

Автор: Федор Андреевич Лянгузов, [<Fedor.Lyanguzov@ya.ru>](mailto:Fedor.Lyanguzov@ya.ru).

Дата: апрель 2017.

Преамбула

В этом документе представлен один из способов установки интерпретатора языка программирования `Python` на домашний компьютер с использованием прав администратора. Способ будет хорошо работать с `Python 3.6`. Все вопросы направляйте на электронную почту.

Необходимые условия:

- Компьютер
- Свободное место на жестком диске: не менее 250 мегабайт
- Операционная система: Windows 7 или новее
- Права администратора
- Доступ в интернет
- Браузер

Установка Python

Вопрос: Для всех пользователей или для текущего пользователя?

`Python` может быть установлен тремя различными способами: для текущего пользователя, для всех пользователей компьютера, без полной установки. Какой способ подойдет для разработки на домашнем компьютере?

Различия способов установки

Критерий	Для текущего пользователя	Для всех пользователей	Без полной установки
Права администратора	нет	да	нет
Работает у других пользователей этого компьютера?	нет	да	с трудом
Путь установки	<code>C:\Users\%user%\AppData\Local\Programs\Python\Python36-*</code>	<code>C:\Program Files\Python36</code> или <code>C:\Program Files(x86)\Python36</code>	куда угодно, хоть на флешку
Запуск <code>Python</code> из командной строки	да	да	с трудом
Запуск <code>pip</code> из командной строки	с трудом	да	врядли
Применение	Написание программ, некоторые функции труднодоступны	Написание программ, все функции легкодоступны	Распространение программ, требует отдельного описания

Примечание: `pip` - программа для установки дополнительных пакетов(программ или библиотек) для `Python`.

Можно сделать вывод, что для наибольшего удобства разработки следует использовать второй вариант установки. Третий вариант будет интересен в момент *релиза* - выпуска программы в общий доступ и распространения. Однако к нему стоит обратиться гораздо позже, так как он значительно сложнее в настройке. Стоит также отметить, что существуют программы, облегчающие релиз.

Шаг 1: Скачать Python

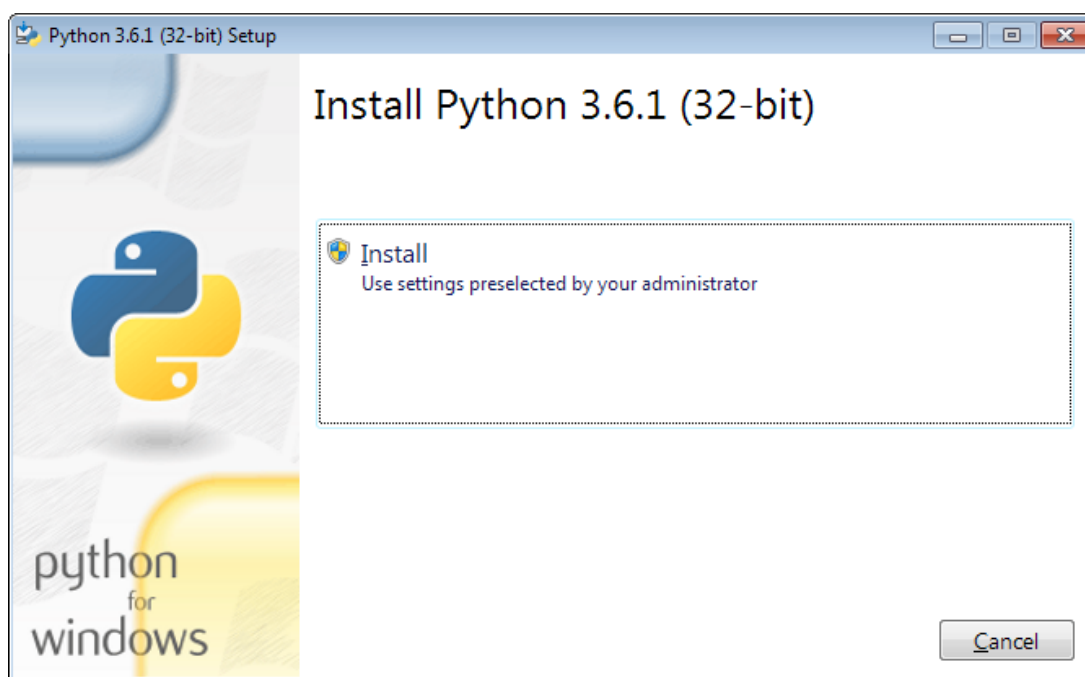
Скачать `Python 3.6` можно по ссылке <https://www.python.org/ftp/python/3.6.1/python-3.6.1.exe>.

