**Установка Python3 в Windows**

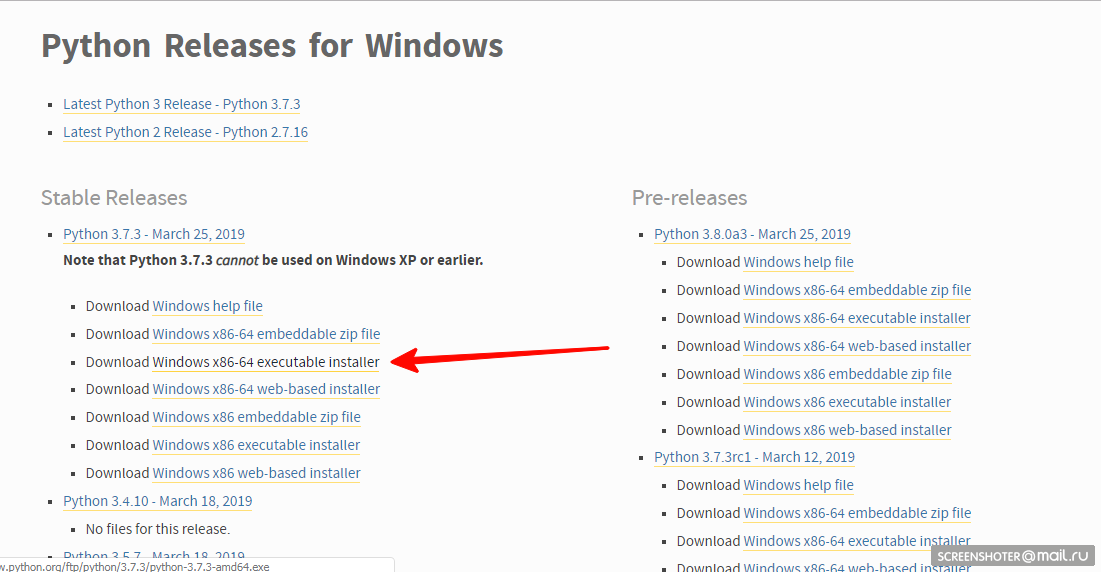
Если вы используете другую ОС, то смело пропускайте этот шаг и переходите к следующему.

**Внимание!** В курсе вы будете работать в командной строке (консоли). Если вы никогда ничего подобного не делали, то не нужно бояться — в этом нет ничего страшного. Можете почитать про некоторые полезные команды по этой ссылке — https://cmd.readthedocs.io/cmd.html.

Для начала вам нужно скачать установочный файл для Python3. Для этого на https://www.python.org/downloads/windows/ выберите стабильную версию для Windows, и скачайте файл для вашей системы (64-разрядная или 32-разрядная).

Если Python у вас уже есть в системе, убедитесь, что установлена версия 3.6 или выше, чтобы вы могли использовать все современные возможности языка. Обновите версию Python, если у вас более старая версия.

**Важно!** Во время установки убедитесь в том, что вы поставили галочку в разделе **Add Python 3.x to PATH**. В этом случае вызов интерпретатора Python будет доступен вам из командной строки. (Подробнее о том, что это за переменная такая PATH можно почитать здесь: <http://barancev.github.io/what-is-path-env-var/>)



Запустите скачанный файл и следуйте инструкциям установщика.

Вот и все, через несколько минут у вас будет установлен интерпретатор Python!

Чтобы проверить правильность установки, откройте командную строку Windows (Пуск - Найти программу - cmd.exe и Запустить). В консоли введите python --version (на некоторых сборках Windows нужно вводить короткую команду, например, py --version):

