

IN20 Informatique

# Mini-Projet

## Application Web en Python

*Sébastien Combéfis, Quentin Lurkin*

*lundi 5 octobre 2015*

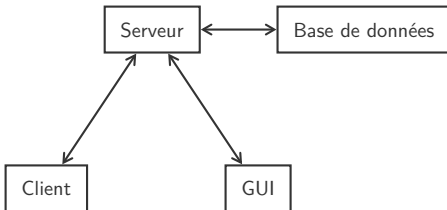


Ce(tte) œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Pas de Modification 4.0 International.

# Vue globale

- **Application web** de gestion

- Serveur web en Bottle
- Client d'édition avec interface graphique Tk
- Base de données en JSON



Bottle



# Framework Bottle

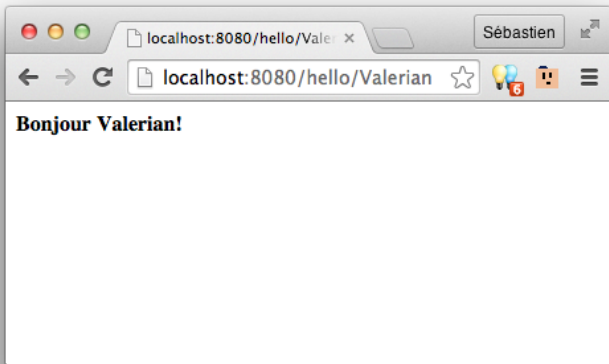
- Chargement de modules depuis la **librairie Bottle**
- Définition d'une nouvelle **route**
  - Décorateur @route pour chaque route désirée
  - Paramètres d'une route entre chevrons <>
  - Construction d'un contenu HTML avec template

```
1 from bottle import route, run, template
2
3 @route('/hello/<name>')
4 def index(name):
5     return template('<b>Bonjour {{name}}!</b>', name=name)
6
7 run(host='localhost', port=8080)
```

# Mon premier site web

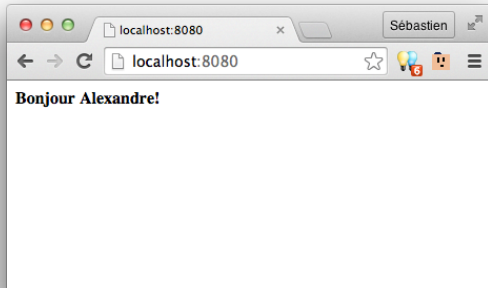
- Accès à la route `/hello/Valerian`

*Site web local sur `http://localhost:8080`*



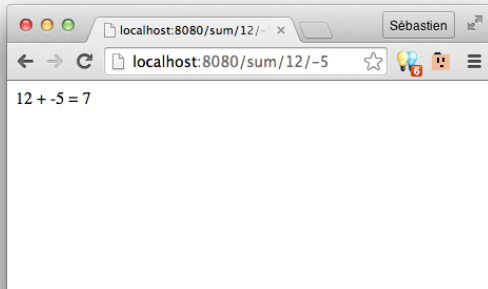
# Route par défaut

```
1 from bottle import route, run, template
2
3 @route('/')
4 @route('/hello/<name>')
5 def index(name='Alexandre'):
6     return template('<b>Bonjour {{name}}!</b>', name=name)
7
8 run(host='localhost', port=8080)
```



# Route dynamique

```
1 from bottle import route, run, template
2
3 @route('/sum/<a>/<b>')
4 def sum(a, b):
5     sum = int(a) + int(b)
6     return template('{{a}} + {{b}} = {{s}}', a=a, b=b, s=sum)
7
8 run(host='localhost', port=8080)
```





## ■ Limiter/transformer les URLs valides

*:int pour un entier...*

```
1 from bottle import route, run, template
2
3 @route('/sum/<a>/<b>')
4 def sum(a, b):
5     sum = int(a) + int(b)
6     return template('{{a}} + {{b}} = {{s}}', a=a, b=b, s=sum)
7
8 run(host='localhost', port=8080)
```

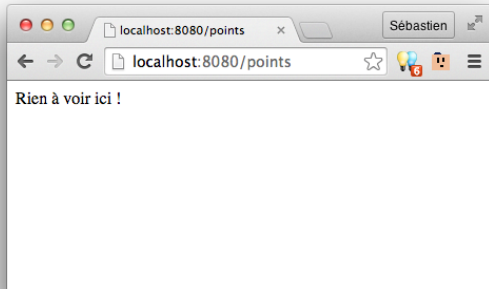
# Contenu statique

```
1 from bottle import route, run, template, static_file
2
3 @route('/static/<filename>')
4 def server_static(filename):
5     return static_file(filename, root='./static')
6
7 run(host='localhost', port=8080)
```