Шаг 2: Скачать unattend.xml

`unattend.xml` - файл с настройками установки, специально подготовленный для этой инструкции. Его нужно положить в ту же папку, что и скачанный установщик. Скачать его можно по ссылке https://drive.google.com/open?id=0B_UDbco4kunlU1lVQlh2V3NwNVE.

Шаг 3: Запустить установку

1. Запустить установочный файл.
2. Разрешить ему запускаться, даже если он был скачан из интернета.
3. Проверить, что внешний вид соответствует картинке:



Скриншот установщика

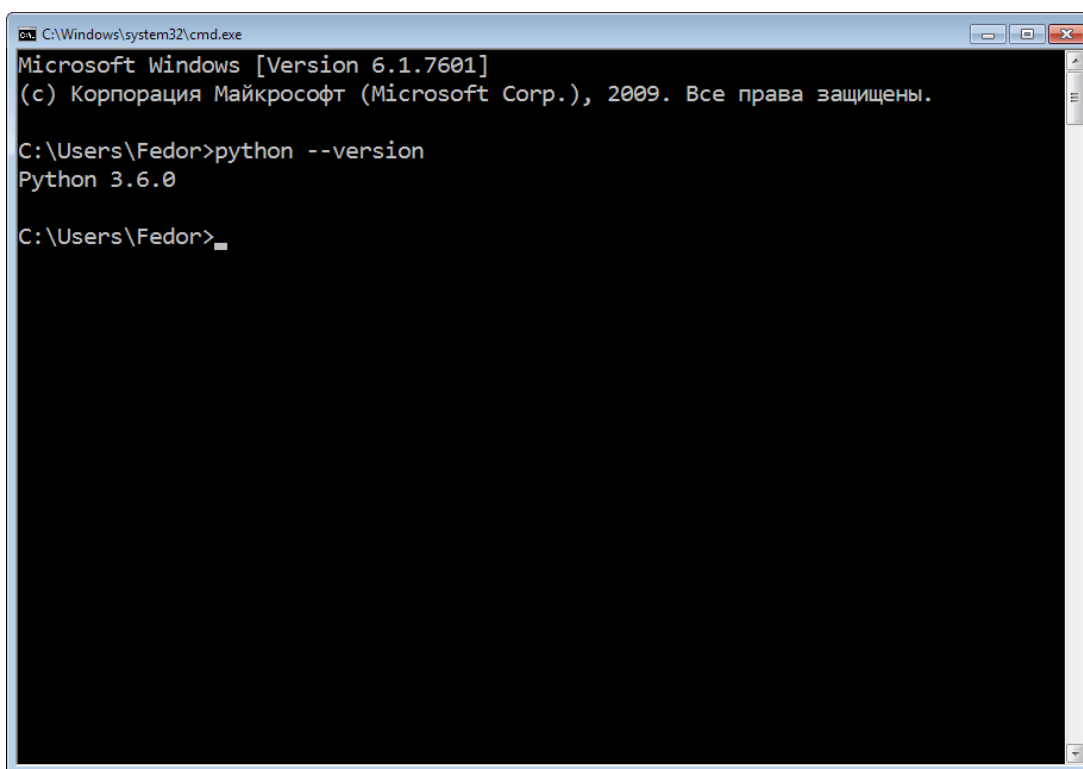
4. Нажать кнопку `Install`.
5. Разрешить установщику работать с правами администратора.
6. Дождаться завершения, закрыть окно установщика.

