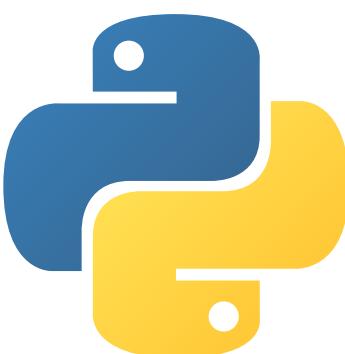


Programmation Python

API : Application Programming Interface

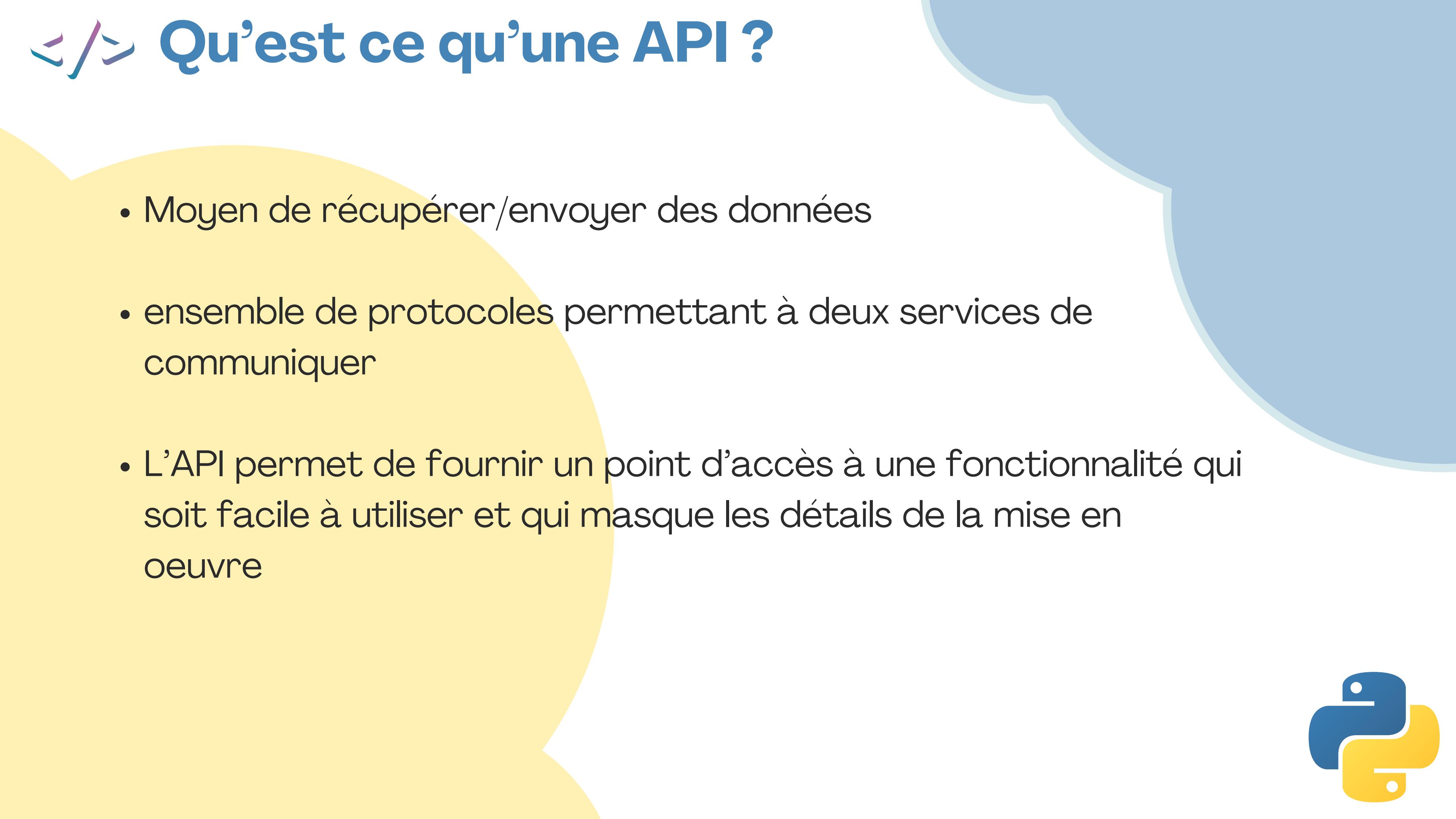
SAÏDI MASSINISSA





Qu'est ce qu'une API ?



