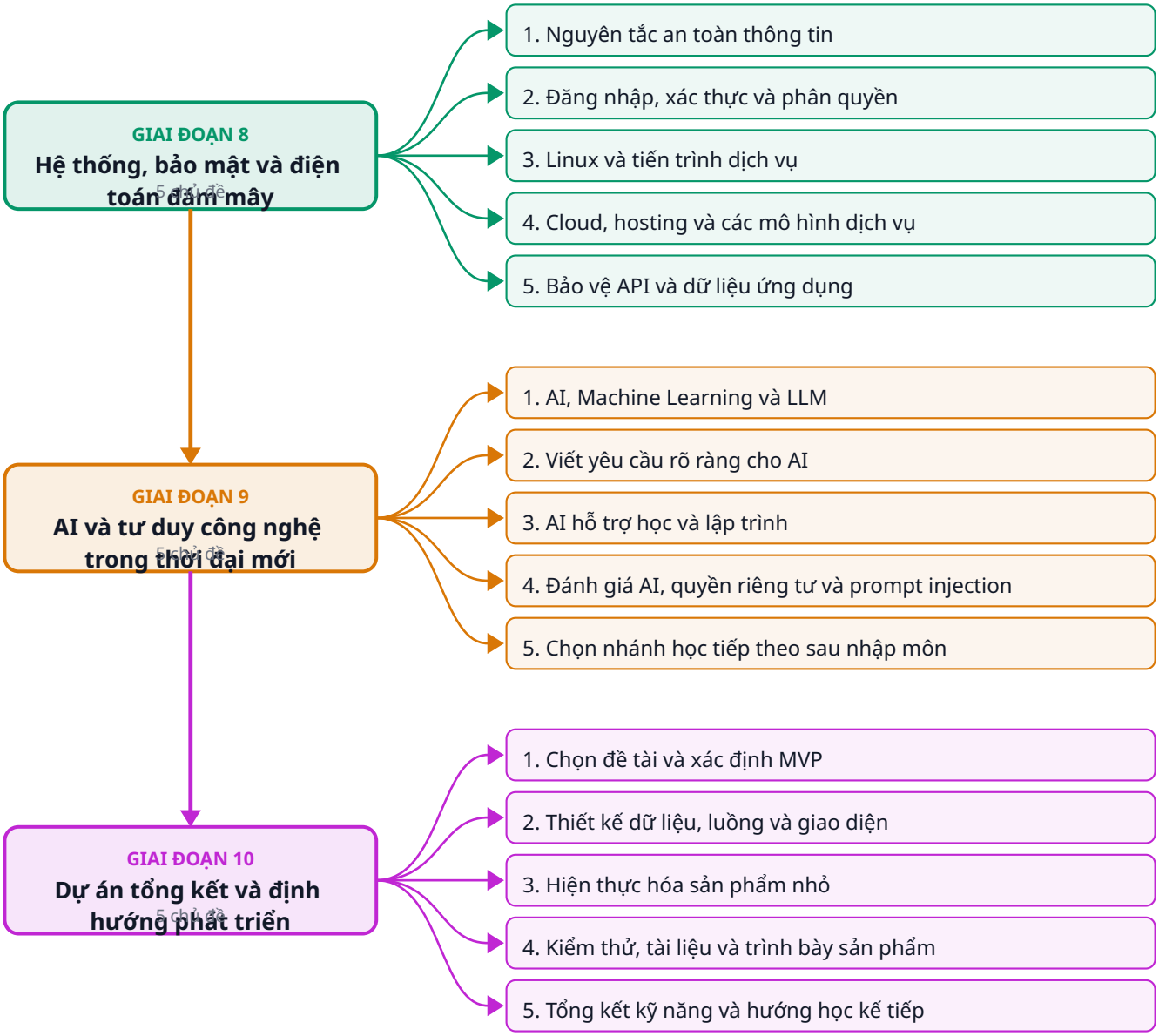
Giai đoạn 9: AI và tư duy công nghệ trong thời đại mới
Biết dùng AI để học và làm việc hiệu quả nhưng luôn kiểm chứng, bảo mật và tự chịu trách nhiệm. (5 chủ đề)

Giai đoạn 10: Dự án tổng kết và định hướng phát triển
Biến kiến thức thành minh chứng năng lực qua dự án cuối khóa và kế hoạch học tiếp. (5 chủ đề)

SƠ ĐỒ LỘ TRÌNH




GIAI ĐOẠN 4
Tư duy thuật toán và lập trình
5 chủ đề
1. Phân rã vấn đề và mô tả đầu vào–đầu ra
2. Thuật toán, pseudocode và lưu đồ
3. Biến, kiểu dữ liệu và toán tử
4. Điều kiện, vòng lặp và hàm
5. Danh sách, từ điển và độ phức tạp trực giác
GIAI ĐOẠN 5
Thực hành lập trình với Python
5 chủ đề
1. Cài Python và môi trường lập trình
2. Nhập xuất dữ liệu và kiểm tra đầu vào
3. Hàm, module và sử dụng thư viện
4. Lỗi cú pháp, lỗi chạy và debug có hệ thống
5. Kiểm thử chương trình và bộ ca thử
GIAI ĐOẠN 6
Dữ liệu và cơ sở dữ liệu
5 chủ đề
1. CSV, JSON và bảng dữ liệu
2. Cơ sở dữ liệu quan hệ và khóa
3. SQL nền tảng: SELECT, WHERE, JOIN
4. Chất lượng dữ liệu và quyền riêng tư
5. Phân tích dữ liệu và biểu đồ cơ bản
GIAI ĐOẠN 7
Xây dựng phần mềm và cộng tác
5 chủ đề
1. Yêu cầu phần mềm và user story
2. Giao diện và trải nghiệm người dùng
3. Git và quản lý phiên bản
4. Quy trình phát triển phần mềm
5. Triển khai, cấu hình và theo dõi ứng dụng


GIAI ĐOẠN 8
Hệ thống, bảo mật và điện toán đám mây
5 chủ đề
1. Nguyên tắc an toàn thông tin
2. Đăng nhập, xác thực và phân quyền
3. Linux và tiến trình dịch vụ
4. Cloud, hosting và các mô hình dịch vụ
5. Bảo vệ API và dữ liệu ứng dụng
GIAI ĐOẠN 9
AI và tư duy công nghệ trong thời đại mới
5 chủ đề
1. AI, Machine Learning và LLM
2. Viết yêu cầu rõ ràng cho AI
3. AI hỗ trợ học và lập trình
4. Đánh giá AI, quyền riêng tư và prompt injection
5. Chọn nhánh học tiếp theo sau nhập môn
GIAI ĐOẠN 10
Dự án tổng kết và định hướng phát triển
5 chủ đề
1. Chọn đề tài và xác định MVP
2. Thiết kế dữ liệu, luồng và giao diện
3. Hiện thực hóa sản phẩm nhỏ
4. Kiểm thử, tài liệu và trình bày sản phẩm
5. Tổng kết kỹ năng và hướng học kế tiếp

GIAI ĐOẠN 1

Hiểu ngành Công nghệ thông tin

Định hướng trước khi chọn công cụ: hiểu ngành, sản phẩm số, nghề nghiệp và nguyên tắc học có trách nhiệm.

1. CNTT, Khoa học máy tính và Kỹ thuật phần mềm

Phân biệt ngành nghề, tri thức nền và công việc thực tế

Mục tiêu học tập

- Giải thích đúng IT và Computer Science bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Software Engineering, Information Systems
- Thực hiện và giải thích kết quả bài tập: Vẽ sơ đồ 4 khối: hạ tầng, dữ liệu, phần mềm, người dùng; mô tả vị trí của một ứng dụng đặt lịch.
- Nhận diện và tránh sai lầm: Nhầm CNTT chỉ là viết code hoặc sửa máy tính.

Kiến thức cần học

IT · Computer Science · Software Engineering · Information Systems

Giải thích chi tiết

Vì sao cần học

Phân biệt ngành nghề, tri thức nền và công việc thực tế. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- IT: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Computer Science: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Software Engineering: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Information Systems: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức CNTT, Khoa học máy tính và Kỹ thuật phần mềm ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: IT, Computer Science, Software Engineering, Information Systems.
3. Vẽ sơ đồ 4 khối: hạ tầng, dữ liệu, phần mềm, người dùng; mô tả vị trí của một ứng dụng đặt lịch.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI so sánh ba nghề bằng ví dụ, rồi đối chiếu mô tả của cơ sở đào tạo.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "CNTT, Khoa học máy tính và Kỹ thuật phần mềm" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm CNTT chỉ là viết code hoặc sửa máy tính.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được IT, Computer Science, Software Engineering; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: CNTT, Khoa học máy tính và Kỹ thuật phần mềm
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được IT và Computer Science bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm IT, Computer Science, Software Engineering, Information Systems
- [ ] Đã làm: Vẽ sơ đồ 4 khối: hạ tầng, dữ liệu, phần mềm, người dùng; mô tả vị trí của một ứng dụng đặt lịch.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Nhầm CNTT chỉ là viết code hoặc sửa máy tính.
- [ ] Đã tự kiểm chứng: Nhờ AI so sánh ba nghề bằng ví dụ, rồi đối chiếu mô tả của cơ sở đào tạo.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS2023 Knowledge Areas Tài liệu
https://csed.acm.org/knowledge-areas/

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

2. Phần mềm và sản phẩm số hoạt động ra sao

Từ nhu cầu người dùng tới frontend, backend, database và vận hành

Mục tiêu học tập

- Giải thích đúng Product và Frontend bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Backend, Database, API, Cloud
- Thực hiện và giải thích kết quả bài tập: Lập sơ đồ đường đi khi người dùng nhấn Đăng nhập trong một ứng dụng.
- Nhận diện và tránh sai lầm: Tưởng giao diện đẹp đồng nghĩa sản phẩm đã hoàn chỉnh.

Kiến thức cần học

Product · Frontend · Backend · Database · API · Cloud

Giải thích chi tiết

Vì sao cần học

Từ nhu cầu người dùng tới frontend, backend, database và vận hành. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Product: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Frontend: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Backend: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Database: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- API: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Cloud: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Phần mềm và sản phẩm số hoạt động ra sao ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Product, Frontend, Backend, Database.
3. Lập sơ đồ đường đi khi người dùng nhấn Đăng nhập trong một ứng dụng.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Cho AI phê bình sơ đồ nhưng tự kiểm chứng vai trò từng thành phần.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Phần mềm và sản phẩm số hoạt động ra sao" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tưởng giao diện đẹp đồng nghĩa sản phẩm đã hoàn chỉnh.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Product, Frontend, Backend; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Phần mềm và sản phẩm số hoạt động ra sao
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Product và Frontend bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Product, Frontend, Backend, Database

[ ] Đã làm: Lập sơ đồ đường đi khi người dùng nhấn Đăng nhập trong một ứng dụng.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Tưởng giao diện đẹp đồng nghĩa sản phẩm đã hoàn chỉnh.

[ ] Đã tự kiểm chứng: Cho AI phê bình sơ đồ nhưng tự kiểm chứng vai trò từng thành phần.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS2023 Knowledge Areas Tài liệu
https://csed.acm.org/knowledge-areas/

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

Nên học trước

CNTT, Khoa học máy tính và Kỹ thuật phần mềm

3. Bản đồ nghề nghiệp trong CNTT

Nhận biết lộ trình Web, Mobile, AI, Data, Security, Cloud, QA, BA và UI/UX

Mục tiêu học tập

- Giải thích đúng Frontend và Backend bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Full-stack, Data, AI, Cybersecurity, QA, DevOps, BA, UI/UX
- Thực hiện và giải thích kết quả bài tập: Chọn ba vị trí nghề và tìm hiểu một ngày làm việc, kỹ năng cốt lõi, kết quả bàn giao.
- Nhận diện và tránh sai lầm: Chọn nghề chỉ vì xu hướng hoặc mức lương nghe kể.

Kiến thức cần học

Frontend · Backend · Full-stack · Data · AI · Cybersecurity · QA · DevOps · BA · UI/UX

Giải thích chi tiết

Vì sao cần học

Nhận biết lộ trình Web, Mobile, AI, Data, Security, Cloud, QA, BA và UI/UX. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Frontend: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Backend: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Full-stack: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Data: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- AI: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Cybersecurity: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- QA: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- DevOps: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- BA: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- UI/UX: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Bản đồ nghề nghiệp trong CNTT ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.

2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Frontend, Backend, Full-stack, Data.
3. Chọn ba vị trí nghề và tìm hiểu một ngày làm việc, kỹ năng cốt lõi, kết quả bàn giao.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Yêu cầu AI mô tả công việc rồi so với 3 tin tuyển dụng thực tế.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Bản đồ nghề nghiệp trong CNTT" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chọn nghề chỉ vì xu hướng hoặc mức lương nghe kể.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Frontend, Backend, Full-stack; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Bản đồ nghề nghiệp trong CNTT
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Frontend và Backend bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Frontend, Backend, Full-stack, Data
[ ] Đã làm: Chọn ba vị trí nghề và tìm hiểu một ngày làm việc, kỹ năng cốt lõi, kết quả bàn giao.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Chọn nghề chỉ vì xu hướng hoặc mức lương nghe kể.
[ ] Đã tự kiểm chứng: Yêu cầu AI mô tả công việc rồi so với 3 tin tuyển dụng thực tế.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS2023 Knowledge Areas Tài liệu
https://csed.acm.org/knowledge-areas/

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

Nên học trước

Phần mềm và sản phẩm số hoạt động ra sao

---

4. Học CNTT hiệu quả và tự giải quyết vấn đề

Học theo chu trình đọc–thử–sai–sửa–giải thích lại

Mục tiêu học tập

- Giải thích đúng Active Learning và Deliberate Practice bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Spaced Repetition, Debugging Mindset
- Thực hiện và giải thích kết quả bài tập: Thiết lập nhật ký học với cột đã hiểu, đã thử, lỗi, cách sửa và câu hỏi còn mở.
- Nhận diện và tránh sai lầm: Chỉ xem video hoặc chép code AI mà không tự chạy.

Kiến thức cần học

Active Learning · Deliberate Practice · Spaced Repetition · Debugging Mindset

Giải thích chi tiết

Vì sao cần học

Học theo chu trình đọc–thử–sai–sửa–giải thích lại. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Active Learning: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Deliberate Practice: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Spaced Repetition: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Debugging Mindset: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Học CNTT hiệu quả và tự giải quyết vấn đề ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Active Learning, Deliberate Practice, Spaced Repetition, Debugging Mindset.
3. Thiết lập nhật ký học với cột đã hiểu, đã thử, lỗi, cách sửa và câu hỏi còn mở.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI làm gia sư đặt câu hỏi thay vì đưa ngay lời giải.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Học CNTT hiệu quả và tự giải quyết vấn đề" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chỉ xem video hoặc chép code AI mà không tự chạy.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Active Learning, Deliberate Practice, Spaced Repetition; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Học CNTT hiệu quả và tự giải quyết vấn đề
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được Active Learning và Deliberate Practice bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm Active Learning, Deliberate Practice, Spaced Repetition, Debugging Mindset
- [ ] Đã làm: Thiết lập nhật ký học với cột đã hiểu, đã thử, lỗi, cách sửa và câu hỏi còn mở.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Chỉ xem video hoặc chép code AI mà không tự chạy.
- [ ] Đã tự kiểm chứng: Dùng AI làm gia sư đặt câu hỏi thay vì đưa ngay lời giải.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS2023 Knowledge Areas Tài liệu
https://csed.acm.org/knowledge-areas/

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

Nên học trước

Bản đồ nghề nghiệp trong CNTT

5. Đạo đức nghề và trách nhiệm số

Quyền riêng tư, bản quyền, thiên lệch và hệ quả xã hội của công nghệ

Mục tiêu học tập

- Giải thích đúng Privacy và Copyright bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Bias, Consent, Responsible AI
- Thực hiện và giải thích kết quả bài tập: Phân tích tình huống chia sẻ ảnh và dữ liệu khách hàng cho một chatbot công khai.
- Nhận diện và tránh sai lầm: Cho rằng dữ liệu tìm được trên mạng đều được dùng tùy ý.

Kiến thức cần học

Privacy · Copyright · Bias · Consent · Responsible AI

Giải thích chi tiết

Vì sao cần học

Quyền riêng tư, bản quyền, thiên lệch và hệ quả xã hội của công nghệ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Privacy: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Copyright: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Bias: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Consent: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Responsible AI: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Đạo đức nghề và trách nhiệm số ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Privacy, Copyright, Bias, Consent.
3. Phân tích tình huống chia sẻ ảnh và dữ liệu khách hàng cho một chatbot công khai.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không gửi dữ liệu cá nhân, bí mật hoặc tài liệu có hạn chế quyền cho AI.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Đạo đức nghề và trách nhiệm số" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cho rằng dữ liệu tìm được trên mạng đều được dùng tùy ý.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Privacy, Copyright, Bias; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Đạo đức nghề và trách nhiệm số
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Privacy và Copyright bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Privacy, Copyright, Bias, Consent

[ ] Đã làm: Phân tích tình huống chia sẻ ảnh và dữ liệu khách hàng cho một chatbot công khai.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Cho rằng dữ liệu tìm được trên mạng đều được dùng tùy ý.

[ ] Đã tự kiểm chứng: Không gửi dữ liệu cá nhân, bí mật hoặc tài liệu có hạn chế quyền cho AI.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS2023 Knowledge Areas Tài liệu
https://csed.acm.org/knowledge-areas/

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

Nên học trước

Học CNTT hiệu quả và tự giải quyết vấn đề

GIAI ĐOẠN 2

Máy tính và hệ điều hành

Làm chủ kiến thức kỹ thuật số tối thiểu: phần cứng, dữ liệu số, hệ điều hành, tệp và terminal.

1. CPU, RAM, SSD, GPU và thiết bị ngoại vi

Giải thích mỗi bộ phận làm gì và ảnh hưởng tới ứng dụng

Mục tiêu học tập

- Giải thích đúng CPU và RAM bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Storage, GPU, I/O
- Thực hiện và giải thích kết quả bài tập: Xem thông số máy hiện tại; giải thích vì sao RAM không thay thế SSD.
- Nhận diện và tránh sai lầm: Đồng nhất GHz với tốc độ toàn hệ thống.

Kiến thức cần học

CPU · RAM · Storage · GPU · I/O

Giải thích chi tiết

Vì sao cần học

Giải thích mỗi bộ phận làm gì và ảnh hưởng tới ứng dụng. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- CPU: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- RAM: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Storage: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- GPU: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- I/O: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức CPU, RAM, SSD, GPU và thiết bị ngoại vi ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: CPU, RAM, Storage, GPU.
3. Xem thông số máy hiện tại; giải thích vì sao RAM không thay thế SSD.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không chạy script tối ưu phần cứng do AI sinh nếu chưa hiểu tác dụng.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "CPU, RAM, SSD, GPU và thiết bị ngoại vi" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đồng nhất GHz với tốc độ toàn hệ thống.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được CPU, RAM, Storage; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: CPU, RAM, SSD, GPU và thiết bị ngoại vi
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được CPU và RAM bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm CPU, RAM, Storage, GPU
[ ] Đã làm: Xem thông số máy hiện tại; giải thích vì sao RAM không thay thế SSD.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Đồng nhất GHz với tốc độ toàn hệ thống.
[ ] Đã tự kiểm chứng: Không chạy script tối ưu phần cứng do AI sinh nếu chưa hiểu tác dụng.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Computer Science Field Guide Website
https://www.csfieldguide.org.nz/en/

Microsoft Windows Support Tài liệu
https://support.microsoft.com/windows

Linux Journey Khóa học
https://linuxjourney.com/

Nên học trước

Đạo đức nghề và trách nhiệm số

2. Bit, byte, nhị phân và mã hóa ký tự

Hiểu dữ liệu số, Unicode, UTF-8, kích thước tệp và đơn vị lưu trữ

Mục tiêu học tập

- Giải thích đúng Bit và Byte bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Binary, Unicode, UTF-8, Encoding
- Thực hiện và giải thích kết quả bài tập: Đổi 13 sang nhị phân; lưu một câu tiếng Việt UTF-8 và xem kích thước byte.
- Nhận diện và tránh sai lầm: Nhầm ký tự với byte; nhầm MB và MiB.

Kiến thức cần học

Bit · Byte · Binary · Unicode · UTF-8 · Encoding

Giải thích chi tiết

Vì sao cần học

Hiểu dữ liệu số, Unicode, UTF-8, kích thước tệp và đơn vị lưu trữ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Bit: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Byte: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Binary: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Unicode: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- UTF-8: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Encoding: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Bit, byte, nhị phân và mã hóa ký tự ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Bit, Byte, Binary, Unicode.
3. Đổi 13 sang nhị phân; lưu một câu tiếng Việt UTF-8 và xem kích thước byte.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.

- Phải tự kiểm chứng: Tự tính lại chuyển đổi thay vì tin đáp án AI.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Bit, byte, nhị phân và mã hóa ký tự" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm ký tự với byte; nhầm MB và MiB.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Bit, Byte, Binary; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Bit, byte, nhị phân và mã hóa ký tự

```
number = 13
print(bin(number))
print(len("Xin chào".encode("utf-8")))
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Bit và Byte bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Bit, Byte, Binary, Unicode
[ ] Đã làm: Đổi 13 sang nhị phân; lưu một câu tiếng Việt UTF-8 và xem kích thước byte.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Nhầm ký tự với byte; nhầm MB và MiB.
[ ] Đã tự kiểm chứng: Tự tính lại chuyển đổi thay vì tin đáp án AI.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Computer Science Field Guide Website
https://www.csfieldguide.org.nz/en/

Microsoft Windows Support Tài liệu
https://support.microsoft.com/windows

Linux Journey Khóa học
https://linuxjourney.com/

Nên học trước

CPU, RAM, SSD, GPU và thiết bị ngoại vi

3. Hệ điều hành, tiến trình và bộ nhớ

Vai trò của Windows, macOS, Linux; ứng dụng và tiến trình

Mục tiêu học tập

- Giải thích đúng Operating System và Process bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Thread, Memory, Permissions
- Thực hiện và giải thích kết quả bài tập: Mở trình quản lý tác vụ, tìm ba tiến trình và giải thích chức năng.
- Nhận diện và tránh sai lầm: Tắt các tiến trình hệ thống một cách tùy tiện.

Kiến thức cần học

Operating System · Process · Thread · Memory · Permissions

Giải thích chi tiết

Vì sao cần học

Vai trò của Windows, macOS, Linux; ứng dụng và tiến trình. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Operating System: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Process: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Thread: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Memory: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Permissions: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Hệ điều hành, tiến trình và bộ nhớ ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Operating System, Process, Thread, Memory.
3. Mở trình quản lý tác vụ, tìm ba tiến trình và giải thích chức năng.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Chỉ quan sát, không yêu cầu AI xóa hay dừng tiến trình lạ.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Hệ điều hành, tiến trình và bộ nhớ" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tắt các tiến trình hệ thống một cách tùy tiện.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Operating System, Process, Thread; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Hệ điều hành, tiến trình và bộ nhớ
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Operating System và Process bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Operating System, Process, Thread, Memory

[ ] Đã làm: Mở trình quản lý tác vụ, tìm ba tiến trình và giải thích chức năng.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Tắt các tiến trình hệ thống một cách tùy tiện.

[ ] Đã tự kiểm chứng: Chỉ quan sát, không yêu cầu AI xóa hay dừng tiến trình lạ.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Computer Science Field Guide Website
https://www.csfieldguide.org.nz/en/

Microsoft Windows Support Tài liệu
https://support.microsoft.com/windows

Linux Journey Khóa học
https://linuxjourney.com/

Nên học trước

Bit, byte, nhị phân và mã hóa ký tự

4. Tệp, thư mục, phần mở rộng và sao lưu

Quản lý đường dẫn, phân loại file, nén và đồng bộ

Mục tiêu học tập

- Giải thích đúng Path và Extension bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Filesystem, Archive, Backup
- Thực hiện và giải thích kết quả bài tập: Tạo cây thư mục khoa học và sao lưu một tài liệu bằng hai nơi lưu trữ.
- Nhận diện và tránh sai lầm: Đổi đuôi tệp và nghĩ đã chuyển đổi định dạng.

Kiến thức cần học

Path · Extension · Filesystem · Archive · Backup

Giải thích chi tiết

Vì sao cần học

Quản lý đường dẫn, phân loại file, nén và đồng bộ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Path: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Extension: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Filesystem: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Archive: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Backup: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Tệp, thư mục, phần mở rộng và sao lưu ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Path, Extension, Filesystem, Archive.
3. Tạo cây thư mục khoa học và sao lưu một tài liệu bằng hai nơi lưu trữ.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không dùng lệnh xóa hàng loạt do AI gợi ý nếu chưa có bản sao.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Tệp, thư mục, phần mở rộng và sao lưu" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đổi đuôi tệp và nghĩ đã chuyển đổi định dạng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Path, Extension, Filesystem; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Tệp, thư mục, phần mở rộng và sao lưu
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Path và Extension bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Path, Extension, Filesystem, Archive

[ ] Đã làm: Tạo cây thư mục khoa học và sao lưu một tài liệu bằng hai nơi lưu trữ.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Đổi đuôi tệp và nghĩ đã chuyển đổi định dạng.

[ ] Đã tự kiểm chứng: Không dùng lệnh xóa hàng loạt do AI gợi ý nếu chưa có bản sao.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Computer Science Field Guide Website
https://www.csfieldguide.org.nz/en/

Microsoft Windows Support Tài liệu
https://support.microsoft.com/windows

Linux Journey Khóa học
https://linuxjourney.com/

Nên học trước

Hệ điều hành, tiến trình và bộ nhớ

5. Dòng lệnh và terminal cho người mới

Làm quen pwd, ls, cd, mkdir, cat và trợ giúp lệnh

Mục tiêu học tập

- Giải thích đúng Terminal và Shell bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: CLI, Working Directory, PATH
- Thực hiện và giải thích kết quả bài tập: Tạo thư mục it-lab, di chuyển, tạo tệp README và liệt kê nội dung.
- Nhận diện và tránh sai lầm: Chạy lệnh trên sai thư mục hoặc dán lệnh tải từ nguồn lạ.

Kiến thức cần học

Terminal · Shell · CLI · Working Directory · PATH

Giải thích chi tiết

Vì sao cần học

Làm quen pwd, ls, cd, mkdir, cat và trợ giúp lệnh. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Terminal: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Shell: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- CLI: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Working Directory: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- PATH: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Dòng lệnh và terminal cho người mới ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Terminal, Shell, CLI, Working Directory.
3. Tạo thư mục it-lab, di chuyển, tạo tệp README và liệt kê nội dung.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Yêu cầu AI giải thích từng tham số và tác động trước khi chạy lệnh.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Dòng lệnh và terminal cho người mới" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chạy lệnh trên sai thư mục hoặc dán lệnh tải từ nguồn lạ.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Terminal, Shell, CLI; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Dòng lệnh và terminal cho người mới

```
pwd
mkdir it-lab
cd it-lab
printf "# IT Lab\n" > README.md
ls -la
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Terminal và Shell bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Terminal, Shell, CLI, Working Directory
[ ] Đã làm: Tạo thư mục it-lab, di chuyển, tạo tệp README và liệt kê nội dung.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Chạy lệnh trên sai thư mục hoặc dán lệnh tải từ nguồn lạ.
[ ] Đã tự kiểm chứng: Yêu cầu AI giải thích từng tham số và tác động trước khi chạy lệnh.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Computer Science Field Guide Website
https://www.csfieldguide.org.nz/en/

Microsoft Windows Support Tài liệu
https://support.microsoft.com/windows

Linux Journey Khóa học
https://linuxjourney.com/

Nên học trước

Tệp, thư mục, phần mở rộng và sao lưu

GIAI ĐOẠN 3

Internet, Web và an toàn trực tuyến

Tìm hiểu Internet và Web như một hệ thống; đồng thời rèn thói quen an toàn số.

1. Mạng máy tính cơ bản

Thiết bị, LAN/WAN, router, IP và gói tin

Mục tiêu học tập

- Giải thích đúng Network và LAN bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: WAN, Router, IP Address, Packet
- Thực hiện và giải thích kết quả bài tập: Vẽ mạng trong nhà từ thiết bị tới router rồi tới nhà cung cấp dịch vụ.
- Nhận diện và tránh sai lầm: Nhầm Wi-Fi với Internet.

