

---

---

---

---

---



**Github Repo:** [https://github.com/architsharm/flask\\_demo2](https://github.com/architsharm/flask_demo2)

Please ensure you are using scimitar-learn==1.0.2 for the model in code to work

# Api development using Flask

Start at 9:05

⇒ Last class

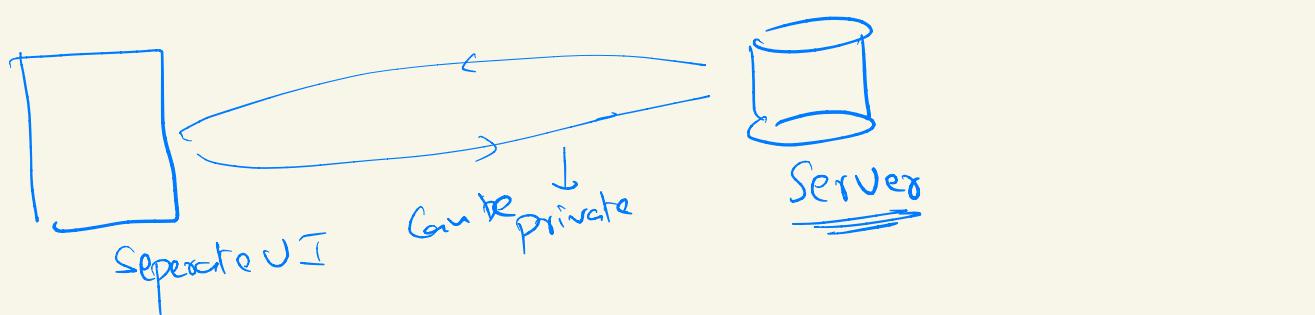
- ↳ Created a UI → powered by ML model.
- ↳ Deploy it → → github ↳ URL

⇒ \* Streamlit UI →

- ↳ Flipkart → • Complex UI →

- Deployment → private

\*



⇒ API →

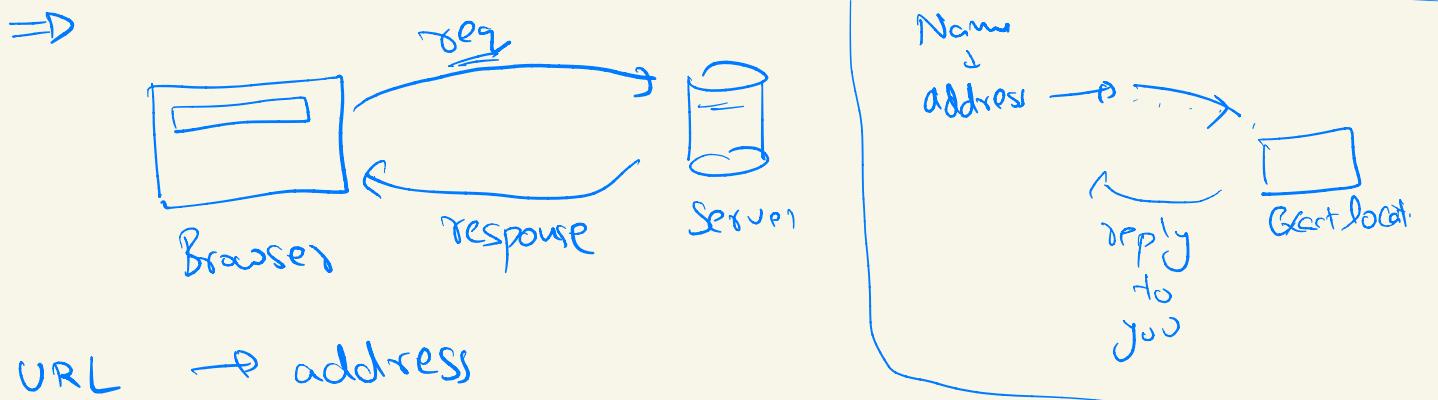
- ↳ Scraping → fetch data from a URL  
using tags. → \*\*

