Автоматизация на Python + Selenium + Pytest

***Подготовка виртуального окружения***

Для того, чтобы изолировать проект от каких-либо внешних зависимостей (всяких плагинов и прочего) и сделать его независимым, то необходимо создать виртуальное окружение. Будем использовать «venv», но есть всякие его аналоги («virtualenv», «pipenv» и др.).

Находясь в папке с проектом в терминале прописываем:



где **selenium\_env** – название папки с нашим виртуальным окружением (можем придумать любое, я назвал как «**selenium\_env**»)

Если выдаёт ошибку 

то попробовать через python -m venv selenium\_venv

Теперь в папке с проектом создастся папка «selenium\_env» со всякими скриптами и плагинами, которые будем использовать при работе с конкретным проектом

***Активация и деактивация виртуального окружения***

Находясь в той же директории с проектом в терминале прописать:



Таким образом мы указываем путь к .bat файлу, который активирует нашу среду. Он так и называется – «**activate.bat**» (а можно и просто «**activate**»). После активации мы должны увидеть, что в терминале перед путём к нашей директории с проектом в круглых скобках появилось название нашей среды:

**(selenium\_env) C:\Users\777\Desktop\Проекты\Vitzmarket>**

Значит активация прошла успешно и отныне мы работаем именно с теми сторонними плагинами, которые будут устанавливаться именно в эту папку (selenium\_env). Это нужно делать каждый раз при начале работы с проектом.

Теперь через pip install можем устанавливать разные сторонние плагины именно в папку с нашим окружением.

*Ошибки при обновлении pip? Вот как можно пофиксить*: <https://python-scripts.com/upgrade-pip-windows>

Чтобы деактивировать окружение, достаточно прописать в терминале «**deactivate.bat**» (можно «**deactivate**»). Тогда увидим, что название нашей среды в круглых скобках перед путём к проекту исчезло. Значит деактивация прошла успешно.

***Создание файла requirements.txt***

Этот файл содержит в себе список всех сторонних плагинов, которые используются в проекте. Для того, чтобы создать его, сначала нужно «заморозить» все эти плагины, чтобы питон запомнил весь их список. Для этого пишем в терминале **pip freeze > requirements.txt**

Теперь нам нужно создать этот самый файл. Пишем **pip install requirements.txt**

Готово. Теперь у нас есть файл со всеми зависимостями.

А ЛУЧШЕ ДЕЛАТЬ ТАК:

