



Kostenloses eBook

LERNEN

virtualenv

Free unaffiliated eBook created from
Stack Overflow contributors.

#virtualenv

Inhaltsverzeichnis

Über	1
Kapitel 1: Erste Schritte mit virtualenv	2
Bemerkungen.....	2
Examples.....	2
Installation oder Setup.....	2
Credits	4



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

It is an unofficial and free virtualenv 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 virtualenv.

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

Kapitel 1: Erste Schritte mit virtualenv

Bemerkungen

In diesem Abschnitt erhalten Sie einen Überblick darüber, was virtualenv ist und warum ein Entwickler es möglicherweise verwenden möchte.

Es sollte auch alle großen Themen innerhalb von Virtualenv erwähnen und auf die verwandten Themen verweisen. Da die Dokumentation für virtualenv neu ist, müssen Sie möglicherweise erste Versionen dieser verwandten Themen erstellen.

Examples

Installation oder Setup

Das Virtual Environment-Tool ([virtualenv](#)) dient zum [Isolieren](#) verschiedener Projekte und ihrer Abhängigkeiten, indem für jedes einzelne eine individuelle Python-Umgebung erstellt wird. Es ist wie das Installieren eines Pakets lokal (und nicht global), ähnlich wie bei der [Option zur Installation des Pakets npm](#) . Im Folgenden finden Sie ein Beispiel zum Installieren und Testen von virtualenv zum Erstellen von zwei Projekten (Project1-A Django-Anwendung und Project2-A Flask-Anwendung):

1. Überprüfen Sie zunächst, ob bereits eine `$ virtualenv --version` installiert ist. `$ virtualenv --version`
2. Führen Sie `$ pip install virtualenv` (für Mac und Linux) oder `$ sudo apt-get install python-virtualenv` für Ubuntu aus, [easy_install](#) für Windows , um die Python-Umgebung zu installieren.
3. `$ mkdir Project1` und `$ cd Project1`
4. Führen Sie `$ virtualenv venvp1` Dadurch wird ein Ordner venvp1 im Project1-Verzeichnis erstellt.
5. Um die Umgebung zu aktivieren, führen Sie den `source venvp1/bin/activate` (bei Linux) und `venvp1\Scripts\activate` (bei Windows) und die Eingabeaufforderung ändert sich zu `(venvp1)Your-Computer:your_project UserName$`
6. Führen Sie `pip install Django` , um Django für project1 zu installieren, und `deactivate` (falls erforderlich), um zur globalen Umgebung zurückzukehren.
7. Wiederholen Sie die Schritte 3 bis 6 für die Flask-Anwendung mit unterschiedlichen Verzeichnissen, Virtualenv-Namen und `pip install Flask` , um Flask zu installieren.

Sobald die obigen Schritte (ohne Fehler) ausgeführt wurden, könnte (möglicherweise und) gleichzeitig zwischen beiden Umgebungen ohne Konflikte gearbeitet werden.

Anmerkungen:

1. Bei [virtualenvwrapper](#) handelt es sich um ein weiteres nützliches Tool, das eine erweiterte Version von virtualenv ist, obwohl der Installationsvorgang für beide nahezu gleich ist.
2. Die `virtualenv` Befehls `virtualenv` mit `--no-site-packages` schließt die global installierten

Pakete aus.

3. Um den aktuellen Status der Umgebung `$ pip freeze > installedpkgp1.txt` führen Sie `$ pip freeze > installedpkgp1.txt` . Diese Textdatei enthält eine Liste der installierten Pakete (einschließlich ihrer Versionen) in der aktuellen Umgebung. Wenn die gleiche Umgebung in einem anderen Ordner (oder einer anderen Maschine) bereitgestellt werden muss, kann durch Ausführen des Befehls `$ pip install -r installedpkgp1.txt` `installpkgp1.txt` dieselbe Umgebung erstellt werden.

4. Nützliche Befehle:

- `lsvirtualenv` - Liste aller Umgebungen
- `cdvirtualenv` - `cdvirtualenv` aktuell aktivierten virtuellen Umgebung
- `cdsitepackages` - wie zuvor, geht aber direkt in das `site-packages` Verzeichnis
- `lssitepackages` - zeigt den Inhalt des `site-packages` Verzeichnisses

Erste Schritte mit virtualenv online lesen: <https://riptutorial.com/de/virtualenv/topic/9733/erste-schritte-mit-virtualenv>

Credits

S. No	Kapitel	Contributors
1	Erste Schritte mit virtualenv	Ayrat , Community , w1n5rx