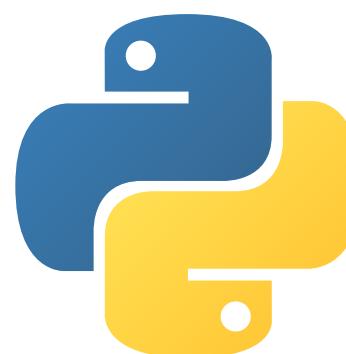
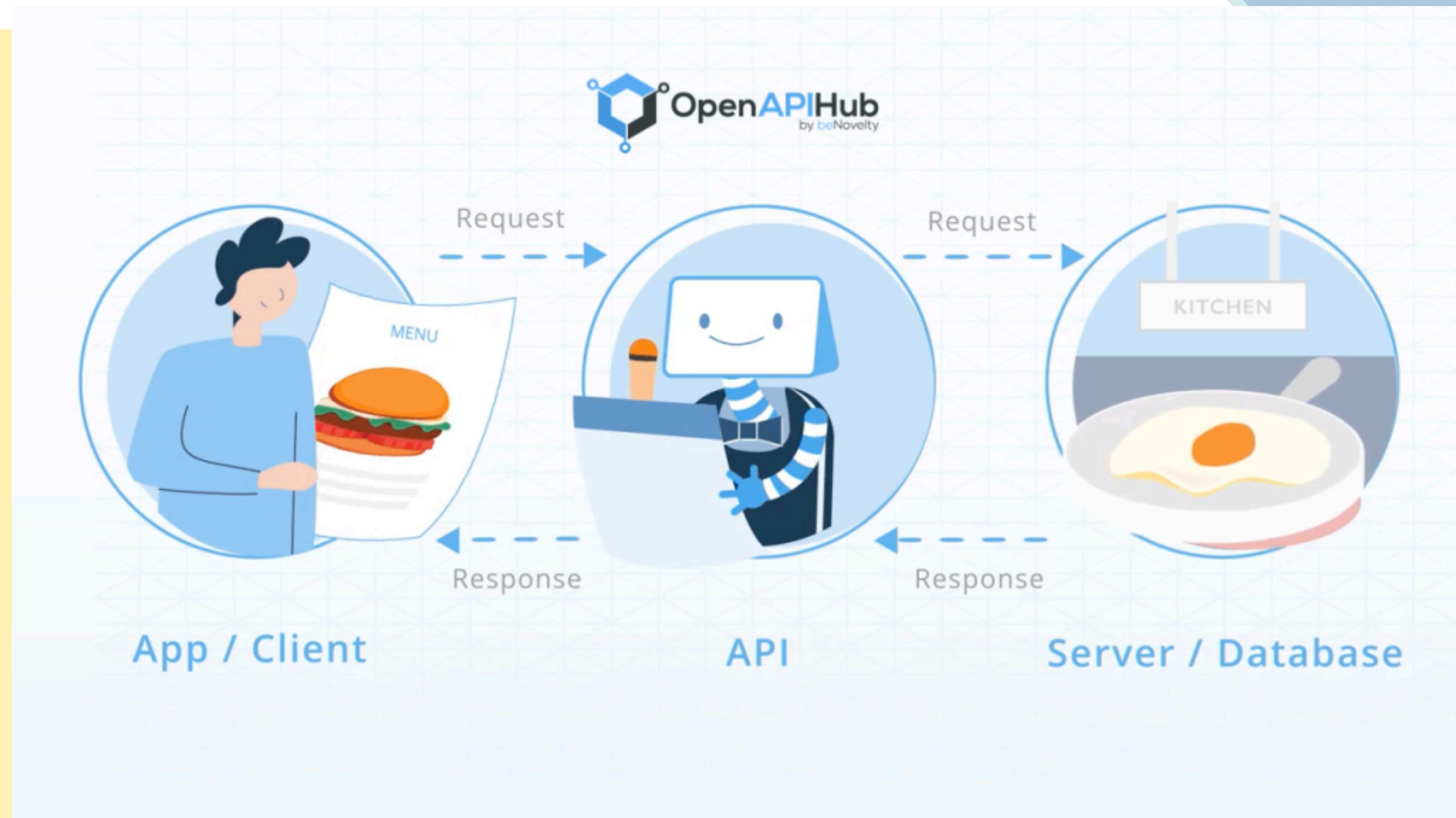