python --version

Python 3.7.3

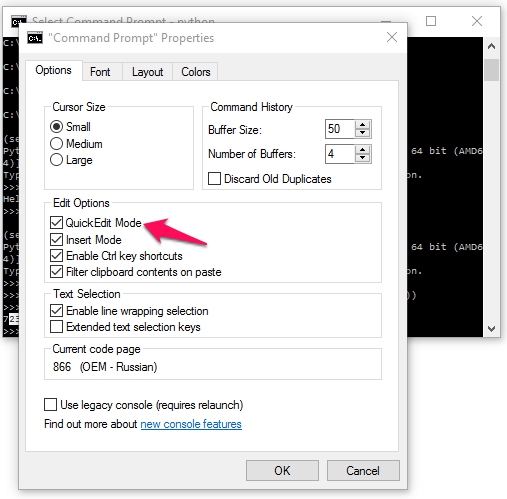
Если Python не найден, то попробуйте переустановить его еще раз или вручную добавить в переменную окружения PATH путь к директории, куда был установлен Python. Как добавлять путь в PATH в разных версиях Windows, вы можете прочитать здесь: <https://www.computerhope.com/issues/ch000549.htm>. После переустановки Python не забудьте перезапустить командную строку, чтобы изменения переменной окружения стали доступны.

Чтобы не засорять наше основное Python-окружение, мы будем работать в виртуальном окружении, которое в дальнейшем можно будет легко удалить или изменить под ваши нужды. Все пакеты для Python, которые мы будем устанавливать в курсе, будут доступны только в этом виртуальном окружении.

**Внимание!**У консоли Windows специфичные отношения с операциями копировать/вставить. Чтобы разрешить выделение текста мышкой, нужно сделать следующие настройки:

* Правой кнопкой нажать на заголовок окна, и выбрать в меню пункт "Настройки" (Settings)
* В настройках поставить галочку "Разрешить выделение мышью" (QuickEdit Mode). Найти настройку вам также поможет скриншот приведенный ниже.
* Сохранить настройки.

Теперь вы сможете выбрать текст в консоли, зажав левую кнопку мыши. После выделения текста и нажатия Enter текст окажется в буфере обмена. Чтобы вставить текст в консоль, нужно нажать правую кнопку мыши.



Создадим папку, где будут храниться наши виртуальные окружения, и перейдем в неё:

mkdir environments

cd environments

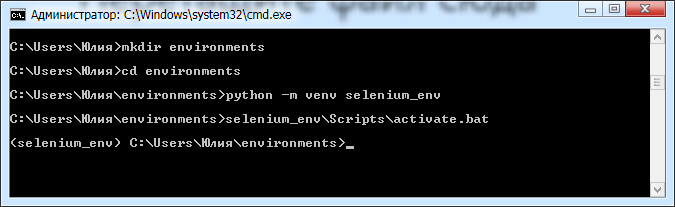
Создадим виртуальное окружение:

python -m venv selenium\_env

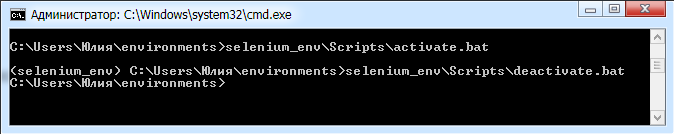
Запустим созданный для нас приложением venv файл activate.bat, чтобы активировать окружение:

selenium\_env\Scripts\activate.bat

Если окружение активировано, то мы увидим в начале командной строки терминала название окружения в круглых скобках. Теперь мы можем устанавливать нужные нам пакеты и запускать скрипты для тестов, которые мы напишем:



Если понадобится выйти из нашего окружения, достаточно выполнить команду deactivate:



Обратите внимание, что после перезапуска терминала необходимо снова активировать нужное окружение.

Проверим, что мы можем теперь использовать Python. Запустим интерпретатор Python и напишем собственную версию HelloWorld:

> selenium\_env\Scripts\activate.bat

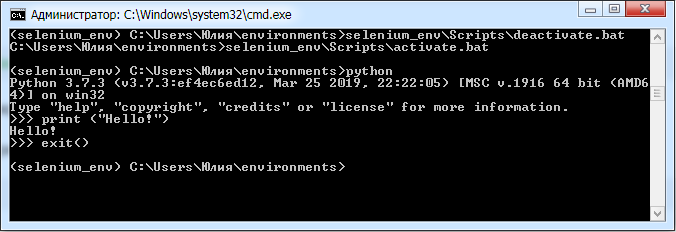
(selenium\_env) С:\Users\Юлия\environments> python

>>> print("Hello, Selenium!")

