

Déployer en production

Cette partie du tutoriel suppose que vous disposez d'un serveur sur lequel vous souhaitez déployer votre application. Elle donne un aperçu de la manière de créer le fichier de distribution et de l'installer, mais n'entre pas dans les détails concernant le serveur ou le logiciel à utiliser. Vous pouvez configurer un nouvel environnement sur votre ordinateur de développement pour essayer les instructions ci-dessous, mais vous ne devriez probablement pas l'utiliser pour héberger une véritable application publique. Voir [Deployment Options](#) pour une liste des différentes manières d'héberger votre application.

Construction et installation

Lorsque vous voulez déployer votre application ailleurs, vous construisez un fichier de distribution. Le standard actuel pour la distribution Python est le format *wheel*, avec l'extension `.whl`. Assurez-vous que la bibliothèque `wheel` est installée en premier :

```
$ pip install wheel
```

L'exécution de `setup.py` avec Python vous donne un outil de ligne de commande pour lancer des commandes liées à la construction. La commande `bdist_wheel` va construire un fichier de distribution `wheel`.

```
$ python setup.py bdist_wheel
```

Vous pouvez trouver le fichier dans `dist/flaskr-1.0.0-py3-none-any.whl`. Le nom du fichier est au format {nom du projet}-{version}-{balises python} -{balise abi}-{balise plateforme}.

Copiez ce fichier sur une autre machine, [mettez en place un nouveau virtualenv](#), puis installez le fichier avec `pip`.

```
$ pip install flaskr-1.0.0-py3-none-any.whl
```

Pip installera votre projet ainsi que ses dépendances.

Comme il s'agit d'une machine différente, vous devez relancer `init-db` pour créer la base de données dans le dossier de l'instance.

Bash

CMD

Powershell

 v: latest ▼

```
$ export FLASK_APP=flaskr
$ flask init-db
```

Lorsque Flask détecte qu'il est installé (pas en mode modifiable), il utilise un répertoire différent pour le dossier de l'instance. Vous pouvez le trouver dans `venv/var/flaskr-instance` à la place.

Configurer la clé secrète

Au début du tutoriel, vous avez donné une valeur par défaut pour `SECRET_KEY`. Cette valeur doit être remplacée par des octets aléatoires en production. Sinon, les attaquants pourraient utiliser la clé publique `'dev'` pour modifier le cookie de session, ou tout autre élément qui utilise la clé secrète.

Vous pouvez utiliser la commande suivante pour générer une clé secrète aléatoire :

```
$ python -c 'import os; print(os.urandom(16))'

b'_5#y2L"F4Q8z\n\xec]/'
```

Créez le fichier `config.py` dans le dossier de l'instance, que la fabrique lira s'il existe. Copiez la valeur générée dans ce fichier.

```
venv/var/flaskr-instance/config.py
```

```
SECRET_KEY = b'_5#y2L"F4Q8z\n\xec]/'
```


Vous pouvez également définir toute autre configuration nécessaire ici, bien que `SECRET_KEY` soit la seule nécessaire pour Flask.

Exécution avec un serveur de production

Lorsque vous exécutez publiquement plutôt qu'en développement, vous ne devriez pas utiliser le serveur de développement intégré (`flask run`). Le serveur de développement est fourni par Werkzeug pour des raisons pratiques, mais il n'est pas conçu pour être particulièrement efficace, stable ou sécurisé.

A la place, utilisez un serveur WSGI de production. Par exemple, pour utiliser [Waitress](#), installez-le d'abord dans l'environnement virtuel :

```
$ pip install waitress
```

Vous devez informer Waitress de votre application, mais il n'utilise pas `FLASK_APP`  `v: latest` ▼ fait `flask run`. Vous devez lui dire d'importer et d'appeler la fabrique d'applications pour obtenir un objet d'application.

```
$ waitress-serve --call 'flaskr:create_app'
```

Serving on http://0.0.0.0:8080

Voir [Deployment Options](#) pour une liste des différentes manières d’héberger votre application. Waitress est juste un exemple, choisi pour le tutoriel parce qu’il supporte à la fois Windows et Linux. Il existe de nombreux autres serveurs WSGI et options de déploiement que vous pouvez choisir pour votre projet.

Passez à [Continuez à vous développer !](#).