

Activité : les routes avec Flask

Définir une route à partir d'un lien

dans la page HTML

- une URL dans un lien en précisant le nom de la route :

```
<a href="/exemple">accéder à la page exemple</a>
```

- une URL dans un lien en utilisant la fonction `urlfor()` qui précise le nom de la vue : `<code html> accéder à la page exemple </code> ===== dans la page python de l'application ===== * pour la réponse du serveur, utilisation d'une page HTML comme template avec des variables <code python> # @ est un décorateur et app.route définit la route /exemple @app.route('/exemple') def exemple(): # la fonction exemple est une vue # définir les actions à faire # définir le contenu des variables à utiliser dans le template donnee = "Information à afficher" rendertemplate('exemple.html', message=donnee) </code> * pour la réponse du serveur, utilisation d'une redirection en précisant la vue : Méthode conseillée <code python> @app.route('/autreexemple') def exemple(): # définir les actions à faire return redirect(urlfor('exemple')) </code> * pour la réponse du serveur, utilisation d'une redirection en précisant la vue : Méthode NON conseillée <code python> @app.route('/autreexemple') def exemple(): # définir les actions à faire return redirect('/exemple') </code> ===== Définir une route à partir d'un formulaire ===== dans la page HTML ===== * une URL dans un formulaire en précisant la méthode HTTP POST <code html> <form action="/exemple" method="POST"> </code> ===== dans la page python de l'application ===== * filtrage sur la méthode POST <code python> # @ est un décorateur et app.route définit la route /exemple @app.route('/exemple', methods=['POST']) def exemple(): # la fonction exemple est une vue donnee = "Information à afficher" render_template('exemple.html', message=donnee) </code> ===== Définir une route personnalisée ===== dans la page HTML ===== * une URL dans un lien en précisant le nom de la route et en ajoutant une information : ici 1 <code html> accéder à la page exemple </code> * une URL dans un lien en utilisant la fonction urlfor() qui précise le nom de la vue et le nom de l'information supplémentaire: <code html> accéder à la page exemple </code> ===== dans la page python de l'application ===== * un contrôle sur le type entier de l'information supplémentaire est ajouté <code python> @app.route('/exemple/<int:id>') def exemple(id): # le paramètre id reçu est passé en paramètre à la fonction # définir les actions à faire render_template('exemple.html') </code> ===== Retour à l'activité : Utiliser le framework Flask pour créer un site Web dynamique =====`

- Activité : Utiliser le framework Flask pour créer un site Web dynamique

From:
/ - Les cours du BTS SIO

Permanent link:
/doku.php/isn/flask_todo06

Last update: 2019/03/23 18:44