Kiến thức cần học

Network · LAN · WAN · Router · IP Address · Packet

Giải thích chi tiết

Vì sao cần học

Thiết bị, LAN/WAN, router, IP và gói tin. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Network: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- LAN: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- WAN: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Router: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- IP Address: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Packet: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Mạng máy tính cơ bản ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Network, LAN, WAN, Router.
3. Vẽ mạng trong nhà từ thiết bị tới router rồi tới nhà cung cấp dịch vụ.

4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Tự kiểm chứng địa chỉ riêng/tuyến đường thay vì suy đoán từ AI.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Mạng máy tính cơ bản" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm Wi-Fi với Internet.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Network, LAN, WAN; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Mạng máy tính cơ bản
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Network và LAN bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Network, LAN, WAN, Router
[ ] Đã làm: Vẽ mạng trong nhà từ thiết bị tới router rồi tới nhà cung cấp dịch vụ.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Nhầm Wi-Fi với Internet.
[ ] Đã tự kiểm chứng: Tự kiểm chứng địa chỉ riêng/tuyến đường thay vì suy đoán từ AI.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

MDN: HTTP overview Tài liệu
https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview

Cloudflare Learning Center Website
https://www.cloudflare.com/learning/

Nên học trước

Dòng lệnh và terminal cho người mới

2. Tên miền, DNS và địa chỉ web

URL, domain, DNS record và quá trình phân giải

Mục tiêu học tập

- Giải thích đúng URL và Domain bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: DNS, A Record, CNAME
- Thực hiện và giải thích kết quả bài tập: Phân tích https://example.com/docs?lang=vi thành các phần và giải thích vai trò DNS.
- Nhận diện và tránh sai lầm: Tưởng mọi tên miền là máy chủ hoặc luôn trỏ trực tiếp tới một IP.

Kiến thức cần học

URL · Domain · DNS · A Record · CNAME

Giải thích chi tiết

Vì sao cần học

URL, domain, DNS record và quá trình phân giải. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- URL: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Domain: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- DNS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- A Record: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- CNAME: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Tên miền, DNS và địa chỉ web ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: URL, Domain, DNS, A Record.
3. Phân tích https://example.com/docs?lang=vi thành các phần và giải thích vai trò DNS.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không thay đổi DNS thực tế chỉ dựa trên câu trả lời AI.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Tên miền, DNS và địa chỉ web" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tưởng mọi tên miền là máy chủ hoặc luôn trỏ trực tiếp tới một IP.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được URL, Domain, DNS; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Tên miền, DNS và địa chỉ web
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được URL và Domain bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm URL, Domain, DNS, A Record
[ ] Đã làm: Phân tích https://example.com/docs?lang=vi thành các phần và giải thích vai trò DNS.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Tưởng mọi tên miền là máy chủ hoặc luôn trỏ trực tiếp tới một IP.
[ ] Đã tự kiểm chứng: Không thay đổi DNS thực tế chỉ dựa trên câu trả lời AI.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

MDN: HTTP overview Tài liệu
https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview

Cloudflare Learning Center Website
https://www.cloudflare.com/learning/

Nên học trước

Mạng máy tính cơ bản

3. Client–Server, HTTP và API

Request/response, status code, phương thức GET/POST và JSON

Mục tiêu học tập

- Giải thích đúng Client và Server bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: HTTP, Status Code, REST, JSON
- Thực hiện và giải thích kết quả bài tập: Mở DevTools > Network, tải lại trang công khai và quan sát request, response, status.
- Nhận diện và tránh sai lầm: Nghĩ HTTP 200 đồng nghĩa dữ liệu đúng về nghiệp vụ.

Kiến thức cần học

Client · Server · HTTP · Status Code · REST · JSON

Giải thích chi tiết

Vì sao cần học

Request/response, status code, phương thức GET/POST và JSON. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Client: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Server: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- HTTP: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Status Code: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- REST: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- JSON: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Client–Server, HTTP và API ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Client, Server, HTTP, Status Code.
3. Mở DevTools > Network, tải lại trang công khai và quan sát request, response, status.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.

- Phải tự kiểm chứng: Che cookie, token và thông tin nhạy cảm trước khi gửi log cho AI.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Client–Server, HTTP và API" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nghĩ HTTP 200 đồng nghĩa dữ liệu đúng về nghiệp vụ.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Client, Server, HTTP; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Client-Server, HTTP và API
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Client và Server bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Client, Server, HTTP, Status Code

[ ] Đã làm: Mở DevTools > Network, tải lại trang công khai và quan sát request, response, status.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Nghĩ HTTP 200 đồng nghĩa dữ liệu đúng về nghiệp vụ.

[ ] Đã tự kiểm chứng: Che cookie, token và thông tin nhạy cảm trước khi gửi log cho AI.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

MDN: HTTP overview Tài liệu
https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview

Cloudflare Learning Center Website
https://www.cloudflare.com/learning/

Nên học trước

Tên miền, DNS và địa chỉ web

4. Trình duyệt hiển thị website như thế nào

Phân biệt HTML, CSS, JavaScript, DOM và rendering

Mục tiêu học tập

- Giải thích đúng HTML và CSS bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: JavaScript, DOM, Browser
- Thực hiện và giải thích kết quả bài tập: Tạo trang HTML nhỏ, đổi một thẻ và quan sát bằng Inspect Element.
- Nhận diện và tránh sai lầm: Nhầm HTML là ngôn ngữ lập trình tổng quát.

Kiến thức cần học

HTML · CSS · JavaScript · DOM · Browser

Giải thích chi tiết

Vì sao cần học

Phân biệt HTML, CSS, JavaScript, DOM và rendering. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- HTML: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- CSS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- JavaScript: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- DOM: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Browser: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Trình duyệt hiển thị website như thế nào ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: HTML, CSS, JavaScript, DOM.
3. Tạo trang HTML nhỏ, đổi một thẻ và quan sát bằng Inspect Element.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI giải thích DOM nhưng tự kiểm chứng trong DevTools.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Trình duyệt hiển thị website như thế nào" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm HTML là ngôn ngữ lập trình tổng quát.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được HTML, CSS, JavaScript; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Trình duyệt hiển thị website như thế nào

```
<!doctype html>
<html lang="vi">
<head><meta charset="utf-8"><title>Trang đầu tiên</title></head>
<body><h1>Xin chào CNTT</h1><p>Đây là HTML.</p></body>
</html>
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được HTML và CSS bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm HTML, CSS, JavaScript, DOM

[ ] Đã làm: Tạo trang HTML nhỏ, đổi một thẻ và quan sát bằng Inspect Element.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Nhầm HTML là ngôn ngữ lập trình tổng quát.

[ ] Đã tự kiểm chứng: Dùng AI giải thích DOM nhưng tự kiểm chứng trong DevTools.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

MDN: HTTP overview Tài liệu
https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview

Cloudflare Learning Center Website
https://www.cloudflare.com/learning/

Nên học trước

Client–Server, HTTP và API

5. An toàn tài khoản và đánh giá nguồn trên Internet

Mật khẩu, MFA, phishing, HTTPS và kiểm tra nguồn

Mục tiêu học tập

- Giải thích đúng MFA và Phishing bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: HTTPS, Password Manager, Digital Literacy
- Thực hiện và giải thích kết quả bài tập: Phân biệt các dấu hiệu email lừa đảo qua ví dụ giả lập và cấu hình MFA cho tài khoản thử.
- Nhận diện và tránh sai lầm: Tin mọi trang có biểu tượng ổ khóa là uy tín.

Kiến thức cần học

MFA · Phishing · HTTPS · Password Manager · Digital Literacy

Giải thích chi tiết

Vì sao cần học

Mật khẩu, MFA, phishing, HTTPS và kiểm tra nguồn. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- MFA: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Phishing: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- HTTPS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Password Manager: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Digital Literacy: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức An toàn tài khoản và đánh giá nguồn trên Internet ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: MFA, Phishing, HTTPS, Password Manager.
3. Phân biệt các dấu hiệu email lừa đảo qua ví dụ giả lập và cấu hình MFA cho tài khoản thử.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không nhập mật khẩu, OTP hoặc khóa dự phòng vào chatbot.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "An toàn tài khoản và đánh giá nguồn trên Internet" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tin mọi trang có biểu tượng ổ khóa là uy tín.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được MFA, Phishing, HTTPS; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: An toàn tài khoản và đánh giá nguồn trên Internet
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được MFA và Phishing bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm MFA, Phishing, HTTPS, Password Manager
[ ] Đã làm: Phân biệt các dấu hiệu email lừa đảo qua ví dụ giả lập và cấu hình MFA cho tài khoản thử.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Tin mọi trang có biểu tượng ổ khóa là uy tín.
[ ] Đã tự kiểm chứng: Không nhập mật khẩu, OTP hoặc khóa dự phòng vào chatbot.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

MDN: HTTP overview Tài liệu
https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview

Cloudflare Learning Center Website
https://www.cloudflare.com/learning/

Nên học trước

Trình duyệt hiển thị website như thế nào

GIAI ĐOẠN 4

Tư duy thuật toán và lập trình

Luyện tư duy thuật toán độc lập với ngôn ngữ lập trình.

1. Phân rã vấn đề và mô tả đầu vào–đầu ra

Xác định mục tiêu, dữ liệu vào, dữ liệu ra và điều kiện ràng buộc

Mục tiêu học tập

- Giải thích đúng Problem Statement và Input bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Output, Constraint, Edge Case
- Thực hiện và giải thích kết quả bài tập: Mô tả bài toán tính tiền mua hàng, gồm chiết khấu và trường hợp giỏ hàng trống.
- Nhận diện và tránh sai lầm: Nhảy vào code trước khi xác định bài toán.

Kiến thức cần học

Problem Statement · Input · Output · Constraint · Edge Case

Giải thích chi tiết

Vì sao cần học

Xác định mục tiêu, dữ liệu vào, dữ liệu ra và điều kiện ràng buộc. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Problem Statement: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Input: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Output: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Constraint: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Edge Case: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Phân rã vấn đề và mô tả đầu vào–đầu ra ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Problem Statement, Input, Output, Constraint.
3. Mô tả bài toán tính tiền mua hàng, gồm chiết khấu và trường hợp giỏ hàng trống.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.

5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI nêu trường hợp biên, tự quyết định yêu cầu nghiệp vụ.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Phân rã vấn đề và mô tả đầu vào–đầu ra" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhảy vào code trước khi xác định bài toán.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Problem Statement, Input, Output; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Phân rã vấn đề và mô tả đầu vào-đầu ra
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Problem Statement và Input bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Problem Statement, Input, Output, Constraint
[ ] Đã làm: Mô tả bài toán tính tiền mua hàng, gồm chiết khấu và trường hợp giỏ hàng trống.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Nhảy vào code trước khi xác định bài toán.
[ ] Đã tự kiểm chứng: Nhờ AI nêu trường hợp biên, tự quyết định yêu cầu nghiệp vụ.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

VisuAlgo Website
https://visualgo.net/en

Nên học trước

An toàn tài khoản và đánh giá nguồn trên Internet

2. Thuật toán, pseudocode và lưu đồ

Tuần tự, rẽ nhánh, lặp, tính đúng và tính kết thúc

Mục tiêu học tập

- Giải thích đúng Algorithm và Pseudocode bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Flowchart, Branch, Loop
- Thực hiện và giải thích kết quả bài tập: Viết giả mã kiểm tra số chẵn/lẻ và tìm số lớn nhất trong ba số.
- Nhận diện và tránh sai lầm: Viết thuật toán mơ hồ, thiếu điều kiện dừng.

Kiến thức cần học

Algorithm · Pseudocode · Flowchart · Branch · Loop

Giải thích chi tiết

Vì sao cần học

Tuần tự, rẽ nhánh, lặp, tính đúng và tính kết thúc. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Algorithm: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Pseudocode: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Flowchart: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Branch: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Loop: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Thuật toán, pseudocode và lưu đồ ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Algorithm, Pseudocode, Flowchart, Branch.
3. Viết giả mã kiểm tra số chẵn/lẻ và tìm số lớn nhất trong ba số.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Đề nghị AI đóng vai người kiểm thử phản ví dụ.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Thuật toán, pseudocode và lưu đồ" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Viết thuật toán mơ hồ, thiếu điều kiện dừng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Algorithm, Pseudocode, Flowchart; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Thuật toán, pseudocode và lưu đồ
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Algorithm và Pseudocode bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Algorithm, Pseudocode, Flowchart, Branch

[ ] Đã làm: Viết giả mã kiểm tra số chẵn/lẻ và tìm số lớn nhất trong ba số.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Viết thuật toán mơ hồ, thiếu điều kiện dừng.

[ ] Đã tự kiểm chứng: Đề nghị AI đóng vai người kiểm thử phản ví dụ.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

VisuAlgo Website
https://visualgo.net/en

Nên học trước

Phân rã vấn đề và mô tả đầu vào–đầu ra

---

3. Biến, kiểu dữ liệu và toán tử

Số, chuỗi, boolean, biểu thức và ép kiểu

Mục tiêu học tập

- Giải thích đúng Variable và Integer bằng ngôn ngữ phổ thông

- Phân biệt được các khái niệm liên quan: Float, String, Boolean, Operator
- Thực hiện và giải thích kết quả bài tập: Dùng Python tạo biến tên, tuổi, điểm trung bình; in ra kết quả và kiểm tra kiểu.
- Nhận diện và tránh sai lầm: Cộng chuỗi với số mà không hiểu ép kiểu.

Kiến thức cần học

Variable · Integer · Float · String · Boolean · Operator

Giải thích chi tiết

Vì sao cần học

Số, chuỗi, boolean, biểu thức và ép kiểu. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Variable: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Integer: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Float: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- String: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Boolean: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Operator: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Biến, kiểu dữ liệu và toán tử ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Variable, Integer, Float, String.
3. Dùng Python tạo biến tên, tuổi, điểm trung bình; in ra kết quả và kiểm tra kiểu.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không chỉ copy code mẫu, thử thay giá trị để dự đoán đầu ra.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Biến, kiểu dữ liệu và toán tử" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cộng chuỗi với số mà không hiểu ép kiểu.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Variable, Integer, Float; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Biến, kiểu dữ liệu và toán tử

```
name = "An"
age = 18
average = 8.5
print(name, age, average, type(average))
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Variable và Integer bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Variable, Integer, Float, String

[ ] Đã làm: Dùng Python tạo biến tên, tuổi, điểm trung bình; in ra kết quả và kiểm tra kiểu.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Cộng chuỗi với số mà không hiểu ép kiểu.