`</>` Qu'est ce qu'une API ?

- Moyen de récupérer/envoyer des données
- ensemble de protocoles permettant à deux services de communiquer
- L'API permet de fournir un point d'accès à une fonctionnalité qui soit facile à utiliser et qui masque les détails de la mise en oeuvre



Qu'est ce qu'une API ?





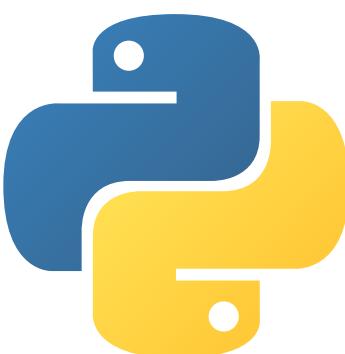
Pourquoi utiliser une API ?



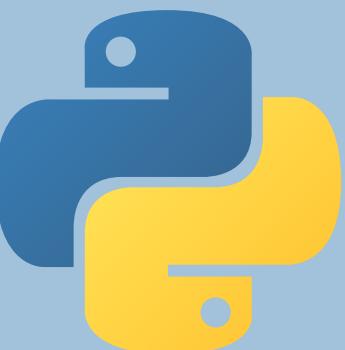


Pourquoi utiliser une API ?

- Communication entre deux services
- Partage de données de manière standardisée
- Gagner du temps de développement
- **Sécurité** : Une API ne donne pas tout accès aux serveurs, elle ne fournit que ce qui est prévu



