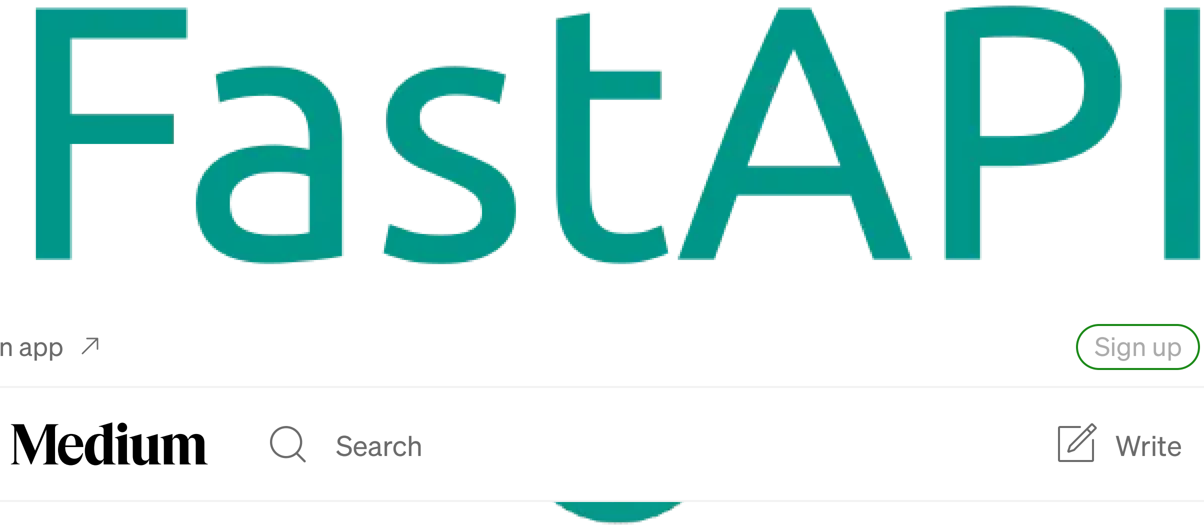
FastAPI: An Example of Test Driven Development ⚡

Kasper Junge · Follow
2 min read · Jun 6, 2023

137 1



Here is a simple example on how to apply test driven development (TDD) using FastAPI.

Prerequisites: Install FastAPI, Uvicorn (an ASGI server), and pytest.

```
pip install fastapi uvicorn pytest
```

No time to waste, let’s get started 🏃

Step 1: Write a Failing Test

We’ll start by creating an endpoint `/books/{id}` that retrieves a book by its ID. First, write a failing test in `test_main.py`:

```
from fastapi.testclient import TestClient
from main import app

client = TestClient(app)

def test_read_book():
    response = client.get("/books/1")
    assert response.status_code == 200
    assert response.json() == {
        "id": 1,
        "title": "1984",
        "author": "George Orwell",
        "availability": True
    }
```

Run the tests using pytest:

```
pytest
```

Our test fails because we’ve not yet implemented the endpoint.

Step 2: Implement The Code

Let’s build our FastAPI app in `main.py`:

```
from fastapi import FastAPI

app = FastAPI()

@app.get("/books/{id}")
def read_book(id: int):
    pass
```

If we run the tests again, they’ll still fail as our endpoint doesn’t return the expected response.

Step 3: Make the Test Pass

Now, let’s return a hard-coded book from the endpoint:

```
@app.get("/books/{id}")
def read_book(id: int):
    return {
        "id": id,
        "title": "1984",
        "author": "George Orwell",
        "availability": True
    }
```

Running the tests again, they should pass because our endpoint now behaves as expected.

Step 4: Refactor

At this stage, you refactor your code while ensuring the tests still pass. Let’s assume we’ve created a `Book` model and a `FakeDB` class that simulates a database:

```
from pydantic import BaseModel
from typing import Dict

class Book(BaseModel):
    id: int
    title: str
    author: str
    availability: bool

class FakeDB:
    def __init__(self):
        self.data: Dict[int, Book] = {}
    def get_book(self, id: int) -> Book:
        return self.data.get(id)

db = FakeDB()
db.data = {
    1: Book(id=1, title="1984", author="George Orwell", availability=True)
}

@app.get("/books/{id}", response_model=Book)
def read_book(id: int):
    return db.get_book(id)
```

If we run the tests again, they should still pass, and we now have a more realistic (though still simplified) application.

Thank you for taking the time to read this! I sincerely hope that your time was well-spent.

