

AIE Entrance Exam & Interview

Student Kit

Resources:

- Python:

- <https://docs.python.org/3/>
- <https://automatetheboringstuff.com/>
- <https://packaging.python.org/>

- Git & GitHub:

- <https://git-scm.com/book/en/v2>
- <https://docs.github.com/>

- HTTP / APIs:

- <https://developer.mozilla.org/en-US/docs/Web/HTTP>
- <https://curl.se/docs/>
- <https://restfulapi.net/>

- FastAPI + Validation:

- <https://fastapi.tiangolo.com/>
- <https://docs.pydantic.dev/>
- <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

- Data Handling (CSV/JSON/Pandas):

- <https://pandas.pydata.org/docs/>
- <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON>

- SQL:

- https://sqlzoo.net/wiki/SQL_Tutorial
 - <https://www.hackerrank.com/domains/sql>
 - **Docker:**
 - <https://docs.docker.com/get-started/>
 - **Testing:**
 - <https://docs.pytest.org/>
-

Exercises: (Implement them in Python)

- Write a function `is_balanced(s)` that supports `()[]{}` and ignores other characters.
 - Write a function `top_k_words(text, k)` that returns the `k` most frequent tokens (case-insensitive, strip punctuation).
 - Write a function `longest_unique_substring(s)` that returns the length of the longest substring without repeating characters ($O(n)$).
 - Given a list of URLs, fetch them concurrently with a maximum concurrency limit and collect the successful JSON results.
 - Build a FastAPI app with GET `/health` and POST `/task` using Pydantic models
-

Mini-Project (Recommended): Task Assistant API

- Build POST `/assist` that returns validated JSON; store each request/response with latency in SQLite; add GET `/runs` to inspect history.

AIE Entrance Exam (Sample)

- **MCQ Questions:** Examples:
 - how you would detect an LLM endpoint getting worse over time.
 - What does RAG stand for in RAG pipelines?

- **Coding Problems:** Example:

- Python: Write a function named `find_duplicate_integer(integers)` that takes a list of integers containing numbers from 1 to n , with exactly $n + 1$ elements. Since there are $n + 1$ numbers but only n possible values, one integer is duplicated. Your function must return the duplicated integer. Input: `[1, 3, 4, 2, 2]` $\Rightarrow$ Output: 2