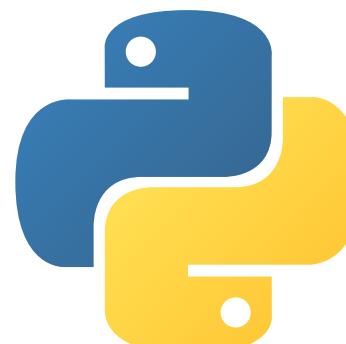
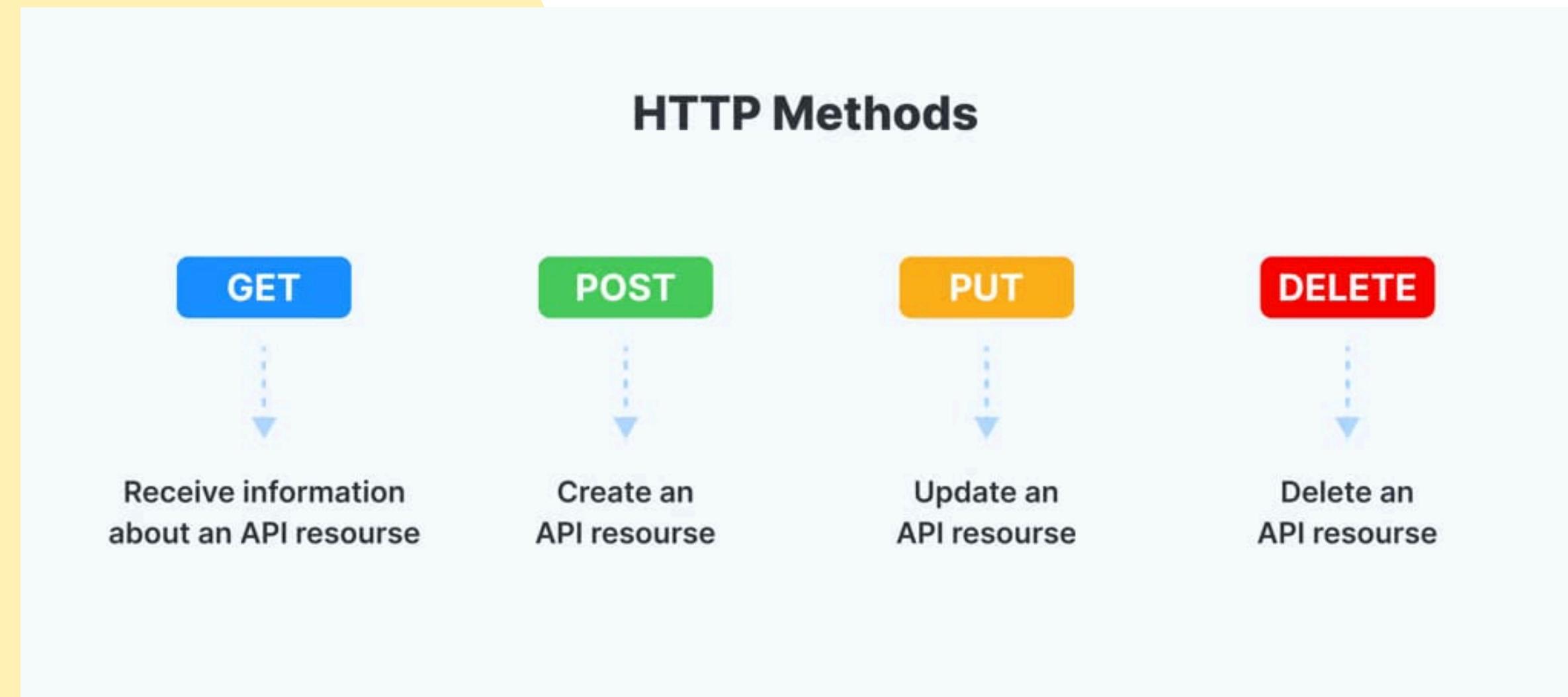
Les différents types d'API



Les différents types d'API

Il existe plusieurs types d'API : **REST**, GraphQL, SOAP

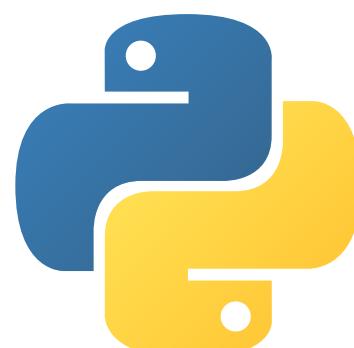
Une API REST (la + courante) fonctionne autour de requêtes HTTP