В итоге вы должны увидеть вывод строки "Hello, Selenium! "

Выйдем из интерпретатора:

>>> exit()



## Установка Selenium для Python

Теперь мы почти готовы запустить браузер с помощью Python и Selenium, и выполнить в нём простые действия. Но предварительно нам еще надо установить библиотеку Selenium для Python и драйвер — программу, которая позволяет взаимодействовать с браузером при помощи Selenium.

В нашем виртуальном окружении установим библиотеку Selenium:

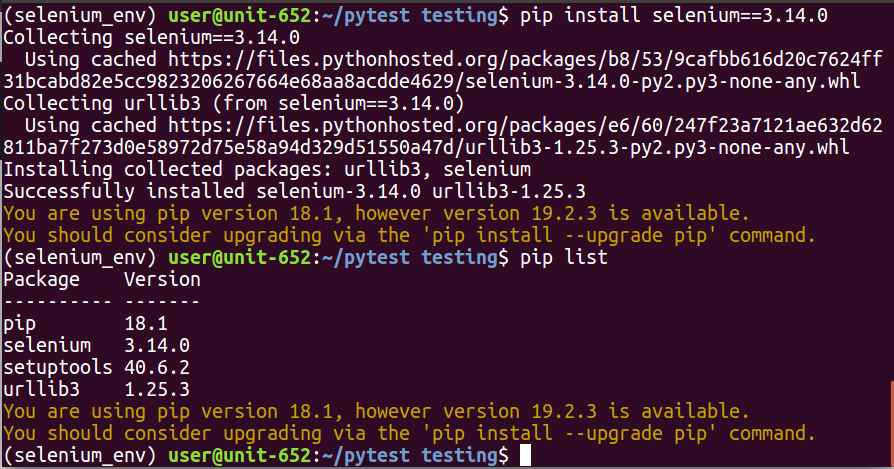
pip install selenium==4.\*

Проект Selenium развивается очень быстро и в вашей установленной версии библиотеки могут быть расхождения в методах или в том, что библиотека выводит в терминал при выполнении скриптов. Если вы просите помощи в комментариях, пожалуйста, указывайте версии python-библиотек, которые вы используете. Версии библиотек в вашем виртуальном окружении можно узнать с помощью команды *pip list*.

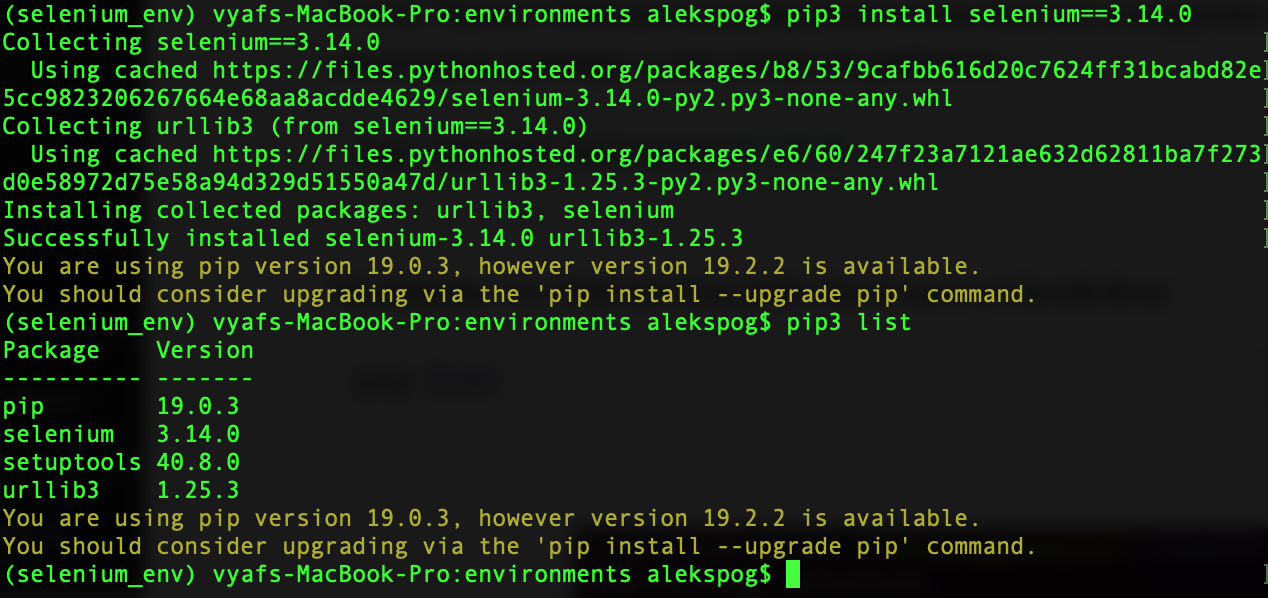
Проверим, что библиотека действительно установлена:

pip list

Ubuntu (скриншот приведён для старой версии библиотеки, но вы должны увидеть похожие логи в терминале):



macOS(скриншот приведён для старой версии библиотеки, но вы должны увидеть похожие логи в терминале):



## Установка драйвера для браузера: Windows

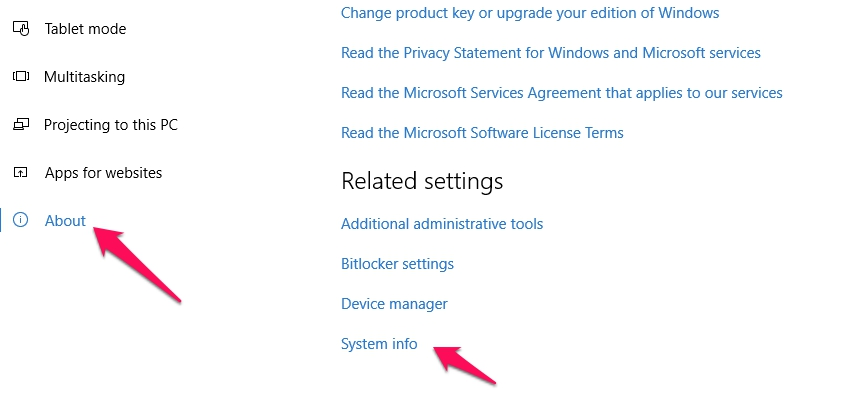
Для установки откройте сайт <https://sites.google.com/chromium.org/driver/> (старая версия сайта <https://sites.google.com/a/chromium.org/chromedriver/downloads>) и скачайте ту версию ChromeDriver, которая соответствует версии вашего браузера. Чтобы узнать версию браузера, откройте новое окно в Chrome, в поисковой строке наберите: chrome://version/ и нажмите Enter. В верхней строчке вы увидите информацию про версию браузера.

* Скачайте с сайта <https://sites.google.com/chromium.org/driver/> (старая версия сайта <https://sites.google.com/a/chromium.org/chromedriver/downloads>) драйвер для вашей версии браузера. Разархивируйте скачанный файл.
* Создайте на диске C: папку chromedriver и положите разархивированный ранее файл chromedriver.exe в папку C:\chromedriver.
* Добавьте в системную переменную PATH папку C:\chromedriver. Как это сделать в разных версиях Windows, описано здесь: <https://www.computerhope.com/issues/ch000549.htm>.

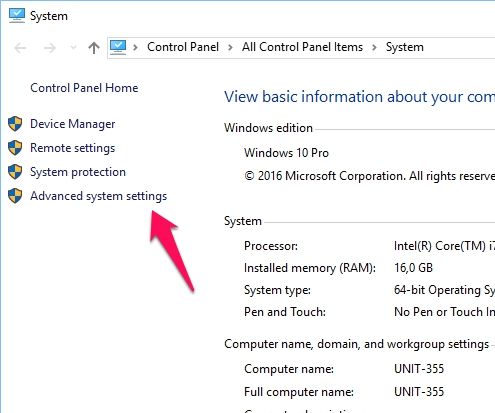
**Пример: как добавить путь в системную переменную PATH на Windows10**

1. Откройте настройки системы.

