**Использование виртуальной среды**

**https://python-scripts.com/virtualenv**

pip install virtualenv

Начнем с создания нового каталога, с которым мы будем работать:

mkdir python-virtual-environments && cd python-virtual-environments

Создание новой виртуальной среды внутри каталога:

Python 2:

$ virtualenv env

# Python 3

$ python3 -m venv env

### Что находится в этих папках?

* **bin** – файлы, которые взаимодействуют с виртуальной средой;
* **include** – С-заголовки, компилирующие пакеты Python;
* **lib** – копия версии Python вместе с папкой «site-packages», в которой установлена каждая зависимость.

Далее, у нас есть копии или символические ссылки нескольких различных **инструментов Python**. Эти файлы используются для обеспечения того, чтобы команды и код Python выполнялись в контексте нынешней среды, таким образом, достигается **изоляция** от глобальной среды.

Более интересные сейчас – скрипты **activate** в папке **bin**. Эти скрипты используются для настройки вашей оболочки для использования исполняемого файла среды Python и его сайтовых пакетов по умолчанию

Чтобы использовать эти пакеты (или ресурсы) среды в изоляции, вам нужно «активировать» их. Чтобы сделать это, просто запустите:

source env/bin/activate

\*(зайти в папку и запустить activate**)**

(env) $ deactivate – выключить

#### Установка виртуальной среды для Windows 10

pip3 install virtualenvwrapper-win

Теперь вы можете создать новую виртуальную среду при помощи команды mkvirtualen.

$ mkvirtualenv my\_django\_environment

### **Использование виртуальной среды**

Есть еще несколько полезных команд, которые вам следует знать (в документации по инструменту их гораздо больше, но эти вы будете использовать регулярно):

* deactivate — Выход из текущей виртуальной среды Python
* workon — Список доступных виртуальных сред
* workon name\_of\_environment — Активация конкретной виртуальной среды Python
* rmvirtualenv name\_of\_environment — Удаление конкретной виртуальной среды.