↳ URL → info

hits an URL + some parameters → get back info

⇒ Uber/ola → all of them uses the maps

↓  
who created maps → ~~google~~



URL → address

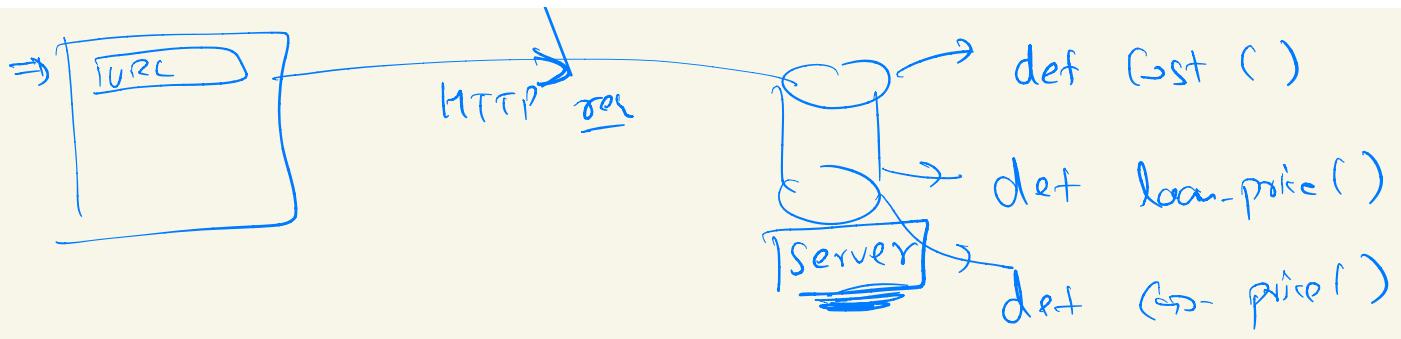
⇒ All req/res are governed by protocols → HTTP  
a set of rules

⇒ ML model 1 → ban ~~price~~ approval ←

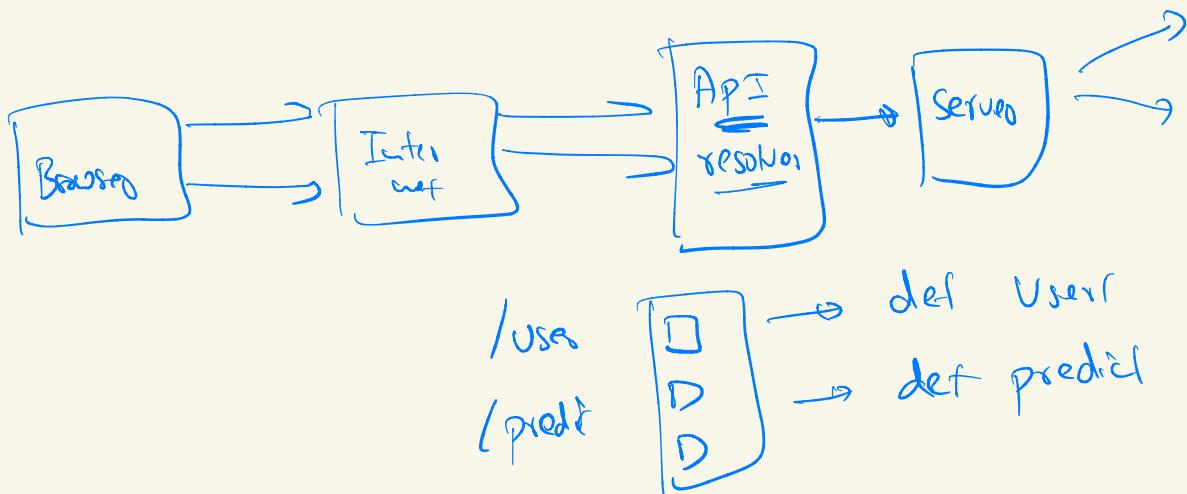
• ML model 2 → Cor price. ←

→ user info ←

→ user address ←



www.scaler.com /users  
api endpoint



⇒ Response Code →

- ① 200 → request was success.
- ② 404 → resource not found → server is missing api-endpt
- ③ 500 → Server Error