# Page d'erreur

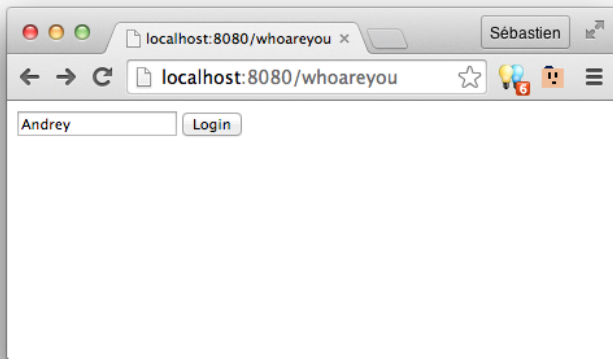
```
1 from bottle import route, run, template, error
2
3 @error(404)
4 def error404(error):
5     return 'Rien à voir ici !'
6
7 run(host='localhost', port=8080)
```



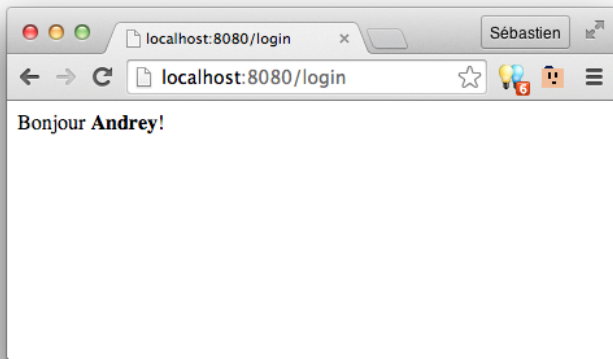
# Traiter un formulaire (1)

```
1 from bottle import route, run, template, post, request
2
3 @route('/whoareyou')
4 def whoareyou():
5     return '''
6         <form action="/login" method="post">
7             <input type="text" name="name" />
8             <input type="submit" value="Login" />
9         </form>
10    '''
11
12 @post('/login')
13 def login():
14     name = request.forms.get('name')
15     return template('Bonjour <b>{{name}}</b>!', name=name)
16
17 run(host='localhost', port=8080)
```

# Traiter un formulaire (2)



# Traiter un formulaire (3)



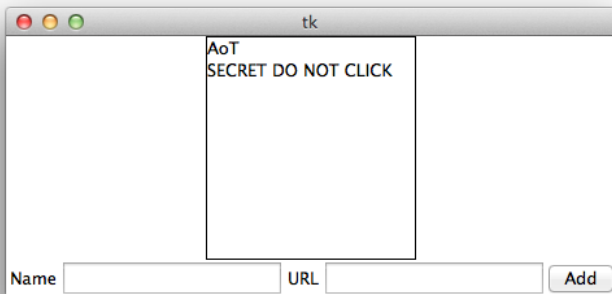


Tk et urllib

# Interface graphique

- Éditer la base de données avec une interface graphique

*Qui va communiquer avec le serveur Bottle*





# Point d'entrée côté serveur

```
1 # La base de données
2 database = {'links': [
3     {'name': 'AoT', 'url': 'http://0.media.dorkly.cvcdn.com
4     /56/67/8187217c79de7ac6ac55e79f80c71232.gif', 'score': 10},
5     {'name': 'SECRET DO NOT CLICK', 'url': 'http://goo.gl/jfc8di',
6     'score': 257}
7 ]}
8
9 # Récupération des liens
10 @route('/links')
11 def links():
12     return json.dumps(database)
13
14 # Ajout d'un nouveau lien
15 @post('/addlink')
16 def addlink():
17     newentry = json.loads(request.body.read().decode('utf-8'))
18     database['links'].append(newentry)
```

# Charger un document via un URL

- **Chargement** d'un URL avec la méthode `urlopen`

*À partir d'un objet `urllib.request`*

- Obtention d'un **dictionnaire** avec `json.loads`

*Il faut avant tout décoder le résultat en UTF-8*

```
1 import urllib.request
2 import json
3
4 url = urllib.request.urlopen("http://localhost:8080/links")
5 rawdata = url.read()
6 data = rawdata.decode('utf-8')
7 database = json.loads(data)
```

# Envoyer des données à un serveur

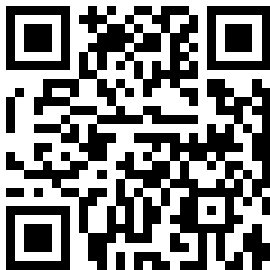
- Construction d'un objet **JSON** avec `json.dumps`

*Il faut encoder les données en UTF-8*

- **Envoi** vers un URL `urlopen`

*À partir d'un objet `urllib.request`*

```
1 import urllib.request
2 import json
3
4 database = {'name': 'kikoo', 'url': 'http://kik.oo', 'score': 0}
5 data = json.dumps(database)
6 rawdata = data.encode('utf-8')
7 urllib.request.urlopen("http://localhost:8080/addlink", rawdata)
```



<http://goo.gl/jfc8di>

# Crédits

- [https://www.flickr.com/photos/fearless\\_craig/3997311986](https://www.flickr.com/photos/fearless_craig/3997311986)
- <https://www.flickr.com/photos/ahotchin/6912923957>