[ ] Đã tự kiểm chứng: Không chỉ copy code mẫu, thử thay giá trị để dự đoán đầu ra.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

VisuAlgo Website
https://visualgo.net/en

Nên học trước

Thuật toán, pseudocode và lưu đồ

---

4. Điều kiện, vòng lặp và hàm

if/else, for/while, tham số, return và phạm vi biến

Mục tiêu học tập

- Giải thích đúng Condition và Loop bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Function, Parameter, Return, Scope
- Thực hiện và giải thích kết quả bài tập: Viết chương trình tính tổng số chẵn từ 1 đến n và tách thành hàm.
- Nhận diện và tránh sai lầm: Vòng lặp vô hạn hoặc hàm không trả kết quả mong muốn.

Kiến thức cần học

Condition · Loop · Function · Parameter · Return · Scope

Giải thích chi tiết

Vì sao cần học

if/else, for/while, tham số, return và phạm vi biến. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Condition: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Loop: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Function: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Parameter: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Return: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Scope: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Điều kiện, vòng lặp và hàm ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Condition, Loop, Function, Parameter.
3. Viết chương trình tính tổng số chẵn từ 1 đến n và tách thành hàm.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Yêu cầu AI gợi ý test trước khi xem đáp án.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Điều kiện, vòng lặp và hàm" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Vòng lặp vô hạn hoặc hàm không trả kết quả mong muốn.

- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Condition, Loop, Function; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Điều kiện, vòng lặp và hàm

```
def sum_even(n):
    return sum(i for i in range(1, n + 1) if i % 2 == 0)

print(sum_even(10))  # 30
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Condition và Loop bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Condition, Loop, Function, Parameter
[ ] Đã làm: Viết chương trình tính tổng số chẵn từ 1 đến n và tách thành hàm.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Vòng lặp vô hạn hoặc hàm không trả kết quả mong muốn.
[ ] Đã tự kiểm chứng: Yêu cầu AI gợi ý test trước khi xem đáp án.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

VisuAlgo Website
https://visualgo.net/en

Nên học trước

Biến, kiểu dữ liệu và toán tử

---

5. Danh sách, từ điển và độ phức tạp trực giác

List, dictionary, tìm kiếm và vì sao chọn cấu trúc dữ liệu phù hợp

Mục tiêu học tập

- Giải thích đúng List và Dictionary bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Set, Search, Big O
- Thực hiện và giải thích kết quả bài tập: Lưu danh sách sản phẩm; tra cứu theo mã bằng list rồi dictionary và giải thích khác biệt.
- Nhận diện và tránh sai lầm: Chỉ chọn cấu trúc vì code ngắn nhất.

Kiến thức cần học

List · Dictionary · Set · Search · Big O

Giải thích chi tiết

Vì sao cần học

List, dictionary, tìm kiếm và vì sao chọn cấu trúc dữ liệu phù hợp. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- List: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Dictionary: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Set: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Search: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Big O: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Danh sách, từ điển và độ phức tạp trực giác ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: List, Dictionary, Set, Search.
3. Lưu danh sách sản phẩm; tra cứu theo mã bằng list rồi dictionary và giải thích khác biệt.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Kiểm chứng bằng nhiều bộ dữ liệu và hiểu chi phí thao tác.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Danh sách, từ điển và độ phức tạp trực giác" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chỉ chọn cấu trúc vì code ngắn nhất.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được List, Dictionary, Set; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Danh sách, từ điển và độ phức tạp trực giác

```
products = [{"id": "P1", "name": "Bút"}, {"id": "P2", "name": "Vở"}]
by_id = {item["id"]: item for item in products}
print(by_id.get("P2"))
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được List và Dictionary bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm List, Dictionary, Set, Search

[ ] Đã làm: Lưu danh sách sản phẩm; tra cứu theo mã bằng list rồi dictionary và giải thích khác biệt.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Chỉ chọn cấu trúc vì code ngắn nhất.

[ ] Đã tự kiểm chứng: Kiểm chứng bằng nhiều bộ dữ liệu và hiểu chi phí thao tác.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

CS50x 2026 Khóa học
https://cs50.harvard.edu/x/

VisuAlgo Website
https://visualgo.net/en

Nên học trước

Điều kiện, vòng lặp và hàm

GIAI ĐOẠN 5

Thực hành lập trình với Python

Chuyển thuật toán thành chương trình Python nhỏ có thể chạy, kiểm thử và sửa lỗi.

1. Cài Python và môi trường lập trình

Interpreter, code editor, terminal, file .py và môi trường ảo

Mục tiêu học tập

- Giải thích đúng Python và Interpreter bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: VS Code, Virtual Environment
- Thực hiện và giải thích kết quả bài tập: Cài Python từ nguồn chính thức, chạy python --version và file hello.py.
- Nhận diện và tránh sai lầm: Cài package vào nhầm môi trường hoặc chạy mã không rõ nguồn.

Kiến thức cần học

Python · Interpreter · VS Code · Virtual Environment

Giải thích chi tiết

Vì sao cần học

Interpreter, code editor, terminal, file .py và môi trường ảo. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Python: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Interpreter: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- VS Code: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Virtual Environment: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Cài Python và môi trường lập trình ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Python, Interpreter, VS Code, Virtual Environment.
3. Cài Python từ nguồn chính thức, chạy python --version và file hello.py.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI đọc thông báo lỗi sau khi che đường dẫn và thông tin riêng.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Cài Python và môi trường lập trình" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cài package vào nhầm môi trường hoặc chạy mã không rõ nguồn.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Python, Interpreter, VS Code; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Cài Python và môi trường lập trình

```
print("Xin chào, thế giới CNTT!")
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Python và Interpreter bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Python, Interpreter, VS Code, Virtual Environment
[ ] Đã làm: Cài Python từ nguồn chính thức, chạy python --version và file hello.py.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Cài package vào nhầm môi trường hoặc chạy mã không rõ nguồn.
[ ] Đã tự kiểm chứng: Nhờ AI đọc thông báo lỗi sau khi che đường dẫn và thông tin riêng.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Python Tutorial Tài liệu
https://docs.python.org/3/tutorial/

Python venv Tài liệu
https://docs.python.org/3/library/venv.html

pytest documentation Tài liệu
https://docs.pytest.org/en/stable/

Nên học trước

Danh sách, từ điển và độ phức tạp trực giác

2. Nhập xuất dữ liệu và kiểm tra đầu vào

input, print, ép kiểu, xử lý chuỗi và dữ liệu không hợp lệ

Mục tiêu học tập

- Giải thích đúng Input và Output bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Validation, Parsing
- Thực hiện và giải thích kết quả bài tập: Viết chương trình nhận năm sinh, kiểm tra số và tính tuổi ước lượng theo năm.
- Nhận diện và tránh sai lầm: Cho rằng input luôn đúng kiểu dữ liệu.

Kiến thức cần học

Input · Output · Validation · Parsing

Giải thích chi tiết

Vì sao cần học

input, print, ép kiểu, xử lý chuỗi và dữ liệu không hợp lệ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Input: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Output: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Validation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Parsing: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Nhập xuất dữ liệu và kiểm tra đầu vào ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Input, Output, Validation, Parsing.
3. Viết chương trình nhận năm sinh, kiểm tra số và tính tuổi ước lượng theo năm.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Yêu cầu AI liệt kê các trường hợp trống/sai định dạng để tự thử.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Nhập xuất dữ liệu và kiểm tra đầu vào" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cho rằng input luôn đúng kiểu dữ liệu.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Input, Output, Validation; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Nhập xuất dữ liệu và kiểm tra đầu vào

```
raw = input("Năm sinh: ").strip()
if raw.isdigit() and 1900 <= int(raw) <= 2026:
    print("Tuổi ước lượng:", 2026 - int(raw))
else:
    print("Năm sinh không hợp lệ")
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Input và Output bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Input, Output, Validation, Parsing
[ ] Đã làm: Viết chương trình nhận năm sinh, kiểm tra số và tính tuổi ước lượng theo năm.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Cho rằng input luôn đúng kiểu dữ liệu.
[ ] Đã tự kiểm chứng: Yêu cầu AI liệt kê các trường hợp trống/sai định dạng để tự thử.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Python Tutorial Tài liệu
https://docs.python.org/3/tutorial/

Python venv Tài liệu
https://docs.python.org/3/library/venv.html

pytest documentation Tài liệu
https://docs.pytest.org/en/stable/

Nên học trước

Cài Python và môi trường lập trình

---

3. Hàm, module và sử dụng thư viện

Tách chương trình, import và tái sử dụng logic

Mục tiêu học tập

- Giải thích đúng Function và Module bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Import, Standard Library

- Thực hiện và giải thích kết quả bài tập: Tách chương trình quản lý chi tiêu thành các hàm thêm, tổng hợp, hiển thị.
- Nhận diện và tránh sai lầm: Viết một hàm quá dài hoặc cài thư viện ngoài không cần thiết.

Kiến thức cần học

Function · Module · Import · Standard Library

Giải thích chi tiết

Vì sao cần học

Tách chương trình, import và tái sử dụng logic. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Function: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Module: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Import: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Standard Library: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Hàm, module và sử dụng thư viện ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Function, Module, Import, Standard Library.
3. Tách chương trình quản lý chi tiêu thành các hàm thêm, tổng hợp, hiển thị.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI đề xuất cấu trúc nhưng kiểm soát dependency và license.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Hàm, module và sử dụng thư viện" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Viết một hàm quá dài hoặc cài thư viện ngoài không cần thiết.

- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Function, Module, Import; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Hàm, module và sử dụng thư viện

```
def total_expenses(amounts):
    return sum(amounts)

if __name__ == "__main__":
    print(total_expenses([12000, 25000, 18000]))
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Function và Module bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Function, Module, Import, Standard Library
[ ] Đã làm: Tách chương trình quản lý chi tiêu thành các hàm thêm, tổng hợp, hiển thị.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Viết một hàm quá dài hoặc cài thư viện ngoài không cần thiết.
[ ] Đã tự kiểm chứng: Nhờ AI đề xuất cấu trúc nhưng kiểm soát dependency và license.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Python Tutorial Tài liệu
https://docs.python.org/3/tutorial/

Python venv Tài liệu
https://docs.python.org/3/library/venv.html

pytest documentation Tài liệu
https://docs.pytest.org/en/stable/

Nên học trước

Nhập xuất dữ liệu và kiểm tra đầu vào

---

4. Lỗi cú pháp, lỗi chạy và debug có hệ thống

Traceback, breakpoint, tái hiện lỗi và kiểm tra giả thuyết

Mục tiêu học tập

- Giải thích đúng Syntax Error và Runtime Error bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Logic Bug, Traceback, Breakpoint
- Thực hiện và giải thích kết quả bài tập: Cố ý tạo ba lỗi khác loại và ghi lại cách phát hiện, nguyên nhân, sửa lỗi.
- Nhận diện và tránh sai lầm: Sửa ngẫu nhiên nhiều chỗ cùng lúc rồi không biết nguyên nhân.

Kiến thức cần học

Syntax Error · Runtime Error · Logic Bug · Traceback · Breakpoint

Giải thích chi tiết

Vì sao cần học

Traceback, breakpoint, tái hiện lỗi và kiểm tra giả thuyết. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Syntax Error: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Runtime Error: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Logic Bug: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Traceback: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Breakpoint: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Lỗi cú pháp, lỗi chạy và debug có hệ thống ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Syntax Error, Runtime Error, Logic Bug, Traceback.
3. Cố ý tạo ba lỗi khác loại và ghi lại cách phát hiện, nguyên nhân, sửa lỗi.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Đưa AI lỗi tối thiểu có thể tái hiện, không gửi bí mật dự án.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Lỗi cú pháp, lỗi chạy và debug có hệ thống" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Sửa ngẫu nhiên nhiều chỗ cùng lúc rồi không biết nguyên nhân.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Syntax Error, Runtime Error, Logic Bug; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Lỗi cú pháp, lỗi chạy và debug có hệ thống

```
def divide(a, b):
    if b == 0:
        raise ValueError("Không thể chia cho 0")
    return a / b

print(divide(10, 2))
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Syntax Error và Runtime Error bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Syntax Error, Runtime Error, Logic Bug, Traceback

[ ] Đã làm: Cố ý tạo ba lỗi khác loại và ghi lại cách phát hiện, nguyên nhân, sửa lỗi.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Sửa ngẫu nhiên nhiều chỗ cùng lúc rồi không biết nguyên nhân.

[ ] Đã tự kiểm chứng: Đưa AI lỗi tối thiểu có thể tái hiện, không gửi bí mật dự án.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Python Tutorial Tài liệu
https://docs.python.org/3/tutorial/

Python venv Tài liệu
https://docs.python.org/3/library/venv.html

pytest documentation Tài liệu
https://docs.pytest.org/en/stable/

Nên học trước

Hàm, module và sử dụng thư viện

---

5. Kiểm thử chương trình và bộ ca thử

Happy path, boundary, invalid input, assertion và regression

Mục tiêu học tập

- Giải thích đúng Test Case và Assertion bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Boundary, Regression
- Thực hiện và giải thích kết quả bài tập: Viết ít nhất 6 test cho hàm tính điểm trung bình, gồm trường hợp biên và sai dữ liệu.
- Nhận diện và tránh sai lầm: Chỉ thử một đầu vào đúng rồi cho rằng chương trình hoàn chỉnh.