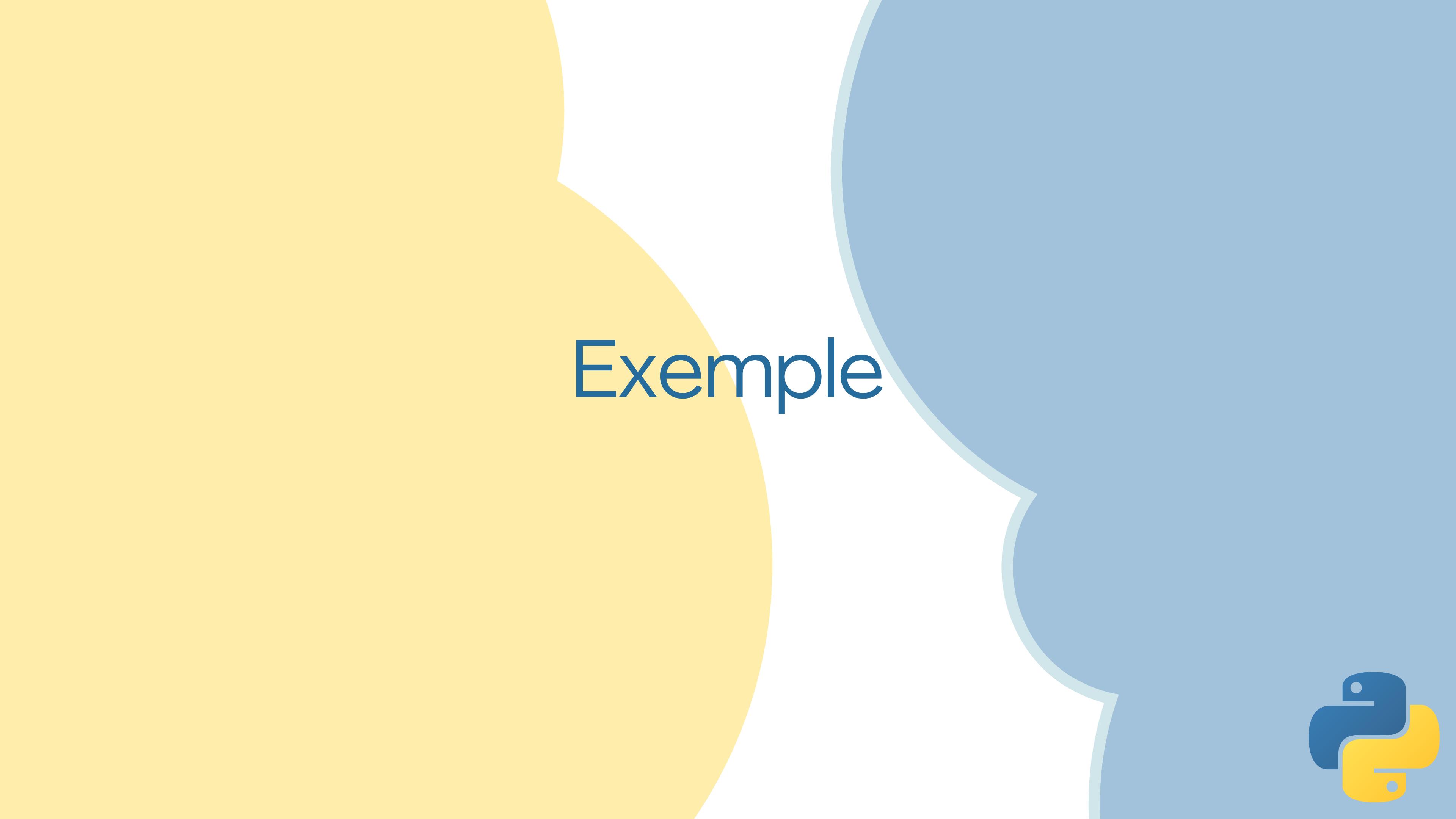


</> Les différents types d'API

Structure d'une requête API

- **url** : adresse du service → <https://api.exemple.com>
- **endpoint** : <https://api.exemple.com/users/5>
- **query** : <https://api.exemple.com/search/?param1=X¶m2=Y>
- **retour** :
 - HTTP code : 200, 400, 500... [doc mozilla](#)
 - body : {"**id**": 5, "**last_name**": "Jaeger", "**first_name**": "Eren"}

