

 eBook Gratuit

APPRENEZ cherry.py

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

#cherry.py

Table des matières

À propos.....	1
Chapitre 1: Commencer avec cherrypy.....	2
Remarques.....	2
Exemples.....	2
Instructions d'installation.....	2
Conditions préalables.....	2
Installer.....	2
Bonjour tout le monde dans CherryPy.....	3
Téléchargement de fichier avec CherryPy.....	3
server.py.....	3
webpage.html.....	4
cli.py.....	5
upload.txt.....	5
Télécharger depuis le navigateur.....	5
Télécharger depuis la ligne de commande.....	5
Crédits.....	6

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [cherry.py](#)

It is an unofficial and free cherry.py ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official cherry.py.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Commencer avec cherrypy

Remarques

Cette section fournit une vue d'ensemble de ce qu'est cherrypy et pourquoi un développeur peut vouloir l'utiliser.

Il devrait également mentionner tous les grands sujets dans cherrypy, et établir un lien avec les sujets connexes. La documentation de cherrypy étant nouvelle, vous devrez peut-être créer des versions initiales de ces rubriques connexes.

Exemples

Instructions d'installation

Conditions préalables

- Ces instructions supposent que vous avez tout type de Linux, Unix, Mac avec bash ou Windows Git-bash.
- Windows: Téléchargez et installez [Git-bash pour Windows](#) , puis exécutez «bash» à partir de la ligne de commande.
- D'autres shells que bash vont bien aussi, remplacez simplement la commande `activate` ci-dessous par `activate.csh` ou Google: "[virtualenv activate your-shell-name](#)".

Avant de commencer, vérifiez que Python, virtualenv et pip sont installés:

- `$ python --version`
- `$ virtualenv --version`
- `$ pip --version`

Installer

Créez un répertoire avec votre application Web /, créez un environnement et installez le package CherryPy.

- `$ mkdir /develop/myapp/`
- `$ cd /develop/myapp/`
- `$ virtualenv venv`
- `$ source venv/bin/activate`
 - **Sous Windows dans Git-bash:** `$ source venv/Scripts/activate`
- `(venv) $ pip install cherrypy`
- `(venv) $ python`

```
Python 3.5.2 ...
>>> import cherrypy
```

```
>>> cherrypy
<module 'cherrypy' from '... venv/site-packages/cherrypy/__init__.py'>
```

Toutes nos félicitations! Vous êtes maintenant prêt pour votre première application CherryPy.

Bonjour tout le monde dans CherryPy

Si vous avez un virtualenv et que CherryPy y est déjà installé, créez un fichier `hello.py` :

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

import cherrypy

class HelloWorld(object):
    @cherrypy.expose
    def index(self):
        return 'Hello World!'

    @cherrypy.expose
    def greet(self, name):
        return 'Hello {}'.format(name)

cherrypy.quickstart(HelloWorld())
```

Ensuite, exécutez le fichier: `$ hello.py` ou `$ python hello.py`. Vous devriez voir une sortie similaire à ceci:

```
user@computer /develop/myapp $ python hello.py
[06/Nov/2016:05:58:44] ENGINE Listening for SIGTERM.
[06/Nov/2016:05:58:44] ENGINE Bus STARTING
[06/Nov/2016:05:58:44] ENGINE Set handler for console events.
CherryPy Checker:
The Application mounted at '' has an empty config.

[06/Nov/2016:05:58:44] ENGINE Started monitor thread '_TimeoutMonitor'.
[06/Nov/2016:05:58:44] ENGINE Started monitor thread 'Autoreloader'.
[06/Nov/2016:05:58:45] ENGINE Serving on http://127.0.0.1:8080
[06/Nov/2016:05:58:45] ENGINE Bus STARTED
```

- Pour voir 'Hello World!' pointez votre navigateur sur <http://localhost:8080/>
- Pour voir un message d'accueil, accédez à <http://localhost:8080/greet?Name=John>

Téléchargement de fichier avec CherryPy

Cet exemple comprend trois parties:

- `server.py` - Application CherryPy qui peut recevoir et enregistrer un fichier.
- `webpage.html` - Exemple de téléchargement d'un fichier sur `server.py` à partir d'une page Web.
- `cli.py` - Exemple de téléchargement d'un fichier sur `server.py` à partir d'un outil de ligne de commande.
- Bonus - `upload.txt` - fichier que vous allez télécharger.

server.py

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

import os
import cherrypy

config = {
    'global' : {
        'server.socket_host' : '127.0.0.1',
        'server.socket_port' : 8080
    }
}

class App:

    @cherrypy.expose
    def upload(self, ufile):
        # Either save the file to the directory where server.py is
        # or save the file to a given path:
        # upload_path = '/path/to/project/data/'
        upload_path = os.path.dirname(__file__)

        # Save the file to a predefined filename
        # or use the filename sent by the client:
        # upload_filename = ufile.filename
        upload_filename = 'saved.txt'

        upload_file = os.path.normpath(
            os.path.join(upload_path, upload_filename))
        size = 0
        with open(upload_file, 'wb') as out:
            while True:
                data = ufile.file.read(8192)
                if not data:
                    break
                out.write(data)
                size += len(data)
        out = ''
        File received.
        Filename: {}
        Length: {}
        Mime-type: {}
        ''' .format(ufile.filename, size, ufile.content_type, data)
        return out

if __name__ == '__main__':
    cherrypy.quickstart(App(), '/', config)
```

webpage.html

```
<form method="post" action="http://127.0.0.1:8080/upload" enctype="multipart/form-data">
```

```
<input type="file" name="ufile" />
<input type="submit" />
</form>
```

cli.py

Cet exemple nécessite un paquet de [requêtes Python](#) , cependant le fichier peut être envoyé au serveur en clair Python.

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

import requests

url = 'http://127.0.0.1:8080/upload'
files = {'ufile': open('file.txt', 'rb')}

r = requests.post(url, files=files)

print(r)
print(r.text)
```

upload.txt

```
Hello! This file was uploaded to CherryPy.
```

Télécharger depuis le navigateur

- Exécuter `$ server.py`
- Ouvrez `webpage.html` dans votre navigateur Web.
- Après avoir sélectionné le fichier de votre lecteur et l'avoir `saved.txt` , il sera enregistré sous le nom `saved.txt` .

Télécharger depuis la ligne de commande

- Ouvrez une console et exécutez `$ server.py`
- Ouvrez une autre console et exécutez `$ cli.py`
 - Note: Le fichier de test `upload.txt` doit être dans le même répertoire que `cli.py`
- Le fichier `upload.txt` doit être téléchargé et enregistré sous le nom `saved.txt` .

Lire Commencer avec cherry en ligne: <https://riptutorial.com/fr/cherrypy/topic/7696/commencer-avec-cherrypy>

Crédits

S. No	Chapitres	Contributeurs
1	Commencer avec cherrypy	Community , dwich