Kiến thức cần học

Test Case · Assertion · Boundary · Regression

Giải thích chi tiết

Vì sao cần học

Happy path, boundary, invalid input, assertion và regression. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Test Case: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Assertion: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Boundary: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Regression: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Kiểm thử chương trình và bộ ca thử ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Test Case, Assertion, Boundary, Regression.
3. Viết ít nhất 6 test cho hàm tính điểm trung bình, gồm trường hợp biên và sai dữ liệu.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI đề xuất test, tự kiểm tra kỳ vọng và chạy test thật.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Kiểm thử chương trình và bộ ca thử" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chỉ thử một đầu vào đúng rồi cho rằng chương trình hoàn chỉnh.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Test Case, Assertion, Boundary; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Kiểm thử chương trình và bộ ca thử

```
def average(numbers):
    if not numbers:
        raise ValueError("Danh sách rỗng")
    return sum(numbers) / len(numbers)

assert average([8, 10]) == 9
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

- [ ] Tự định nghĩa được Test Case và Assertion bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm Test Case, Assertion, Boundary, Regression
- [ ] Đã làm: Viết ít nhất 6 test cho hàm tính điểm trung bình, gồm trường hợp biên và sai dữ liệu.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Chỉ thử một đầu vào đúng rồi cho rằng chương trình hoàn chỉnh.
- [ ] Đã tự kiểm chứng: Dùng AI đề xuất test, tự kiểm tra kỳ vọng và chạy test thật.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Python Tutorial Tài liệu
https://docs.python.org/3/tutorial/

Python venv Tài liệu
https://docs.python.org/3/library/venv.html

pytest documentation Tài liệu
https://docs.pytest.org/en/stable/

Nên học trước

Lỗi cú pháp, lỗi chạy và debug có hệ thống

GIAI ĐOẠN 6

Dữ liệu và cơ sở dữ liệu

Hiểu dữ liệu có cấu trúc, mô hình quan hệ và truy vấn để không chỉ biết code mà còn biết dữ liệu đi đâu.

1. CSV, JSON và bảng dữ liệu

Phân biệt dữ liệu cấu trúc, schema, null, định dạng và kiểu

Mục tiêu học tập

- Giải thích đúng CSV và JSON bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Schema, Null, Record
- Thực hiện và giải thích kết quả bài tập: Lập dữ liệu 10 cuốn sách ở CSV và JSON, so sánh lợi/hạn chế.
- Nhận diện và tránh sai lầm: Nhầm chuỗi "10" với số 10 hoặc thiếu quy tắc dữ liệu trống.

Kiến thức cần học

CSV · JSON · Schema · Null · Record

Giải thích chi tiết

Vì sao cần học

Phân biệt dữ liệu cấu trúc, schema, null, định dạng và kiểu. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- CSV: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- JSON: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Schema: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Null: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Record: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức CSV, JSON và bảng dữ liệu ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: CSV, JSON, Schema, Null.
3. Lập dữ liệu 10 cuốn sách ở CSV và JSON, so sánh lợi/hạn chế.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không tải dữ liệu cá nhân thật vào AI công khai.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "CSV, JSON và bảng dữ liệu" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm chuỗi "10" với số 10 hoặc thiếu quy tắc dữ liệu trống.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được CSV, JSON, Schema; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: CSV, JSON và bảng dữ liệu

```
{"books": [{"id": 1, "title": "Nhập môn CNTT", "available": true}]}
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được CSV và JSON bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm CSV, JSON, Schema, Null
[ ] Đã làm: Lập dữ liệu 10 cuốn sách ở CSV và JSON, so sánh lợi/hạn chế.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Nhầm chuỗi "10" với số 10 hoặc thiếu quy tắc dữ liệu trống.
[ ] Đã tự kiểm chứng: Không tải dữ liệu cá nhân thật vào AI công khai.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

SQLite Documentation Tài liệu
https://www.sqlite.org/docs.html

SQLBolt Khóa học
https://sqlbolt.com/

Python json documentation Tài liệu
https://docs.python.org/3/library/json.html

Nên học trước

Kiểm thử chương trình và bộ ca thử

2. Cơ sở dữ liệu quan hệ và khóa

Table, row, column, primary key, foreign key

Mục tiêu học tập

- Giải thích đúng Database và Table bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Primary Key, Foreign Key, Relation
- Thực hiện và giải thích kết quả bài tập: Vẽ mô hình SinhVien–MonHoc–DangKy có khóa chính và khóa ngoại.
- Nhận diện và tránh sai lầm: Lưu một ô chứa danh sách ID phân tách bằng dấu phẩy.

Kiến thức cần học

Database · Table · Primary Key · Foreign Key · Relation

Giải thích chi tiết

Vì sao cần học

Table, row, column, primary key, foreign key. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Database: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Table: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Primary Key: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Foreign Key: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Relation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Cơ sở dữ liệu quan hệ và khóa ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Database, Table, Primary Key, Foreign Key.
3. Vẽ mô hình SinhVien–MonHoc–DangKy có khóa chính và khóa ngoại.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI soát mô hình rồi tự chứng minh quan hệ dữ liệu.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Cơ sở dữ liệu quan hệ và khóa" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Lưu một ô chứa danh sách ID phân tách bằng dấu phẩy.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Database, Table, Primary Key; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Cơ sở dữ liệu quan hệ và khóa
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Database và Table bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Database, Table, Primary Key, Foreign Key
[ ] Đã làm: Vẽ mô hình SinhVien–MonHoc–DangKy có khóa chính và khóa ngoại.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Lưu một ô chứa danh sách ID phân tách bằng dấu phẩy.
[ ] Đã tự kiểm chứng: Nhờ AI soát mô hình rồi tự chứng minh quan hệ dữ liệu.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

SQLite Documentation Tài liệu
https://www.sqlite.org/docs.html

SQLBolt Khóa học
https://sqlbolt.com/

Python json documentation Tài liệu
https://docs.python.org/3/library/json.html

Nên học trước

CSV, JSON và bảng dữ liệu

3. SQL nền tảng: SELECT, WHERE, JOIN

Truy vấn, lọc, sắp xếp, kết nối bảng và giới hạn kết quả

Mục tiêu học tập

- Giải thích đúng SELECT và WHERE bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: ORDER BY, JOIN, GROUP BY
- Thực hiện và giải thích kết quả bài tập: Chạy SQLite trên dữ liệu mẫu; viết 5 câu truy vấn và giải thích kết quả.
- Nhận diện và tránh sai lầm: Dùng SELECT * bất kể nhu cầu hoặc JOIN sai làm nhân dòng.

Kiến thức cần học

SELECT · WHERE · ORDER BY · JOIN · GROUP BY

Giải thích chi tiết

Vì sao cần học

Truy vấn, lọc, sắp xếp, kết nối bảng và giới hạn kết quả. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- SELECT: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- WHERE: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- ORDER BY: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- JOIN: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- GROUP BY: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức SQL nền tảng: SELECT, WHERE, JOIN ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: SELECT, WHERE, ORDER BY, JOIN.
3. Chạy SQLite trên dữ liệu mẫu; viết 5 câu truy vấn và giải thích kết quả.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Đối chiếu kết quả với dữ liệu gốc; không chạy SQL phá hủy do AI đề xuất.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "SQL nền tảng: SELECT, WHERE, JOIN" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Dùng SELECT * bất kể nhu cầu hoặc JOIN sai làm nhân dòng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được SELECT, WHERE, ORDER BY; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: SQL nền tảng: SELECT, WHERE, JOIN

```
SELECT student_name, COUNT(*) AS course_count
FROM enrollments
GROUP BY student_name
ORDER BY course_count DESC;
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được SELECT và WHERE bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm SELECT, WHERE, ORDER BY, JOIN

[ ] Đã làm: Chạy SQLite trên dữ liệu mẫu; viết 5 câu truy vấn và giải thích kết quả.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Dùng SELECT * bất kể nhu cầu hoặc JOIN sai làm nhân dòng.

[ ] Đã tự kiểm chứng: Đối chiếu kết quả với dữ liệu gốc; không chạy SQL phá hủy do AI đề xuất.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

SQLite Documentation Tài liệu
https://www.sqlite.org/docs.html

SQLBolt Khóa học
https://sqlbolt.com/

Python json documentation Tài liệu
https://docs.python.org/3/library/json.html

Nên học trước

Cơ sở dữ liệu quan hệ và khóa

4. Chất lượng dữ liệu và quyền riêng tư

Missing values, trùng lặp, nguồn gốc, tính nhất quán và vòng đời dữ liệu

Mục tiêu học tập

- Giải thích đúng Data Quality và Deduplication bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Lineage, Personal Data
- Thực hiện và giải thích kết quả bài tập: Kiểm tra một bảng mẫu có 3 lỗi trùng, trống, sai định dạng; đề xuất cách xử lý.
- Nhận diện và tránh sai lầm: Tự điền dữ liệu thiếu theo suy đoán không có bằng chứng.

Kiến thức cần học

Data Quality · Deduplication · Lineage · Personal Data

Giải thích chi tiết

Vì sao cần học

Missing values, trùng lặp, nguồn gốc, tính nhất quán và vòng đời dữ liệu. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Data Quality: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Deduplication: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Lineage: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Personal Data: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Chất lượng dữ liệu và quyền riêng tư ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Data Quality, Deduplication, Lineage, Personal Data.
3. Kiểm tra một bảng mẫu có 3 lỗi trùng, trống, sai định dạng; đề xuất cách xử lý.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Ẩn danh dữ liệu khi hỏi AI và ghi rõ phần suy luận.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Chất lượng dữ liệu và quyền riêng tư" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tự điền dữ liệu thiếu theo suy đoán không có bằng chứng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Data Quality, Deduplication, Lineage; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Chất lượng dữ liệu và quyền riêng tư
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Data Quality và Deduplication bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Data Quality, Deduplication, Lineage, Personal Data
[ ] Đã làm: Kiểm tra một bảng mẫu có 3 lỗi trùng, trống, sai định dạng; đề xuất cách xử lý.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Tự điền dữ liệu thiếu theo suy đoán không có bằng chứng.
[ ] Đã tự kiểm chứng: Ẩn danh dữ liệu khi hỏi AI và ghi rõ phần suy luận.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

SQLite Documentation Tài liệu
https://www.sqlite.org/docs.html

SQLBolt Khóa học
https://sqlbolt.com/

Python json documentation Tài liệu
https://docs.python.org/3/library/json.html

Nên học trước

SQL nền tảng: SELECT, WHERE, JOIN

---

5. Phân tích dữ liệu và biểu đồ cơ bản

Trung bình, trung vị, phân nhóm, tương quan và diễn giải đúng

Mục tiêu học tập

- Giải thích đúng Average và Median bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Distribution, Correlation, Visualization

- Thực hiện và giải thích kết quả bài tập: Dùng bảng tính tạo biểu đồ số lượt mượn sách theo tháng và viết 3 nhận xét có căn cứ.
- Nhận diện và tránh sai lầm: Diễn giải tương quan thành quan hệ nhân quả.

Kiến thức cần học

Average · Median · Distribution · Correlation · Visualization

Giải thích chi tiết

Vì sao cần học

Trung bình, trung vị, phân nhóm, tương quan và diễn giải đúng. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Average: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Median: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Distribution: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Correlation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Visualization: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Phân tích dữ liệu và biểu đồ cơ bản ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Average, Median, Distribution, Correlation.
3. Dùng bảng tính tạo biểu đồ số lượt mượn sách theo tháng và viết 3 nhận xét có căn cứ.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Yêu cầu AI chỉ ra cách diễn giải sai có thể xảy ra, kiểm tra bằng số liệu.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Phân tích dữ liệu và biểu đồ cơ bản" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Diễn giải tương quan thành quan hệ nhân quả.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Average, Median, Distribution; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Phân tích dữ liệu và biểu đồ cơ bản
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Average và Median bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Average, Median, Distribution, Correlation
[ ] Đã làm: Dùng bảng tính tạo biểu đồ số lượt mượn sách theo tháng và viết 3 nhận xét có căn cứ.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Diễn giải tương quan thành quan hệ nhân quả.
[ ] Đã tự kiểm chứng: Yêu cầu AI chỉ ra cách diễn giải sai có thể xảy ra, kiểm tra bằng số liệu.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

SQLite Documentation Tài liệu
https://www.sqlite.org/docs.html

SQLBolt Khóa học
https://sqlbolt.com/

Python json documentation Tài liệu
https://docs.python.org/3/library/json.html

Nên học trước

Chất lượng dữ liệu và quyền riêng tư

GIAI ĐOẠN 7

Xây dựng phần mềm và cộng tác

Tập trung vào quy trình phát triển sản phẩm: yêu cầu, thiết kế, Git, kiểm thử, triển khai.

1. Yêu cầu phần mềm và user story

Từ nỗi đau người dùng đến tiêu chí chấp nhận có thể đo

Mục tiêu học tập

- Giải thích đúng Requirement và User Story bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Acceptance Criteria, Scope
- Thực hiện và giải thích kết quả bài tập: Viết 5 user stories cho ứng dụng quản lý việc học, mỗi câu có điều kiện đạt.
- Nhận diện và tránh sai lầm: Yêu cầu như "ứng dụng hiện đại, đẹp" không thể kiểm thử.

Kiến thức cần học

Requirement · User Story · Acceptance Criteria · Scope

Giải thích chi tiết

Vì sao cần học

Từ nỗi đau người dùng đến tiêu chí chấp nhận có thể đo. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Requirement: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- User Story: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Acceptance Criteria: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Scope: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Yêu cầu phần mềm và user story ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Requirement, User Story, Acceptance Criteria, Scope.
3. Viết 5 user stories cho ứng dụng quản lý việc học, mỗi câu có điều kiện đạt.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Cho AI tìm yêu cầu mơ hồ, tự quyết định phạm vi tối thiểu.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Yêu cầu phần mềm và user story" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Yêu cầu như "ứng dụng hiện đại, đẹp" không thể kiểm thử.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Requirement, User Story, Acceptance Criteria; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Yêu cầu phần mềm và user story
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Requirement và User Story bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Requirement, User Story, Acceptance Criteria, Scope

[ ] Đã làm: Viết 5 user stories cho ứng dụng quản lý việc học, mỗi câu có điều kiện đạt.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Yêu cầu như "ứng dụng hiện đại, đẹp" không thể kiểm thử.

[ ] Đã tự kiểm chứng: Cho AI tìm yêu cầu mơ hồ, tự quyết định phạm vi tối thiểu.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Pro Git Book Sách
https://git-scm.com/book/en/v2

GitHub Skills Khóa học
https://skills.github.com/

MDN: Accessibility Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility

Nên học trước

Phân tích dữ liệu và biểu đồ cơ bản

2. Giao diện và trải nghiệm người dùng

Wireframe, luồng thao tác, phản hồi, khả năng tiếp cận

Mục tiêu học tập

- Giải thích đúng UI và UX bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Wireframe, User Flow, Accessibility
- Thực hiện và giải thích kết quả bài tập: Vẽ tay hoặc dùng Figma phác ba màn hình ứng dụng học tập và thử với một người dùng.
- Nhận diện và tránh sai lầm: Đồng nhất UI đẹp với UX dễ sử dụng.

Kiến thức cần học

UI · UX · Wireframe · User Flow · Accessibility

Giải thích chi tiết

Vì sao cần học

Wireframe, luồng thao tác, phản hồi, khả năng tiếp cận. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- UI: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- UX: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Wireframe: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- User Flow: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Accessibility: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Giao diện và trải nghiệm người dùng ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: UI, UX, Wireframe, User Flow.
3. Vẽ tay hoặc dùng Figma phác ba màn hình ứng dụng học tập và thử với một người dùng.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI đề xuất cách viết nhãn rõ nghĩa, thử với người thật.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Giao diện và trải nghiệm người dùng" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đồng nhất UI đẹp với UX dễ sử dụng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được UI, UX, Wireframe; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Giao diện và trải nghiệm người dùng
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được UI và UX bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm UI, UX, Wireframe, User Flow

[ ] Đã làm: Vẽ tay hoặc dùng Figma phác ba màn hình ứng dụng học tập và thử với một người dùng.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Đồng nhất UI đẹp với UX dễ sử dụng.

[ ] Đã tự kiểm chứng: Dùng AI đề xuất cách viết nhãn rõ nghĩa, thử với người thật.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Pro Git Book Sách
https://git-scm.com/book/en/v2

GitHub Skills Khóa học
https://skills.github.com/

MDN: Accessibility Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility

Nên học trước

Yêu cầu phần mềm và user story

3. Git và quản lý phiên bản

Repository, commit, branch, diff và lưu lịch sử thay đổi

Mục tiêu học tập

- Giải thích đúng Git và Repository bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Commit, Branch, Diff
- Thực hiện và giải thích kết quả bài tập: Tạo repo cho dự án nhỏ, làm ba commit có ý nghĩa và kiểm tra diff trước mỗi commit.
- Nhận diện và tránh sai lầm: Nhầm git commit với git push hoặc commit khóa bí mật.

Kiến thức cần học

Git · Repository · Commit · Branch · Diff

Giải thích chi tiết

Vì sao cần học

Repository, commit, branch, diff và lưu lịch sử thay đổi. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Git: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Repository: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Commit: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Branch: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Diff: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Git và quản lý phiên bản ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Git, Repository, Commit, Branch.
3. Tạo repo cho dự án nhỏ, làm ba commit có ý nghĩa và kiểm tra diff trước mỗi commit.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Luôn xem diff trước khi chấp nhận code AI sinh.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Git và quản lý phiên bản" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nhầm git commit với git push hoặc commit khóa bí mật.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Git, Repository, Commit; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Git và quản lý phiên bản

