High Performance API

FastAPI is very fast because it uses **asynchronous programming**.

It runs on **Uvicorn** ASGI server.

Benefits:

- High speed
- Easy documentation
- Automatic API docs

To **save and run your FastAPI code**, follow these simple steps.

---

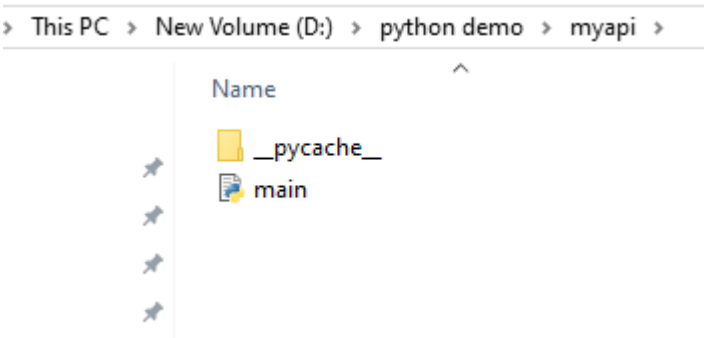
1 Create a Python File

Open any editor like:

- Visual Studio Code
- PyCharm
- Notepad (basic)

Create a file named inside a project folder myapi as shown below:

`main.py`



---

2 Paste Your Code in the File

Open **main.py** and paste this code:

```
from fastapi import FastAPI

app = FastAPI()

@app.get("/")
def home():
    return {"message": "AI API running"}

@app.get("/predict")
def predict(area: int):
    price = area * 5000
    return {"predicted_price": price}
```

Save the file.

---

3 Install Required Libraries

Open terminal and run:

```
pip install fastapi uvicorn
```

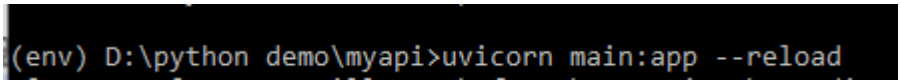
Here:

- **FastAPI** → framework
- **Uvicorn** → server to run the API

4️⃣ Run the FastAPI Server

Go to the project your myapi folder where **main.py** is saved and run:

```
uvicorn main:app –reload
```



Explanation:

```
main  → file name (main.py)
app   → FastAPI object
--reload → auto restart when code changes
```