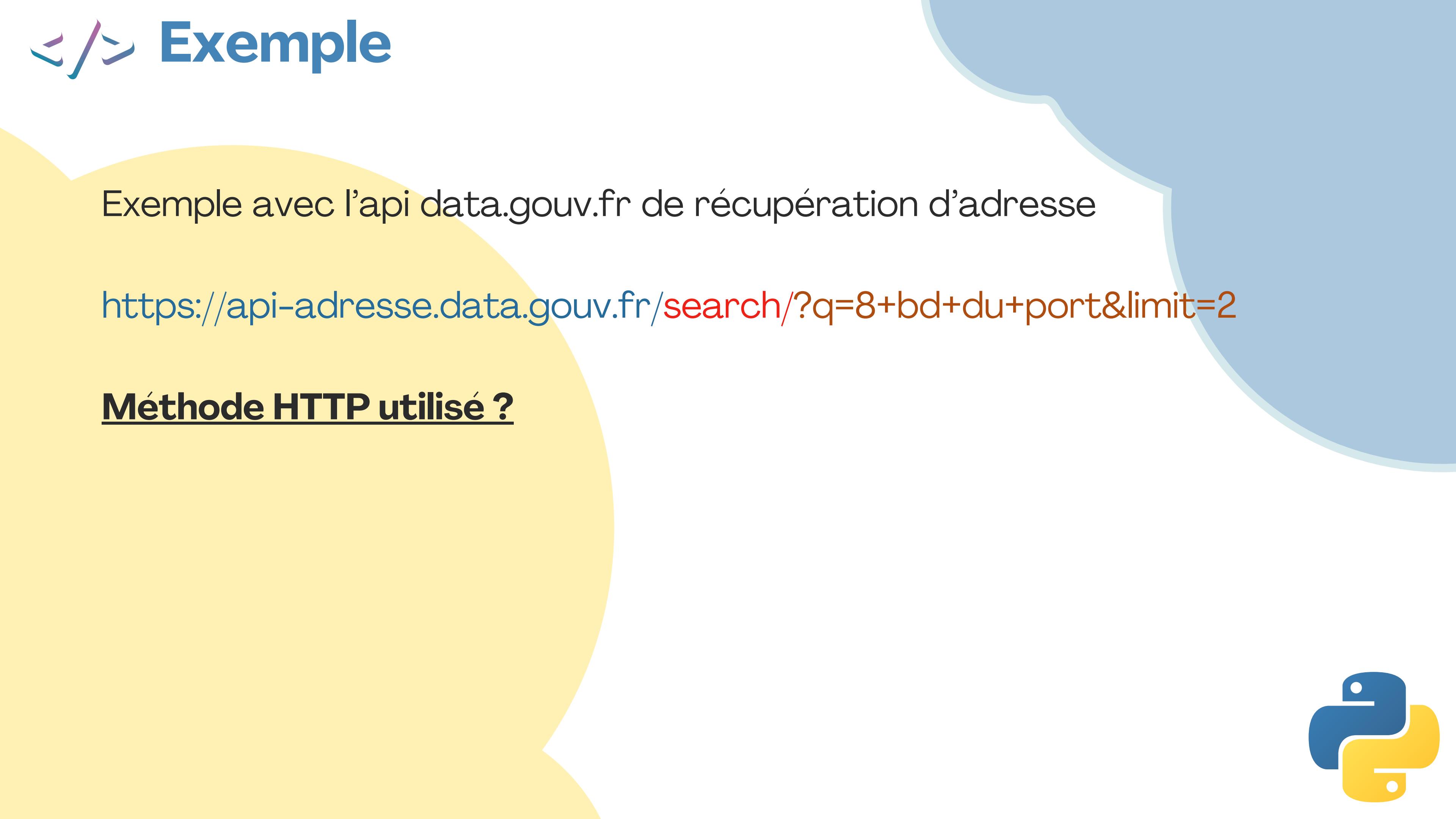




The background features two large, overlapping abstract shapes. On the left is a yellow shape with a white outline, resembling a stylized figure or a rising sun. On the right is a blue shape with a white outline, resembling a wave or a rising moon. The two shapes overlap in the center.