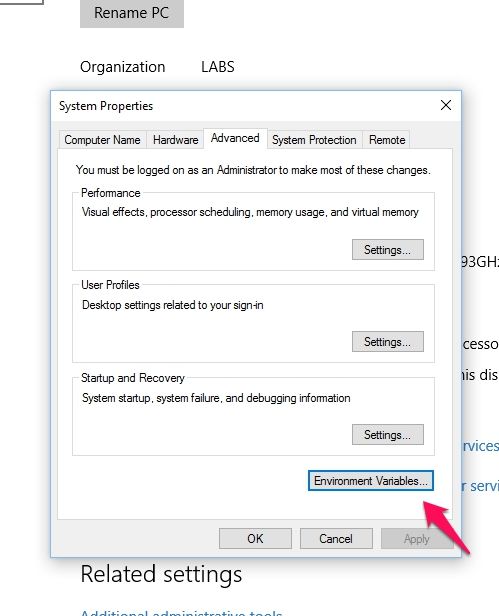
2. В настройках откройте вкладку About, затем System info:



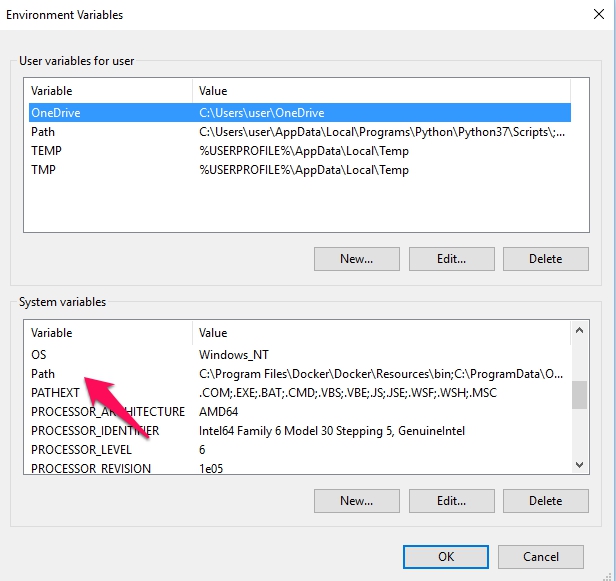
3. Выберите Advanced system settings:



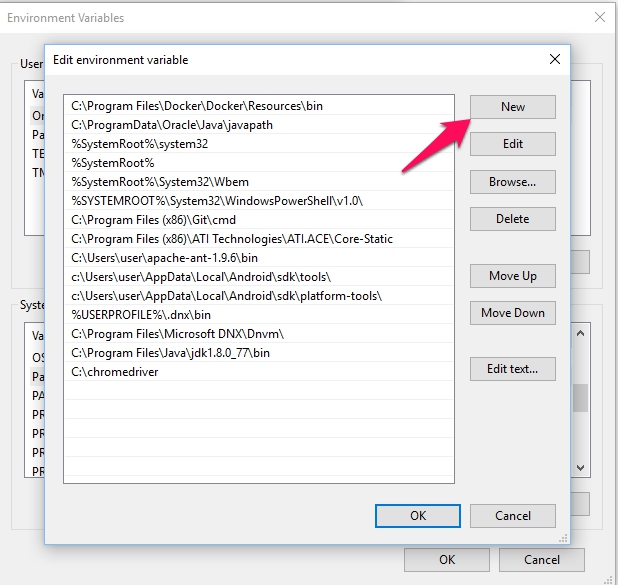
4. Выберите Environment Variables:



5. Кликните два раза на строчке Path в System variables:



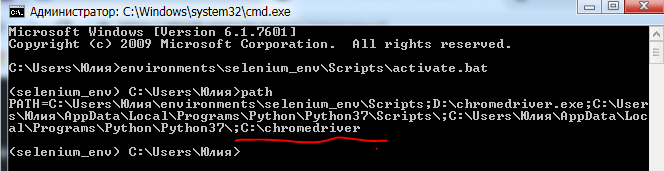
6. Нажмите кнопку New. Введите в новую строку путь к ChromeDriver — C:\chromedriver. Нажмите Enter. У вас должна появится строка с указанным путем:



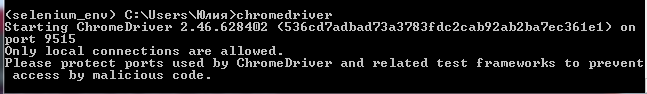
7. Если у вас была открыта командная строка Windows, не забудьте ее закрыть. Затем откройте новую командную строку, чтобы изменения переменной окружения стали доступны. Активируйте снова виртуальное окружение selenium\_env, которое мы создали в предыдущих шагах.