```
git init
git status
git add README.md
git diff --staged
git commit -m "docs: add introduction"
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Git và Repository bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Git, Repository, Commit, Branch

[ ] Đã làm: Tạo repo cho dự án nhỏ, làm ba commit có ý nghĩa và kiểm tra diff trước mỗi commit.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Nhầm git commit với git push hoặc commit khóa bí mật.

[ ] Đã tự kiểm chứng: Luôn xem diff trước khi chấp nhận code AI sinh.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Pro Git Book Sách
https://git-scm.com/book/en/v2

GitHub Skills Khóa học
https://skills.github.com/

MDN: Accessibility Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility

Nên học trước

Giao diện và trải nghiệm người dùng

4. Quy trình phát triển phần mềm

Plan, build, test, release, maintain; vấn đề và phản hồi

Mục tiêu học tập

- Giải thích đúng SDLC và Issue bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Pull Request, Code Review, CI
- Thực hiện và giải thích kết quả bài tập: Lập bảng tiến độ 2 tuần cho ứng dụng nhỏ, chia việc thành issue và điều kiện nghiệm thu.
- Nhận diện và tránh sai lầm: Bỏ qua kiểm thử và bảo trì vì tính năng đã chạy.

Kiến thức cần học

SDLC · Issue · Pull Request · Code Review · CI

Giải thích chi tiết

Vì sao cần học

Plan, build, test, release, maintain; vấn đề và phản hồi. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- SDLC: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Issue: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Pull Request: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Code Review: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- CI: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Quy trình phát triển phần mềm ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: SDLC, Issue, Pull Request, Code Review.
3. Lập bảng tiến độ 2 tuần cho ứng dụng nhỏ, chia việc thành issue và điều kiện nghiệm thu.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không để agent tự merge hoặc deploy production khi chưa duyệt.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Quy trình phát triển phần mềm" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Bỏ qua kiểm thử và bảo trì vì tính năng đã chạy.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được SDLC, Issue, Pull Request; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Quy trình phát triển phần mềm
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được SDLC và Issue bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm SDLC, Issue, Pull Request, Code Review

[ ] Đã làm: Lập bảng tiến độ 2 tuần cho ứng dụng nhỏ, chia việc thành issue và điều kiện nghiệm thu.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Bỏ qua kiểm thử và bảo trì vì tính năng đã chạy.

[ ] Đã tự kiểm chứng: Không để agent tự merge hoặc deploy production khi chưa duyệt.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Pro Git Book Sách
https://git-scm.com/book/en/v2

GitHub Skills Khóa học
https://skills.github.com/

MDN: Accessibility Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility

Nên học trước

Git và quản lý phiên bản

5. Triển khai, cấu hình và theo dõi ứng dụng

Local vs production, môi trường, log, backup và sự cố

Mục tiêu học tập

- Giải thích đúng Deployment và Environment Variable bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Logging, Monitoring, Backup
- Thực hiện và giải thích kết quả bài tập: Xuất bản trang HTML tĩnh miễn phí và ghi checklist kiểm tra sau triển khai.
- Nhận diện và tránh sai lầm: Đẩy file .env chứa khóa lên repository công khai.

Kiến thức cần học

Deployment · Environment Variable · Logging · Monitoring · Backup

Giải thích chi tiết

Vì sao cần học

Local vs production, môi trường, log, backup và sự cố. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Deployment: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Environment Variable: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Logging: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Monitoring: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Backup: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Triển khai, cấu hình và theo dõi ứng dụng ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Deployment, Environment Variable, Logging, Monitoring.
3. Xuất bản trang HTML tĩnh miễn phí và ghi checklist kiểm tra sau triển khai.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Che secrets khi đưa log cho AI và không tự động công khai dịch vụ nhạy cảm.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Triển khai, cấu hình và theo dõi ứng dụng" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đẩy file .env chứa khóa lên repository công khai.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Deployment, Environment Variable, Logging; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Triển khai, cấu hình và theo dõi ứng dụng
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Deployment và Environment Variable bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Deployment, Environment Variable, Logging, Monitoring

[ ] Đã làm: Xuất bản trang HTML tĩnh miễn phí và ghi checklist kiểm tra sau triển khai.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Đẩy file .env chứa khóa lên repository công khai.

[ ] Đã tự kiểm chứng: Che secrets khi đưa log cho AI và không tự động công khai dịch vụ nhạy cảm.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Pro Git Book Sách
https://git-scm.com/book/en/v2

GitHub Skills Khóa học
https://skills.github.com/

MDN: Accessibility Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility

Nên học trước

Quy trình phát triển phần mềm

GIAI ĐOẠN 8

Hệ thống, bảo mật và điện toán đám mây

Hiểu hệ thống và rủi ro ở mức nhập môn, không biến thành khóa chuyên sâu về bảo mật.

1. Nguyên tắc an toàn thông tin

CIA triad, quyền tối thiểu, cập nhật, sao lưu và quản lý bí mật

Mục tiêu học tập

- Giải thích đúng Confidentiality và Integrity bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Availability, Least Privilege, Secrets
- Thực hiện và giải thích kết quả bài tập: Phân loại dữ liệu của ứng dụng học tập; chọn biện pháp bảo vệ phù hợp từng loại.
- Nhận diện và tránh sai lầm: Dùng chung mật khẩu, hardcode API key hoặc cấp quyền admin không cần thiết.

Kiến thức cần học

Confidentiality · Integrity · Availability · Least Privilege · Secrets

Giải thích chi tiết

Vì sao cần học

CIA triad, quyền tối thiểu, cập nhật, sao lưu và quản lý bí mật. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Confidentiality: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Integrity: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Availability: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Least Privilege: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Secrets: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Nguyên tắc an toàn thông tin ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Confidentiality, Integrity, Availability, Least Privilege.
3. Phân loại dữ liệu của ứng dụng học tập; chọn biện pháp bảo vệ phù hợp từng loại.

4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không gửi mã truy cập, khóa riêng hay dữ liệu cá nhân cho AI.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Nguyên tắc an toàn thông tin" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Dùng chung mật khẩu, hardcode API key hoặc cấp quyền admin không cần thiết.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Confidentiality, Integrity, Availability; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Nguyên tắc an toàn thông tin
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Confidentiality và Integrity bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Confidentiality, Integrity, Availability, Least Privilege
[ ] Đã làm: Phân loại dữ liệu của ứng dụng học tập; chọn biện pháp bảo vệ phù hợp từng loại.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Dùng chung mật khẩu, hardcode API key hoặc cấp quyền admin không cần thiết.
[ ] Đã tự kiểm chứng: Không gửi mã truy cập, khóa riêng hay dữ liệu cá nhân cho AI.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

OWASP Top 10 Tài liệu
https://owasp.org/www-project-top-ten/

CISA Secure Our World Website
https://www.cisa.gov/secure-our-world

Microsoft Learn: Cloud concepts Khóa học
https://learn.microsoft.com/en-us/training/paths/describe-cloud-compute/

Nên học trước

Triển khai, cấu hình và theo dõi ứng dụng

---

2. Đăng nhập, xác thực và phân quyền

Authentication vs authorization, session, MFA, vai trò

Mục tiêu học tập

- Giải thích đúng Authentication và Authorization bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Session, RBAC, MFA
- Thực hiện và giải thích kết quả bài tập: Lập bảng quyền student/teacher/admin cho các hành động đọc, sửa, xóa.
- Nhận diện và tránh sai lầm: Tưởng đã đăng nhập thì được xem mọi dữ liệu.

Kiến thức cần học

Authentication · Authorization · Session · RBAC · MFA

Giải thích chi tiết

Vì sao cần học

Authentication vs authorization, session, MFA, vai trò. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Authentication: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Authorization: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Session: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- RBAC: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- MFA: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Đăng nhập, xác thực và phân quyền ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Authentication, Authorization, Session, RBAC.
3. Lập bảng quyền student/teacher/admin cho các hành động đọc, sửa, xóa.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI rà soát bảng quyền nhưng tự kiểm tra từng luồng truy cập.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Đăng nhập, xác thực và phân quyền" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tưởng đã đăng nhập thì được xem mọi dữ liệu.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Authentication, Authorization, Session; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Đăng nhập, xác thực và phân quyền
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Authentication và Authorization bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Authentication, Authorization, Session, RBAC
[ ] Đã làm: Lập bảng quyền student/teacher/admin cho các hành động đọc, sửa, xóa.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Tưởng đã đăng nhập thì được xem mọi dữ liệu.
[ ] Đã tự kiểm chứng: Dùng AI rà soát bảng quyền nhưng tự kiểm tra từng luồng truy cập.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

OWASP Top 10 Tài liệu
https://owasp.org/www-project-top-ten/

CISA Secure Our World Website
https://www.cisa.gov/secure-our-world

Microsoft Learn: Cloud concepts Khóa học
https://learn.microsoft.com/en-us/training/paths/describe-cloud-compute/

Nên học trước

Nguyên tắc an toàn thông tin

3. Linux và tiến trình dịch vụ

File permissions, package manager, process và service

Mục tiêu học tập

- Giải thích đúng Linux và chmod bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Package Manager, Process, Service
- Thực hiện và giải thích kết quả bài tập: Trong máy ảo/thùng chứa thử lệnh ls -l, ps, tìm hiểu quyền đọc/ghi/chạy.
- Nhận diện và tránh sai lầm: Dùng chmod 777 như cách sửa mọi lỗi quyền.