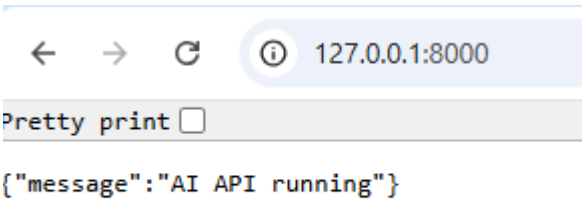
5️⃣ Open API in Browser

Open:

```
http://127.0.0.1:8000
```

You will see:

```
{"message":"AI API running"}
```



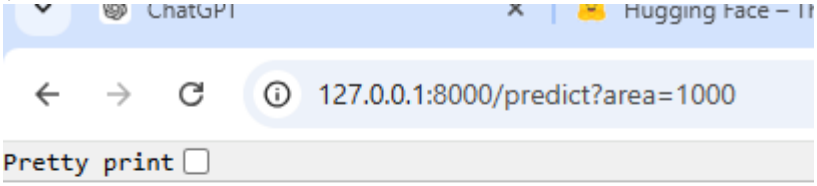
6️⃣ Test the Prediction API

Open:

```
http://127.0.0.1:8000/predict?area=1000
```

Output:

```
{
 "predicted_price": 5000000
}
```



7 Automatic API Documentation

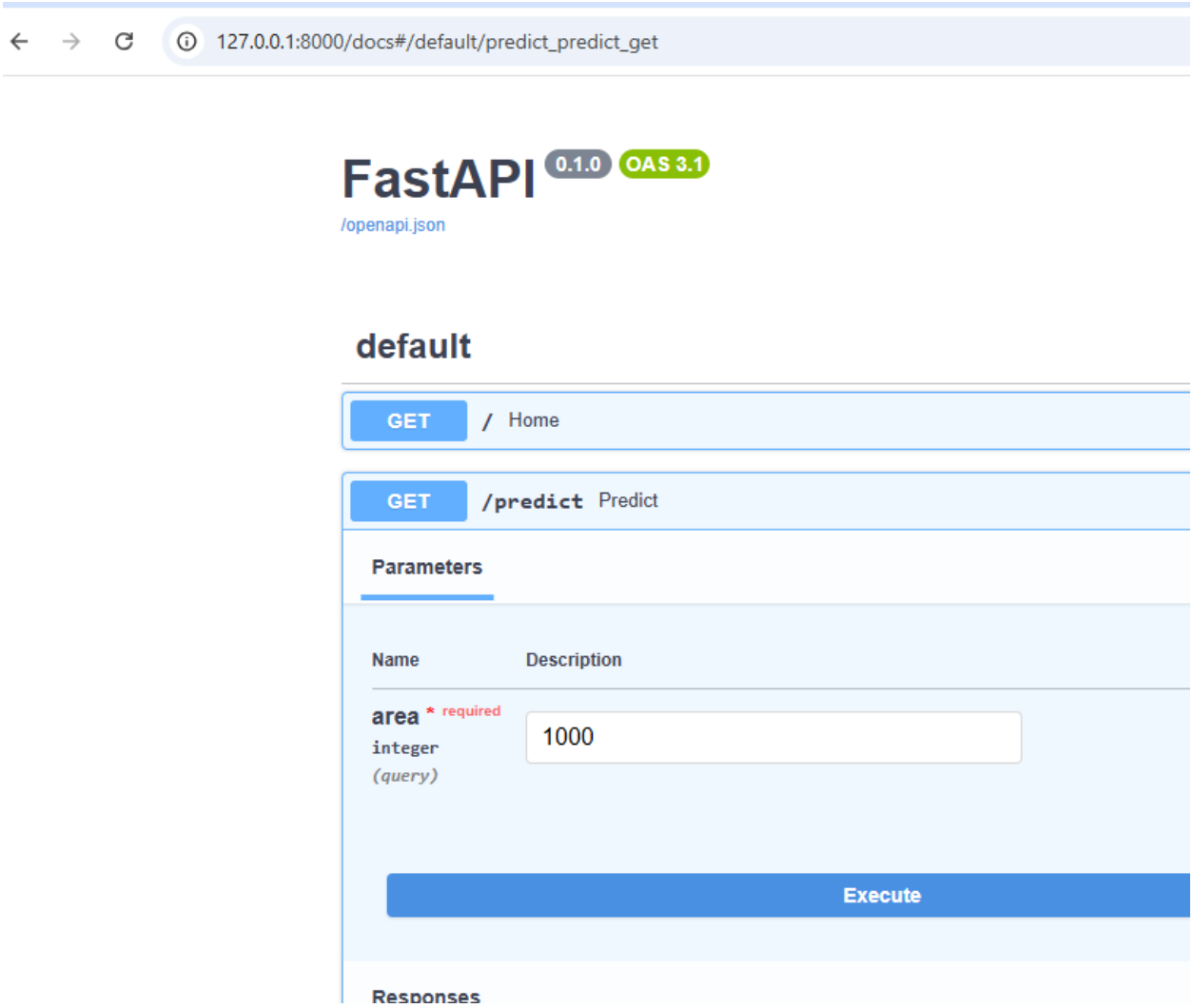
FastAPI automatically creates documentation.

Open:

```
http://127.0.0.1:8000/docs
```

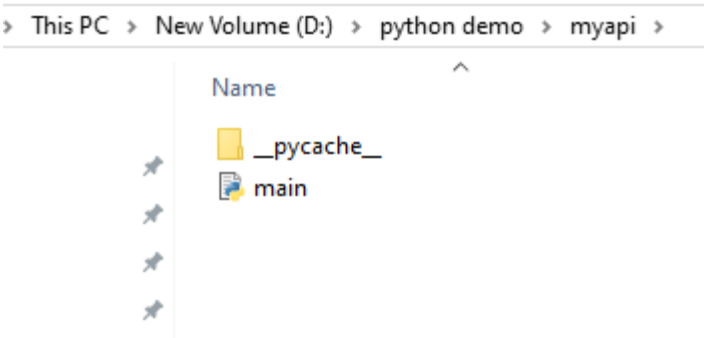
This uses **Swagger UI**.

You can test APIs directly there.



Your project folder should look like

```
myapi/
├── main.py
```



Real AI Architecture

Typical Generative AI system:

```
User
  ↓
Frontend (React)
  ↓
FastAPI Backend
  ↓
LLM Model
  ↓
Database / Vector DB
```

Vector databases used with GenAI:

- **Pinecone**
- **Weaviate**
- **Chroma**

Simple Explanation

Without FastAPI:

```
AI model runs only in Python file
```

With FastAPI:

```
AI model becomes a web service used by apps worldwide
```

 In short:-

FastAPI is used to **deploy and serve AI models as APIs so applications can use them.**