Fastapi Test Driven Development

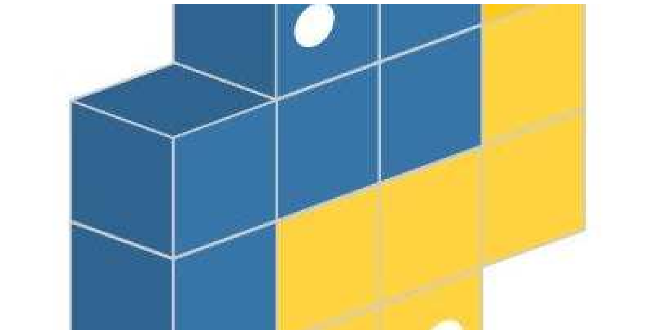
Written by Kasper Junge

Follow ○

448 Followers

Machine Learning Engineer 🧑‍💻 AI, LLMs, Deep Learning, Python 🔥

More from Kasper Junge



Kasper Junge

20 Most Downloaded Python Packages Explained

Python is a powerhouse in the programming world, and its wide array of packages is a big...

3 min read · Jan 23, 2024

Kasper Junge

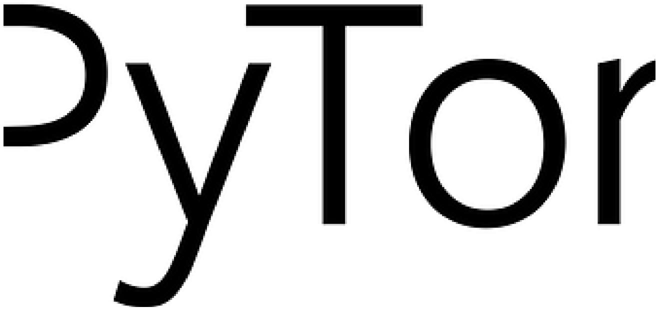
yfinance: 10 Ways to Get Stock Data with Python

The yfinance library offers Python users a seamless way to retrieve stock data from...

2 min read · Oct 31, 2023

👏 404 💬 2

👏 106 💬



Kasper Junge

10 Principles of PyTorch

Welcome to this concise guide on the principles of PyTorch. Whether you're a...

2 min read · Oct 19, 2023

👏 493 💬 6

Kasper Junge

How to Use Celery 🥦

Processing tasks asynchronously is a common requirement for applications that...

6 min read · Jun 12, 2023

👏 137 💬

See all from Kasper Junge

Recommended from Medium

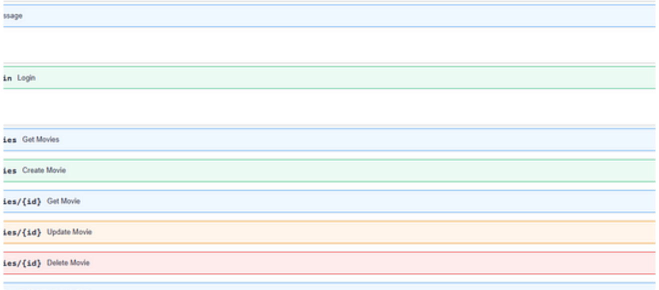
Jeremiah Talamantes in Compliiant.io Blog

FastAPI: Best Security Practices

Best Security Practices for FastAPI

✦ · 4 min read · Jan 2, 2024

254 4



Jordi Oltra

Basic FastAPI good practices

How works a basic API

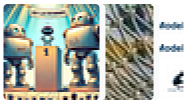
10 min read · Dec 3, 2023

82

Lists

Coding & Development

11 stories · 501 saves



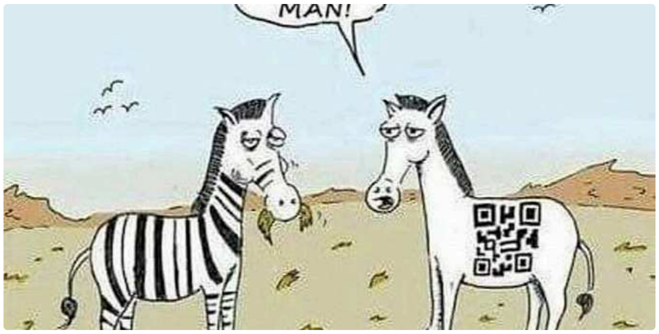
Natural Language Processing

1279 stories · 772 saves

Thomas Aitken

Setting up a FastAPI App with Async SQLALchemy 2.0 & Pydant...

Early this year, a major update was made to SQLAlchemy with the release of SQLAlchem...



Dmytro Parfeniuk in Python in Plain English

🌶️ FastAPI & Pydantic 2.4 & SQLAlchemy 2.0 & More

This article is an update of the 🐍 Python Backend Project Advanced Setup (FastAPI...

11 min read · Oct 20, 2023

282 3

8 min read · Oct 10, 2023

362 7

Abhishek Singh

FastAPI vs. Django Rest Framework: A Comprehensive...

In the world of web development, building robust and efficient APIs is a fundamental...

3 min read · Nov 5, 2023

21

Hirusha Fernando in Stackademic

FastAPI Role Base Access Control With JWT

FastAPI is a modern, high-performance, web framework used to build APIs with Python...

7 min read · Jan 19, 2024

157 2

See more recommendations