Exemple





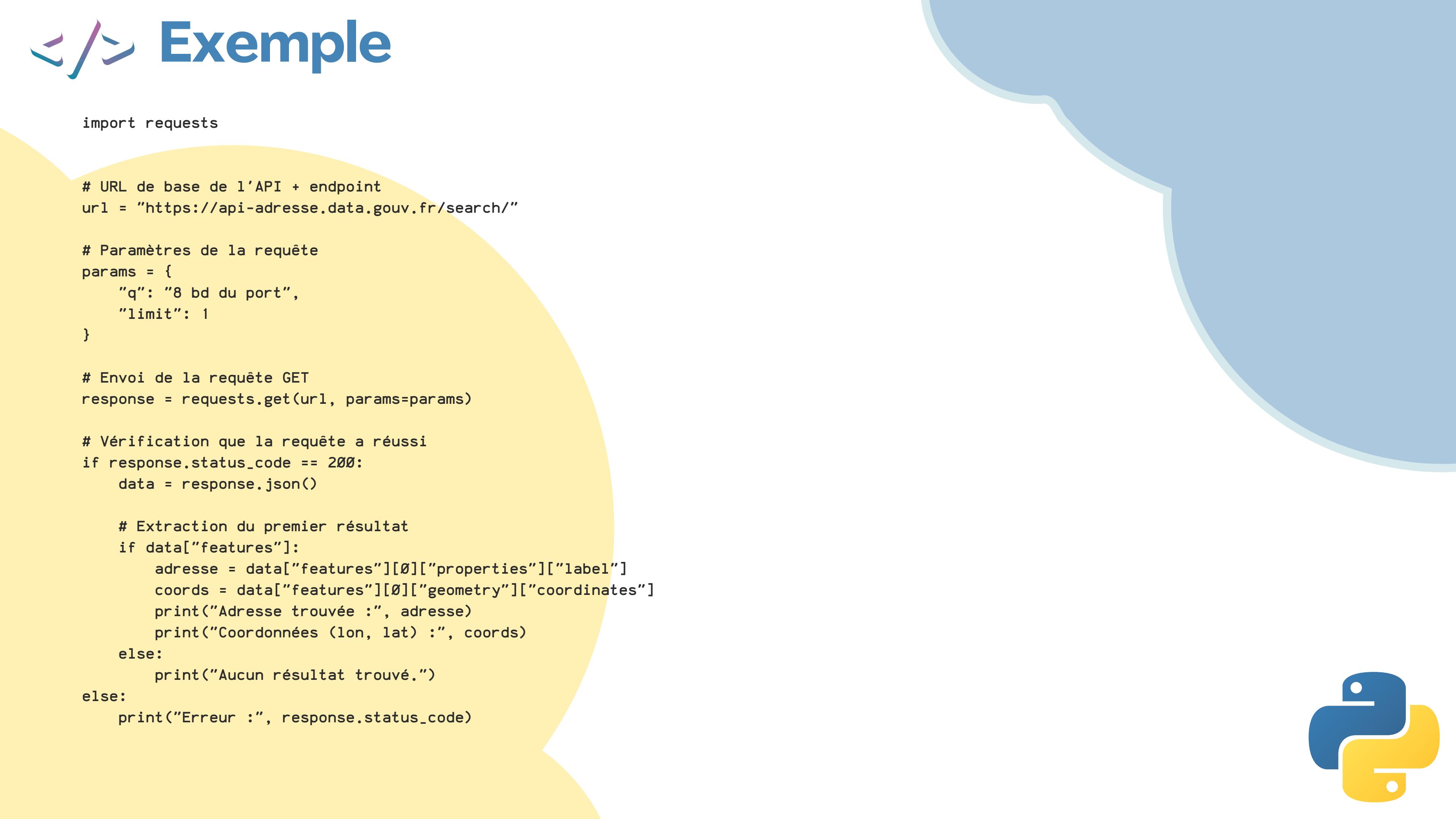
</> Exemple

Exemple avec l'api data.gouv.fr de récupération d'adresse

<https://api-adresse.data.gouv.fr/search/?q=8+bd+du+port&limit=2>

Méthode HTTP utilisé ?





