Tutoriel Flask

Net and Light

2023 - 2024

Contents

1 Introduction

Flask est un micro-framework web pour Python, conçu pour être léger et modulaire. Il est idéal pour les petits projets et les prototypes, mais il est également puissant et extensible pour les applications plus grandes.

2 Installation de flask

Avant de commencer, vous devez installer Flask. Utilisez `pip` pour installer Flask facilement.

```
pip install Flask
```

3 Création d'une application simple

Commençons par créer une application Flask simple. Créez un fichier nommé `app.py` et ajoutez-y le code suivant :

```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def home():
    return 'Hello, World!'

if __name__ == "__main__":
    app.run(debug=True)
```

4 Définir des routes

Les routes définissent les URL auxquelles votre application peut répondre. Utilisez le décorateur `@app.route` pour définir des routes.

```
@app.route('/about')
def about():
    return 'About Page'
```

5 Gestion des paramètres dans les URL

Flask permet de capturer des paramètres depuis l'URL. Voici comment ajouter une route qui accepte un paramètre.

```
@app.route('/user/<username>')
def show_user_profile(username):
    return f'User {username}'
```

6 Gestion des méthodes HTTP

Par défaut, les routes Flask ne répondent qu'aux requêtes GET. Vous pouvez spécifier d'autres méthodes HTTP comme POST.

```
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        return 'Do the login'
    else:
        return 'Show the login form'
```

7 Utilisation de Flask pour retourner des JSON

Flask peut facilement retourner des réponses JSON. Utilisez la fonction `jsonify` pour créer une réponse JSON.

```
from flask import jsonify

@app.route('/data')
def data():
    return jsonify({'key': 'value', 'key2': 'value2'})
```

8 Gestion des erreurs

Vous pouvez gérer les erreurs HTTP et retourner des pages d'erreur personnalisées.

```
@app.errorhandler(404)
def page_not_found(e):
    return 'This page does not exist', 404
```

9 Conclusion

Ce tutoriel vous a introduit les bases de Flask. Pour aller plus loin, explorez la documentation officielle de Flask : `https://flask.palletsprojects.com/`.