⇒ Requests →

↳ 1) to get data from server

2) to send data to Server

3) to update data on server

4) delete something server

a) GET → Just fetches the data

b) POST → Add data to Server

c) PUT | PATCH → Update data on Server

d) DELETE → Delete the data

⇒ Today - Simple api Server

• Deploy ML model as api

↳ Loan Approval Model

⇒ Fleks →

→ Fastapi

→ Django → Complicated

Let's Start  
Coding

⇒ Flask → Server + API-endpt

⇒ ML model on Streamlit

→ take user inputs

→ remember encoding

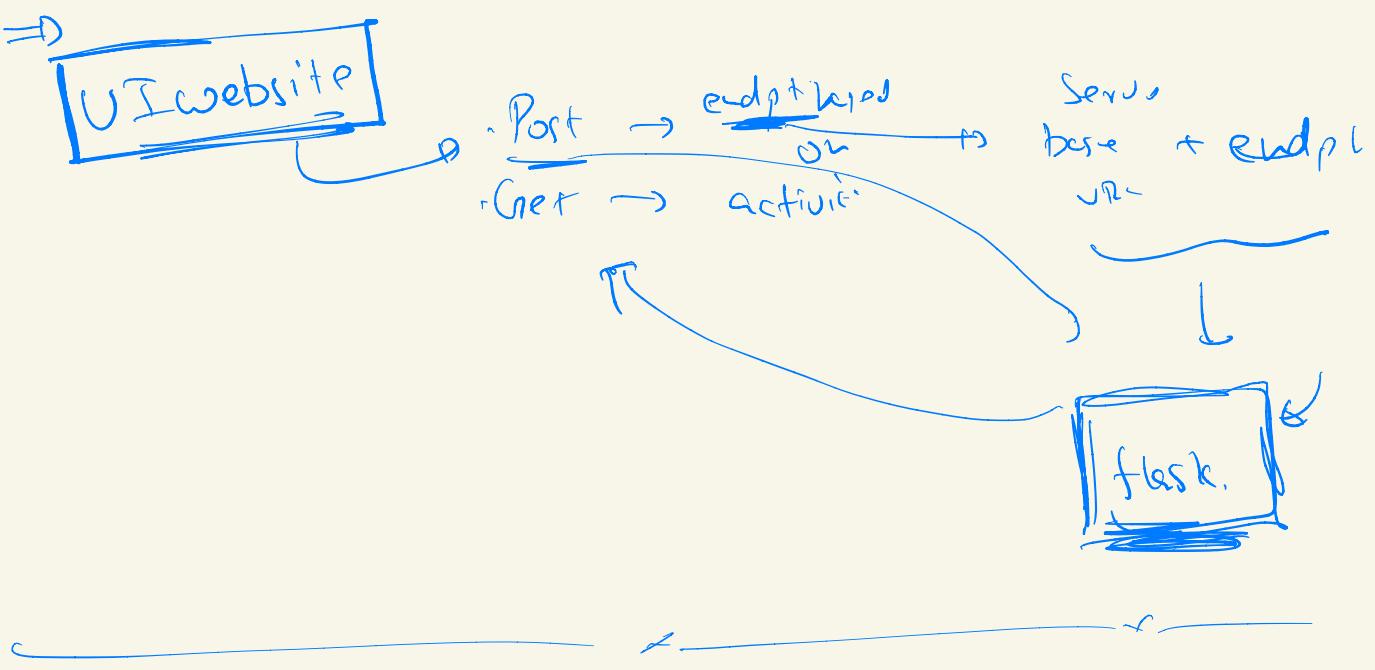
→ load a model → give exact same  
columns as input



Showcase the prediction

⇒ Postman → not everyone has it.

↗ Request body  
↙ Browsing



Devops → APIs are working

→ Python by default is a single core language →

# 16GB Ram 8 core processor ←

2GB ×8

2GB → 1.7GB

→ deploy my application using flask

→ 100 people are requesting

→ flask → less req → sequentially output

→ gunicorn

→ (NGIN + flask + threaded gunicorn) →

→ 100/sec req

→ (flask) → local machine



Web farms / cloud servers → upcoming changes

ECS →