Kiến thức cần học

Linux · chmod · Package Manager · Process · Service

Giải thích chi tiết

Vì sao cần học

File permissions, package manager, process và service. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Linux: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- chmod: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Package Manager: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Process: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Service: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Linux và tiến trình dịch vụ ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Linux, chmod, Package Manager, Process.
3. Trong máy ảo/thùng chứa thử lệnh ls -l, ps, tìm hiểu quyền đọc/ghi/chạy.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không chạy sudo hoặc lệnh sửa hệ thống từ nguồn chưa tin cậy.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Linux và tiến trình dịch vụ" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Dùng chmod 777 như cách sửa mọi lỗi quyền.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Linux, chmod, Package Manager; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Linux và tiến trình dịch vụ

```
pwd
ls -l
ps
whoami
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Linux và chmod bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Linux, chmod, Package Manager, Process
[ ] Đã làm: Trong máy ảo/thùng chứa thử lệnh ls -l, ps, tìm hiểu quyền đọc/ghi/chạy.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Dùng chmod 777 như cách sửa mọi lỗi quyền.
[ ] Đã tự kiểm chứng: Không chạy sudo hoặc lệnh sửa hệ thống từ nguồn chưa tin cậy.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

OWASP Top 10 Tài liệu
https://owasp.org/www-project-top-ten/

CISA Secure Our World Website
https://www.cisa.gov/secure-our-world

Microsoft Learn: Cloud concepts Khóa học
https://learn.microsoft.com/en-us/training/paths/describe-cloud-compute/

Nên học trước

Đăng nhập, xác thực và phân quyền

---

4. Cloud, hosting và các mô hình dịch vụ

IaaS, PaaS, SaaS, region, scale, cost, shared responsibility

Mục tiêu học tập

- Giải thích đúng Cloud và IaaS bằng ngôn ngữ phổ thông

- Phân biệt được các khái niệm liên quan: PaaS, SaaS, Region, Billing
- Thực hiện và giải thích kết quả bài tập: So sánh triển khai một trang tĩnh và một ứng dụng có database theo trách nhiệm, chi phí.
- Nhận diện và tránh sai lầm: Nghĩ cloud tự động miễn phí hoặc nhà cung cấp chịu toàn bộ bảo mật.

Kiến thức cần học

Cloud · IaaS · PaaS · SaaS · Region · Billing

Giải thích chi tiết

Vì sao cần học

IaaS, PaaS, SaaS, region, scale, cost, shared responsibility. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Cloud: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- IaaS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- PaaS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- SaaS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Region: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Billing: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Cloud, hosting và các mô hình dịch vụ ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Cloud, IaaS, PaaS, SaaS.
3. So sánh triển khai một trang tĩnh và một ứng dụng có database theo trách nhiệm, chi phí.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Kiểm tra giới hạn và cảnh báo chi phí trước khi dùng dịch vụ.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Cloud, hosting và các mô hình dịch vụ" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Nghĩ cloud tự động miễn phí hoặc nhà cung cấp chịu toàn bộ bảo mật.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Cloud, IaaS, PaaS; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Cloud, hosting và các mô hình dịch vụ
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Cloud và IaaS bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Cloud, IaaS, PaaS, SaaS
[ ] Đã làm: So sánh triển khai một trang tĩnh và một ứng dụng có database theo trách nhiệm, chi phí.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Nghĩ cloud tự động miễn phí hoặc nhà cung cấp chịu toàn bộ bảo mật.
[ ] Đã tự kiểm chứng: Kiểm tra giới hạn và cảnh báo chi phí trước khi dùng dịch vụ.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

OWASP Top 10 Tài liệu
https://owasp.org/www-project-top-ten/

CISA Secure Our World Website
https://www.cisa.gov/secure-our-world

Microsoft Learn: Cloud concepts Khóa học
https://learn.microsoft.com/en-us/training/paths/describe-cloud-compute/

Nên học trước

Linux và tiến trình dịch vụ

---

5. Bảo vệ API và dữ liệu ứng dụng

Validate input, HTTPS, lỗi lộ dữ liệu, rate limit và logging

Mục tiêu học tập

- Giải thích đúng Input Validation và HTTPS bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Rate Limiting, API Key, Logging

- Thực hiện và giải thích kết quả bài tập: Thiết kế checklist cho API đăng ký tài khoản: đầu vào, quyền, lỗi, giới hạn yêu cầu.
- Nhận diện và tránh sai lầm: Tin mọi dữ liệu do trình duyệt gửi lên và trả lỗi chứa thông tin nội bộ.

Kiến thức cần học

Input Validation · HTTPS · Rate Limiting · API Key · Logging

Giải thích chi tiết

Vì sao cần học

Validate input, HTTPS, lỗi lộ dữ liệu, rate limit và logging. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Input Validation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- HTTPS: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Rate Limiting: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- API Key: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Logging: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Bảo vệ API và dữ liệu ứng dụng ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Input Validation, HTTPS, Rate Limiting, API Key.
3. Thiết kế checklist cho API đăng ký tài khoản: đầu vào, quyền, lỗi, giới hạn yêu cầu.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng ví dụ giả lập; không kiểm thử xâm nhập hệ thống không được phép.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Bảo vệ API và dữ liệu ứng dụng" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tin mọi dữ liệu do trình duyệt gửi lên và trả lỗi chứa thông tin nội bộ.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Input Validation, HTTPS, Rate Limiting; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Bảo vệ API và dữ liệu ứng dụng
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Input Validation và HTTPS bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Input Validation, HTTPS, Rate Limiting, API Key
[ ] Đã làm: Thiết kế checklist cho API đăng ký tài khoản: đầu vào, quyền, lỗi, giới hạn yêu cầu.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Tin mọi dữ liệu do trình duyệt gửi lên và trả lỗi chứa thông tin nội bộ.
[ ] Đã tự kiểm chứng: Dùng ví dụ giả lập; không kiểm thử xâm nhập hệ thống không được phép.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

OWASP Top 10 Tài liệu
https://owasp.org/www-project-top-ten/

CISA Secure Our World Website
https://www.cisa.gov/secure-our-world

Microsoft Learn: Cloud concepts Khóa học
https://learn.microsoft.com/en-us/training/paths/describe-cloud-compute/

Nên học trước

Cloud, hosting và các mô hình dịch vụ

GIAI ĐOẠN 9

AI và tư duy công nghệ trong thời đại mới

Biết dùng AI để học và làm việc hiệu quả nhưng luôn kiểm chứng, bảo mật và tự chịu trách nhiệm.

1. AI, Machine Learning và LLM

Phân biệt các khái niệm, năng lực, giới hạn, xác suất và lỗi bịa

Mục tiêu học tập

- Giải thích đúng Artificial Intelligence và Machine Learning bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: LLM, Token, Hallucination
- Thực hiện và giải thích kết quả bài tập: Chọn ba tác vụ và giải thích khi nào cần quy tắc, thuật toán, ML hay LLM.
- Nhận diện và tránh sai lầm: Cho rằng AI hiểu và luôn biết sự thật như con người.

Kiến thức cần học

Artificial Intelligence · Machine Learning · LLM · Token · Hallucination

Giải thích chi tiết

Vì sao cần học

Phân biệt các khái niệm, năng lực, giới hạn, xác suất và lỗi bịa. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Artificial Intelligence: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Machine Learning: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- LLM: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Token: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Hallucination: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức AI, Machine Learning và LLM ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Artificial Intelligence, Machine Learning, LLM, Token.
3. Chọn ba tác vụ và giải thích khi nào cần quy tắc, thuật toán, ML hay LLM.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.

5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Đối chiếu câu trả lời với nguồn và phép thử độc lập.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "AI, Machine Learning và LLM" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cho rằng AI hiểu và luôn biết sự thật như con người.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Artificial Intelligence, Machine Learning, LLM; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: AI, Machine Learning và LLM
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Artificial Intelligence và Machine Learning bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Artificial Intelligence, Machine Learning, LLM, Token
[ ] Đã làm: Chọn ba tác vụ và giải thích khi nào cần quy tắc, thuật toán, ML hay LLM.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Cho rằng AI hiểu và luôn biết sự thật như con người.
[ ] Đã tự kiểm chứng: Đối chiếu câu trả lời với nguồn và phép thử độc lập.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Google Machine Learning Crash Course Khóa học
https://developers.google.com/machine-learning/crash-course

OpenAI prompting guide Tài liệu
https://platform.openai.com/docs/guides/prompt-engineering

OWASP GenAI Security Tài liệu
https://genai.owasp.org/

Nên học trước

Bảo vệ API và dữ liệu ứng dụng

2. Viết yêu cầu rõ ràng cho AI

Mục tiêu, ngữ cảnh, dữ liệu, ràng buộc, đầu ra và ví dụ

Mục tiêu học tập

- Giải thích đúng Prompt và Context bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Constraint, Few-shot, Structured Output
- Thực hiện và giải thích kết quả bài tập: Viết prompt yêu cầu tổng hợp thông báo lớp; chạy với và không có ví dụ, so kết quả.
- Nhận diện và tránh sai lầm: Viết prompt dài nhưng thiếu tiêu chí hoàn thành.

Kiến thức cần học

Prompt · Context · Constraint · Few-shot · Structured Output

Giải thích chi tiết

Vì sao cần học

Mục tiêu, ngữ cảnh, dữ liệu, ràng buộc, đầu ra và ví dụ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Prompt: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Context: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Constraint: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Few-shot: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Structured Output: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Viết yêu cầu rõ ràng cho AI ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Prompt, Context, Constraint, Few-shot.
3. Viết prompt yêu cầu tổng hợp thông báo lớp; chạy với và không có ví dụ, so kết quả.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Không đưa vào prompt dữ liệu không có quyền chia sẻ.

- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Viết yêu cầu rõ ràng cho AI" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Viết prompt dài nhưng thiếu tiêu chí hoàn thành.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Prompt, Context, Constraint; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Viết yêu cầu rõ ràng cho AI

```
Mục tiêu: Tóm tắt thông báo lớp.
Dữ liệu: [nội dung được phép chia sẻ].
Yêu cầu: 3 gạch đầu dòng, giữ nguyên ngày và địa điểm.
Nếu thiếu thông tin, ghi "chưa được cung cấp".
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Prompt và Context bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Prompt, Context, Constraint, Few-shot
[ ] Đã làm: Viết prompt yêu cầu tổng hợp thông báo lớp; chạy với và không có ví dụ, so kết quả.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Viết prompt dài nhưng thiếu tiêu chí hoàn thành.
[ ] Đã tự kiểm chứng: Không đưa vào prompt dữ liệu không có quyền chia sẻ.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Google Machine Learning Crash Course Khóa học
https://developers.google.com/machine-learning/crash-course

OpenAI prompting guide Tài liệu
https://platform.openai.com/docs/guides/prompt-engineering

OWASP GenAI Security Tài liệu
https://genai.owasp.org/

Nên học trước

AI, Machine Learning và LLM

---

3. AI hỗ trợ học và lập trình

Giải thích mã, gợi ý test, review diff nhưng giữ quyền quyết định

Mục tiêu học tập

- Giải thích đúng AI Coding Assistant và Code Review bằng ngôn ngữ phổ thông

- Phân biệt được các khái niệm liên quan: Test, Diff, Human Oversight
- Thực hiện và giải thích kết quả bài tập: Tự viết hàm nhỏ, nhờ AI góp ý, thử ít nhất ba ca biên trước khi sửa.
- Nhận diện và tránh sai lầm: Chép toàn bộ code AI mà không đọc, không hiểu, không chạy test.

Kiến thức cần học

AI Coding Assistant · Code Review · Test · Diff · Human Oversight

Giải thích chi tiết

Vì sao cần học

Giải thích mã, gợi ý test, review diff nhưng giữ quyền quyết định. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- AI Coding Assistant: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Code Review: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Test: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Diff: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Human Oversight: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức AI hỗ trợ học và lập trình ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: AI Coding Assistant, Code Review, Test, Diff.
3. Tự viết hàm nhỏ, nhờ AI góp ý, thử ít nhất ba ca biên trước khi sửa.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Mọi code do AI sinh phải có người đọc, kiểm thử và chấp thuận.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "AI hỗ trợ học và lập trình" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chép toàn bộ code AI mà không đọc, không hiểu, không chạy test.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được AI Coding Assistant, Code Review, Test; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: AI hỗ trợ học và lập trình
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được AI Coding Assistant và Code Review bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm AI Coding Assistant, Code Review, Test, Diff
[ ] Đã làm: Tự viết hàm nhỏ, nhờ AI góp ý, thử ít nhất ba ca biên trước khi sửa.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Chép toàn bộ code AI mà không đọc, không hiểu, không chạy test.
[ ] Đã tự kiểm chứng: Mọi code do AI sinh phải có người đọc, kiểm thử và chấp thuận.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Google Machine Learning Crash Course Khóa học
https://developers.google.com/machine-learning/crash-course

OpenAI prompting guide Tài liệu
https://platform.openai.com/docs/guides/prompt-engineering

OWASP GenAI Security Tài liệu
https://genai.owasp.org/

Nên học trước

Viết yêu cầu rõ ràng cho AI

---

4. Đánh giá AI, quyền riêng tư và prompt injection

Phát hiện câu trả lời sai, dữ liệu độc hại, thiên lệch và lộ bí mật

Mục tiêu học tập

- Giải thích đúng Evaluation và Grounding bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Prompt Injection, Privacy, Bias
- Thực hiện và giải thích kết quả bài tập: Tạo 5 câu hỏi có đáp án kiểm chứng; ghi tỷ lệ đúng và lỗi AI, thử dữ liệu chứa chỉ dẫn giả.
- Nhận diện và tránh sai lầm: Tưởng câu trả lời tự tin đồng nghĩa đúng; đưa quyền công cụ quá rộng.

Kiến thức cần học

Evaluation · Grounding · Prompt Injection · Privacy · Bias

Giải thích chi tiết

Vì sao cần học

Phát hiện câu trả lời sai, dữ liệu độc hại, thiên lệch và lộ bí mật. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Evaluation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Grounding: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Prompt Injection: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Privacy: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Bias: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Đánh giá AI, quyền riêng tư và prompt injection ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Evaluation, Grounding, Prompt Injection, Privacy.
3. Tạo 5 câu hỏi có đáp án kiểm chứng; ghi tỷ lệ đúng và lỗi AI, thử dữ liệu chứa chỉ dẫn giả.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Phân biệt dữ liệu bên ngoài với mệnh lệnh; duyệt các hành động nhạy cảm.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Đánh giá AI, quyền riêng tư và prompt injection" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Tưởng câu trả lời tự tin đồng nghĩa đúng; đưa quyền công cụ quá rộng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.

- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Evaluation, Grounding, Prompt Injection; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Đánh giá AI, quyền riêng tư và prompt injection

```
{"caseId": "case-01", "question": "...", "expected": "...", "actual": "...", "passed": false}
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Evaluation và Grounding bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Evaluation, Grounding, Prompt Injection, Privacy

[ ] Đã làm: Tạo 5 câu hỏi có đáp án kiểm chứng; ghi tỷ lệ đúng và lỗi AI, thử dữ liệu chứa chỉ dẫn giả.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Tưởng câu trả lời tự tin đồng nghĩa đúng; đưa quyền công cụ quá rộng.

[ ] Đã tự kiểm chứng: Phân biệt dữ liệu bên ngoài với mệnh lệnh; duyệt các hành động nhạy cảm.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Google Machine Learning Crash Course Khóa học
https://developers.google.com/machine-learning/crash-course

OpenAI prompting guide Tài liệu
https://platform.openai.com/docs/guides/prompt-engineering

OWASP GenAI Security Tài liệu
https://genai.owasp.org/

Nên học trước

AI hỗ trợ học và lập trình

---

5. Chọn nhánh học tiếp theo sau nhập môn

Tự đánh giá sở thích và năng lực qua sản phẩm nhỏ

Mục tiêu học tập

- Giải thích đúng Learning Path và Portfolio bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Specialization, Self-assessment
- Thực hiện và giải thích kết quả bài tập: Chọn một trong Web, Mobile, Data, AI, Security, Cloud; viết kế hoạch 6 tuần có sản phẩm đo được.
- Nhận diện và tránh sai lầm: Cố học mọi framework đồng thời trước khi có nền tảng.

Kiến thức cần học

Learning Path · Portfolio · Specialization · Self-assessment

Giải thích chi tiết

Vì sao cần học

Tự đánh giá sở thích và năng lực qua sản phẩm nhỏ. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Learning Path: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Portfolio: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Specialization: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Self-assessment: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Chọn nhánh học tiếp theo sau nhập môn ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Learning Path, Portfolio, Specialization, Self-assessment.
3. Chọn một trong Web, Mobile, Data, AI, Security, Cloud; viết kế hoạch 6 tuần có sản phẩm đo được.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI lập kế hoạch nháp, tự kiểm tra tính phù hợp và nguồn học.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Chọn nhánh học tiếp theo sau nhập môn" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Cố học mọi framework đồng thời trước khi có nền tảng.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Learning Path, Portfolio, Specialization; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Chọn nhánh học tiếp theo sau nhập môn
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được Learning Path và Portfolio bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm Learning Path, Portfolio, Specialization, Self-assessment
- [ ] Đã làm: Chọn một trong Web, Mobile, Data, AI, Security, Cloud; viết kế hoạch 6 tuần có sản phẩm đo được.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Cố học mọi framework đồng thời trước khi có nền tảng.
- [ ] Đã tự kiểm chứng: Dùng AI lập kế hoạch nháp, tự kiểm tra tính phù hợp và nguồn học.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

Google Machine Learning Crash Course Khóa học
https://developers.google.com/machine-learning/crash-course

OpenAI prompting guide Tài liệu
https://platform.openai.com/docs/guides/prompt-engineering

OWASP GenAI Security Tài liệu
https://genai.owasp.org/

Nên học trước

Đánh giá AI, quyền riêng tư và prompt injection

GIAI ĐOẠN 10

Dự án tổng kết và định hướng phát triển

Biến kiến thức thành minh chứng năng lực qua dự án cuối khóa và kế hoạch học tiếp.

1. Chọn đề tài và xác định MVP

Chọn bài toán vừa sức, nhóm người dùng, phạm vi và tiêu chí nghiệm thu

Mục tiêu học tập

- Giải thích đúng MVP và Scope bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: User Persona, Acceptance Criteria
- Thực hiện và giải thích kết quả bài tập: Chọn ứng dụng quản lý ghi chú, thư viện cá nhân hoặc chi tiêu mẫu; giới hạn tối đa 3 tính năng.
- Nhận diện và tránh sai lầm: Mở rộng quá nhiều tính năng trước khi hoàn thành phần cơ bản.

Kiến thức cần học

MVP · Scope · User Persona · Acceptance Criteria

Giải thích chi tiết

Vì sao cần học

Chọn bài toán vừa sức, nhóm người dùng, phạm vi và tiêu chí nghiệm thu. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- MVP: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Scope: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- User Persona: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Acceptance Criteria: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Chọn đề tài và xác định MVP ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: MVP, Scope, User Persona, Acceptance Criteria.
3. Chọn ứng dụng quản lý ghi chú, thư viện cá nhân hoặc chi tiêu mẫu; giới hạn tối đa 3 tính năng.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI đề xuất phạm vi nhưng tự khóa yêu cầu tối thiểu.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Chọn đề tài và xác định MVP" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Mở rộng quá nhiều tính năng trước khi hoàn thành phần cơ bản.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được MVP, Scope, User Persona; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Chọn đề tài và xác định MVP
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được MVP và Scope bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm MVP, Scope, User Persona, Acceptance Criteria
- [ ] Đã làm: Chọn ứng dụng quản lý ghi chú, thư viện cá nhân hoặc chi tiêu mẫu; giới hạn tối đa 3 tính năng.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Mở rộng quá nhiều tính năng trước khi hoàn thành phần cơ bản.
- [ ] Đã tự kiểm chứng: Nhờ AI đề xuất phạm vi nhưng tự khóa yêu cầu tối thiểu.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

GitHub README guidance Tài liệu
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

Nên học trước

Chọn nhánh học tiếp theo sau nhập môn

2. Thiết kế dữ liệu, luồng và giao diện

Kết nối flow, cấu trúc dữ liệu, sơ đồ và đầu ra

Mục tiêu học tập

- Giải thích đúng Flow Diagram và Data Model bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Wireframe, Architecture
- Thực hiện và giải thích kết quả bài tập: Tạo sơ đồ trang hoặc màn hình, dữ liệu mẫu, và mô tả luồng một thao tác quan trọng.
- Nhận diện và tránh sai lầm: Bắt đầu viết code khi chưa biết ứng dụng cần lưu gì.

Kiến thức cần học

Flow Diagram · Data Model · Wireframe · Architecture

Giải thích chi tiết

Vì sao cần học

Kết nối flow, cấu trúc dữ liệu, sơ đồ và đầu ra. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Flow Diagram: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Data Model: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Wireframe: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Architecture: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Thiết kế dữ liệu, luồng và giao diện ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Flow Diagram, Data Model, Wireframe, Architecture.
3. Tạo sơ đồ trang hoặc màn hình, dữ liệu mẫu, và mô tả luồng một thao tác quan trọng.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Cho AI phản biện thiết kế và ghi lại quyết định cuối.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Thiết kế dữ liệu, luồng và giao diện" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Bắt đầu viết code khi chưa biết ứng dụng cần lưu gì.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Flow Diagram, Data Model, Wireframe; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Thiết kế dữ liệu, luồng và giao diện
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

[ ] Tự định nghĩa được Flow Diagram và Data Model bằng lời của mình
[ ] Vẽ được sơ đồ/bảng có các khái niệm Flow Diagram, Data Model, Wireframe, Architecture
[ ] Đã làm: Tạo sơ đồ trang hoặc màn hình, dữ liệu mẫu, và mô tả luồng một thao tác quan trọng.
[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
[ ] Đã phân tích rủi ro: Bắt đầu viết code khi chưa biết ứng dụng cần lưu gì.
[ ] Đã tự kiểm chứng: Cho AI phản biện thiết kế và ghi lại quyết định cuối.
[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

GitHub README guidance Tài liệu
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

Nên học trước

Chọn đề tài và xác định MVP

---

3. Hiện thực hóa sản phẩm nhỏ

Xây theo từng tính năng, lưu phiên bản và xử lý lỗi

Mục tiêu học tập

- Giải thích đúng Implementation và Iteration bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Commit, Debugging

- Thực hiện và giải thích kết quả bài tập: Làm sản phẩm bằng Python CLI hoặc HTML/CSS/JS tối giản; ít nhất 5 commit rõ mục đích.
- Nhận diện và tránh sai lầm: Đổi stack liên tục vì AI đề xuất công nghệ mới.

Kiến thức cần học

Implementation · Iteration · Commit · Debugging

Giải thích chi tiết

Vì sao cần học

Xây theo từng tính năng, lưu phiên bản và xử lý lỗi. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Implementation: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Iteration: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Commit: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Debugging: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Hiện thực hóa sản phẩm nhỏ ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Implementation, Iteration, Commit, Debugging.
3. Làm sản phẩm bằng Python CLI hoặc HTML/CSS/JS tối giản; ít nhất 5 commit rõ mục đích.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Tách việc AI hỗ trợ, đọc diff, không commit dữ liệu nhạy cảm.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Hiện thực hóa sản phẩm nhỏ" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đổi stack liên tục vì AI đề xuất công nghệ mới.

- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Implementation, Iteration, Commit; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Hiện thực hóa sản phẩm nhỏ
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được Implementation và Iteration bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm Implementation, Iteration, Commit, Debugging
- [ ] Đã làm: Làm sản phẩm bằng Python CLI hoặc HTML/CSS/JS tối giản; ít nhất 5 commit rõ mục đích.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Đổi stack liên tục vì AI đề xuất công nghệ mới.
- [ ] Đã tự kiểm chứng: Tách việc AI hỗ trợ, đọc diff, không commit dữ liệu nhạy cảm.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

GitHub README guidance Tài liệu
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

Nên học trước

Thiết kế dữ liệu, luồng và giao diện

---

4. Kiểm thử, tài liệu và trình bày sản phẩm

Test chức năng, accessibility cơ bản, README, demo và giới hạn

Mục tiêu học tập

- Giải thích đúng Functional Test và README bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Accessibility, Demo
- Thực hiện và giải thích kết quả bài tập: Viết tối thiểu 10 ca thử; hoàn thành README có cách chạy, ảnh chụp và danh sách giới hạn.
- Nhận diện và tránh sai lầm: Chỉ quay video demo thành công mà không thử lỗi.

Kiến thức cần học

Functional Test · README · Accessibility · Demo

Giải thích chi tiết

Vì sao cần học

Test chức năng, accessibility cơ bản, README, demo và giới hạn. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Functional Test: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- README: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Accessibility: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Demo: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Kiểm thử, tài liệu và trình bày sản phẩm ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Functional Test, README, Accessibility, Demo.
3. Viết tối thiểu 10 ca thử; hoàn thành README có cách chạy, ảnh chụp và danh sách giới hạn.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Dùng AI review checklist nhưng tự chạy và ghi minh chứng.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Kiểm thử, tài liệu và trình bày sản phẩm" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Chỉ quay video demo thành công mà không thử lỗi.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Functional Test, README, Accessibility; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Thực hành: Kiểm thử, tài liệu và trình bày sản phẩm

```
## Cách chạy
## Tính năng
## Ví dụ sử dụng
## Kiểm thử
## Giới hạn và hướng phát triển
```

Chạy hoặc đọc từng bước và thay ít nhất hai dữ liệu đầu vào; ghi kết quả thật, không chỉ sao chép ví dụ.

Checklist

[ ] Tự định nghĩa được Functional Test và README bằng lời của mình

[ ] Vẽ được sơ đồ/bảng có các khái niệm Functional Test, README, Accessibility, Demo

[ ] Đã làm: Viết tối thiểu 10 ca thử; hoàn thành README có cách chạy, ảnh chụp và danh sách giới hạn.

[ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường

[ ] Đã phân tích rủi ro: Chỉ quay video demo thành công mà không thử lỗi.

[ ] Đã tự kiểm chứng: Dùng AI review checklist nhưng tự chạy và ghi minh chứng.

[ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

GitHub README guidance Tài liệu
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

Nên học trước

Hiện thực hóa sản phẩm nhỏ

---

5. Tổng kết kỹ năng và hướng học kế tiếp

Phân biệt điều đã hiểu, điều chưa chắc, cách tự học tiếp

Mục tiêu học tập

- Giải thích đúng Reflection và Portfolio bằng ngôn ngữ phổ thông
- Phân biệt được các khái niệm liên quan: Learning Plan, Evidence
- Thực hiện và giải thích kết quả bài tập: Viết báo cáo 1 trang: 5 điều hiểu sâu, 3 lỗi đã sửa, 2 việc cần học và link repo/demo.
- Nhận diện và tránh sai lầm: Đánh giá năng lực chỉ bằng số giờ học hoặc chứng chỉ.

Kiến thức cần học

Reflection · Portfolio · Learning Plan · Evidence

Giải thích chi tiết

Vì sao cần học

Phân biệt điều đã hiểu, điều chưa chắc, cách tự học tiếp. Đây là kiến thức nền để hiểu cách hệ thống số hoạt động và ra quyết định có căn cứ, không chỉ ghi nhớ tên công cụ.

Kiến thức trọng tâm

- Reflection: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

- Portfolio: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Learning Plan: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.
- Evidence: xác định khái niệm bằng lời của mình, chỉ ra vai trò và lấy một ví dụ cụ thể trong đời sống hoặc sản phẩm số.

Tình huống thực tế

Giả sử bạn tham gia nhóm làm ứng dụng quản lý học tập. Hãy chỉ ra kiến thức Tổng kết kỹ năng và hướng học kế tiếp ảnh hưởng đến thao tác của người dùng, dữ liệu hoặc cách nhóm xây dựng và vận hành ứng dụng như thế nào. Viết một ví dụ có đầu vào, kết quả mong đợi và điều cần kiểm tra.

Thực hành từng bước

1. Đọc tài liệu tham khảo và tự viết định nghĩa 3–5 dòng, không sao chép nguyên văn.
2. Vẽ sơ đồ hoặc lập bảng giải thích mối quan hệ giữa các khái niệm: Reflection, Portfolio, Learning Plan, Evidence.
3. Viết báo cáo 1 trang: 5 điều hiểu sâu, 3 lỗi đã sửa, 2 việc cần học và link repo/demo.
4. Thử ít nhất một trường hợp thường gặp và một trường hợp biên hoặc bất thường; ghi kết quả thực tế.
5. Tự giải thích sản phẩm thực hành trong 2 phút, có ảnh chụp hoặc tệp làm minh chứng.

Làm việc với AI

- Có thể nhờ AI: Giải thích bằng ví dụ khác, đặt câu hỏi ôn tập, gợi ý tình huống cần thử.
- Phải tự kiểm chứng: Nhờ AI hỏi phản biện để kiểm tra hiểu thật, không nhờ AI tự viết hết báo cáo.
- Không giao cho AI: Quyết định thao tác có thể gây mất dữ liệu, lộ tài khoản hoặc công bố thông tin cá nhân nếu chưa hiểu và chưa được cho phép.

Prompt mẫu:

> Tôi đang học "Tổng kết kỹ năng và hướng học kế tiếp" ở mức nhập môn CNTT. Hãy hỏi tôi lần lượt 5 câu theo độ khó tăng dần, chỉ gợi ý khi tôi bí, sau đó yêu cầu tôi tự nêu một ví dụ thực tế. Đừng làm hộ bài thực hành.

Lỗi thường gặp

- Đánh giá năng lực chỉ bằng số giờ học hoặc chứng chỉ.
- Học thuộc tên thuật ngữ nhưng không diễn giải được bằng ví dụ.
- Không kiểm tra kết quả đầu ra và không ghi lại cách xử lý lỗi.

Điều kiện hoàn thành

Giải thích được Reflection, Portfolio, Learning Plan; hoàn thành bài thực hành có minh chứng; nêu được ít nhất một lỗi phổ biến và cách phòng tránh.

Ví dụ minh họa

Mẫu đầu ra thực hành

```
Chủ đề: Tổng kết kỹ năng và hướng học kế tiếp
Tôi hiểu: ...
Ví dụ thực tế: ...
Điều cần kiểm chứng: ...
Minh chứng đã thực hiện: ...
```

Dùng mẫu để tạo ghi chú bằng lời của chính mình; bổ sung sơ đồ hoặc ảnh chụp kết quả.

Checklist

- [ ] Tự định nghĩa được Reflection và Portfolio bằng lời của mình
- [ ] Vẽ được sơ đồ/bảng có các khái niệm Reflection, Portfolio, Learning Plan, Evidence
- [ ] Đã làm: Viết báo cáo 1 trang: 5 điều hiểu sâu, 3 lỗi đã sửa, 2 việc cần học và link repo/demo.
- [ ] Đã ghi một trường hợp bình thường và một trường hợp biên/bất thường
- [ ] Đã phân tích rủi ro: Đánh giá năng lực chỉ bằng số giờ học hoặc chứng chỉ.
- [ ] Đã tự kiểm chứng: Nhờ AI hỏi phản biện để kiểm tra hiểu thật, không nhờ AI tự viết hết báo cáo.
- [ ] Đã lưu minh chứng và giải thích lại nội dung trong 2 phút

Tài liệu tham khảo

GitHub README guidance Tài liệu
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes

MDN: Getting started Tài liệu
https://developer.mozilla.org/en-US/docs/Learn_web_development/Getting_started

Nên học trước

Kiểm thử, tài liệu và trình bày sản phẩm