Давайте убедимся в том, что вебдрайвер установлен правильно.

Для начала проверим содержимое переменной path, для этого наберем в командной строке **Path:**



Ура, там есть папка с chromedriver! Попробуем вызвать его напрямую из командной строки:



Магия переменной path: хотя программа chromedriver находится где-то в другом каталоге, мы можем напрямую открывать её, используя имя chromedriver. Чтобы завершить процесс в консоли, нажмите Ctrl+C.

Знак на этом этапе, что пошло что-то не так:



В таком случае попробуйте перезапустить консоль, перезапустить компьютер, перепроверить и добавить заново по инструкциям папку с chromedriver в переменную path.

## Запуск браузера и первый скрипт

### **Для всех ОС:**

В нашем виртуальном окружении запустим интерпретатор python:

~/environments$ python

Теперь напишем скрипт, который откроет простую веб-страницу и заполнит форму. Рекомендуем выполнять команды последовательно по одной, чтобы лучше понять, что происходит в каждой строчке. Не пугайтесь, если не понимаете, что делают те или иные команды, — подробнее мы их разберем в следующих уроках.

Внимание: Файл со скриптом находится в папке “code”, с названием **lesson2\_step11.py**

import time

# webdriver это и есть набор команд для управления браузером

from selenium import webdriver

# импортируем класс By, который позволяет выбрать способ поиска элемента

from selenium.webdriver.common.by import By

# инициализируем драйвер браузера. После этой команды вы должны увидеть новое открытое окно браузера

driver = webdriver.Chrome()

# команда time.sleep устанавливает паузу в 5 секунд, чтобы мы успели увидеть, что происходит в браузере

time.sleep(5)

# Метод get сообщает браузеру, что нужно открыть сайт по указанной ссылке

driver.get("https://suninjuly.github.io/text\_input\_task.html")

time.sleep(5)

# Метод find\_element позволяет найти нужный элемент на сайте, указав путь к нему. Способы поиска элементов мы обсудим позже

# Метод принимает в качестве аргументов способ поиска и значение, по которому мы будем искать

# Ищем поле для ввода текста

textarea = driver.find\_element(By.CSS\_SELECTOR, ".textarea")

# Напишем текст ответа в найденное поле

textarea.send\_keys("get()")

time.sleep(5)

# Найдем кнопку, которая отправляет введенное решение

submit\_button = driver.find\_element(By.CSS\_SELECTOR, ".submit-submission")

# Скажем драйверу, что нужно нажать на кнопку. После этой команды мы должны увидеть сообщение о правильном ответе

submit\_button.click()

time.sleep(5)

# После выполнения всех действий мы должны не забыть закрыть окно браузера

driver.quit()

Вы можете запустить этот же скрипт из файла. Скачайте файл [first\_script.py](https://stepik.org/media/attachments/lesson/25969/first_script.py) к себе в папку и запустите его. Для этого создадим папку selenium\_course и скопируем в нее скачанный файл. Проверим линк (он должен быть таким, как в уроке). Затем запустим файл в нашем виртуальном окружении.  
(может появиться предупреждение, но пока можно на него не обращать внимание)

**Пример для Windows**

(Вместо user везде подставляйте имя пользователя в системе)

При выполнении следующих команд не забудьте изменить путь к вашей пользовательской папке вместо C:\Users\user и к папке с загруженным файлом.

Выполните следующие команды в командной строке Windows:

c:\Users\user> mkdir selenium\_course

C:\Users\user> copy C:\Users\user\Downloads\first\_script.py c:\Users\user\selenium\_course

C:\Users\user> environments\selenium\_env\Scripts\activate.bat

(selenium\_env) C:\Users\user>python c:\Users\user\selenium\_course\first\_script.py

**Пример для Ubuntu и macOS**

Выполните следующие команды в консоли:

mkdir ~/selenium\_course

mv ~/Downloads/first\_script.py ~/selenium\_course

python ~/selenium\_course/first\_script.py