Exemple

```
import requests

# URL de base de l'API + endpoint
url = "https://api-adresse.data.gouv.fr/search/"

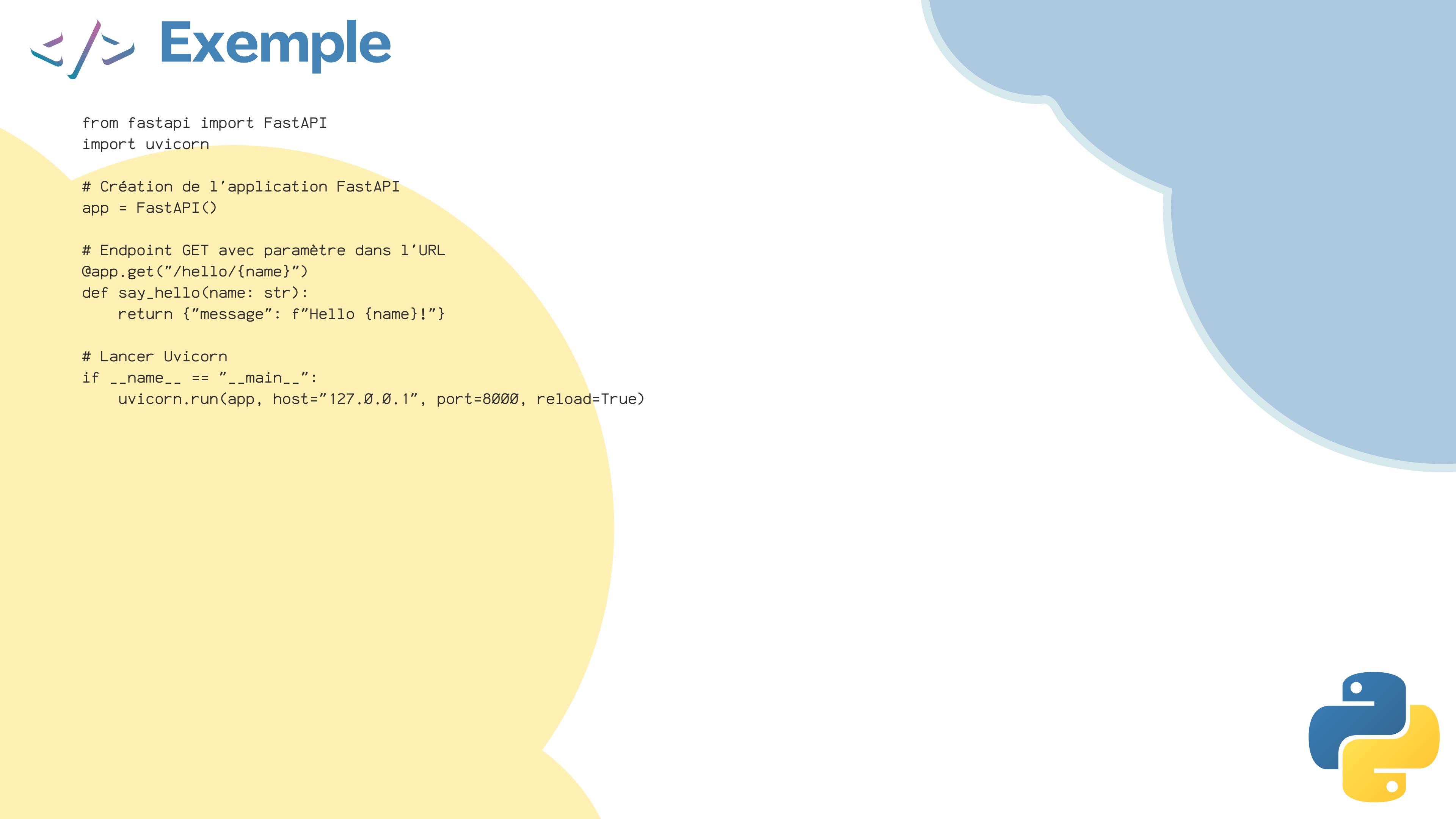
# Paramètres de la requête
params = {
    "q": "8 bd du port",
    "limit": 1
}

# Envoi de la requête GET
response = requests.get(url, params=params)

# Vérification que la requête a réussi
if response.status_code == 200:
    data = response.json()

    # Extraction du premier résultat
    if data["features"]:
        adresse = data["features"][0]["properties"]["label"]
        coords = data["features"][0]["geometry"]["coordinates"]
        print("Adresse trouvée :", adresse)
        print("Coordonnées (lon, lat) :", coords)
    else:
        print("Aucun résultat trouvé.")
else:
    print("Erreur :", response.status_code)
```





Exemple

```
from fastapi import FastAPI
import uvicorn

# Création de l'application FastAPI
app = FastAPI()

# Endpoint GET avec paramètre dans l'URL
@app.get("/hello/{name}")
def say_hello(name: str):
    return {"message": f"Hello {name}!"}

# Lancer Uvicorn
if __name__ == "__main__":
    uvicorn.run(app, host="127.0.0.1", port=8000, reload=True)
```