Проверка установки

После установки чего угодно обязательно нужно проверять корректность установки.

Шаг 1: Проверка установки Python

1. Нажать `Win + R`.
2. В появившемся окне набрать `cmd` и нажать `Enter`.
3. В окне *командной строки* набрать `python --version` и нажать `Enter`. Таким образом будет дана команда `python` с аргументом `--version`. Результат должен быть примерно таким:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Fedor>python --version
Python 3.6.0

C:\Users\Fedor>
```

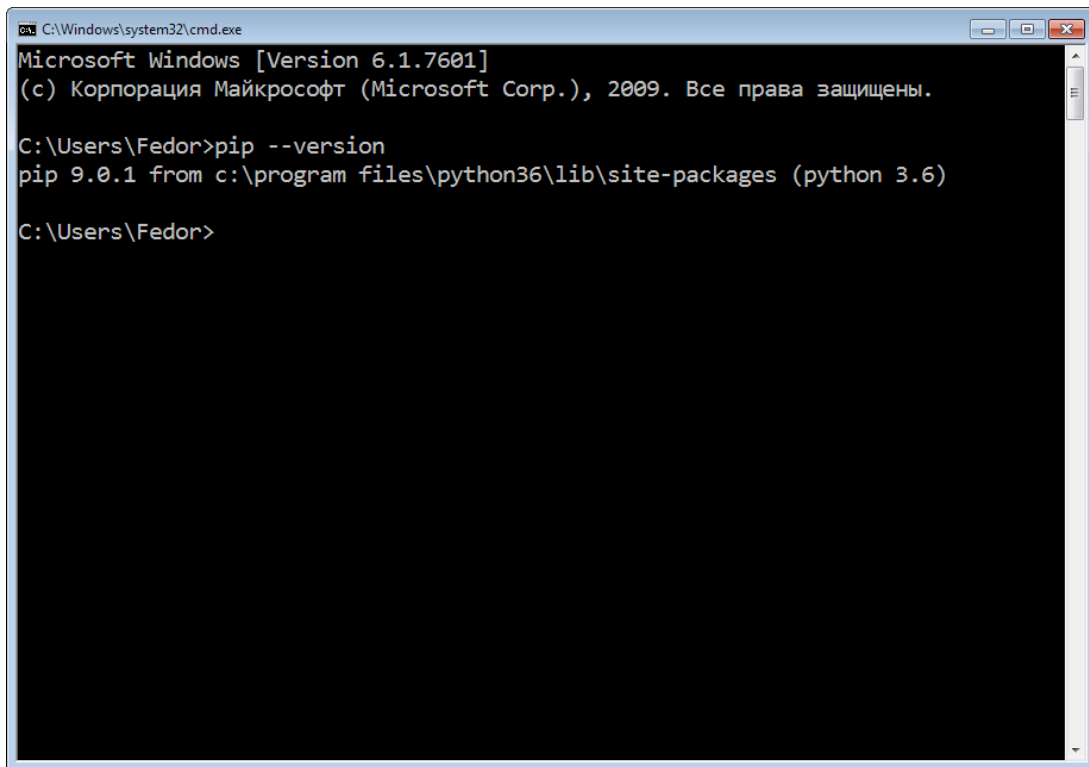
Результат проверки Python

4. Дать команду `exit` или закрыть окно крестиком.

Примечание: Результатом работы команды является весь текст(или отсутствие его) между запросами команд(`C:\Users\Fedor>` на картинке).

Шаг 2: Проверка работы pip

1. Нажать `Win + R`.
2. В появившемся окне набрать `cmd` и нажать `Enter`.
3. Дать команду `pip --version`. Результат должен быть примерно таким:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Fedor>pip --version
pip 9.0.1 from c:\program files\python36\lib\site-packages (python 3.6)

C:\Users\Fedor>
```

Результат проверки pip

4. Дать команду `exit` или закрыть окно крестиком.

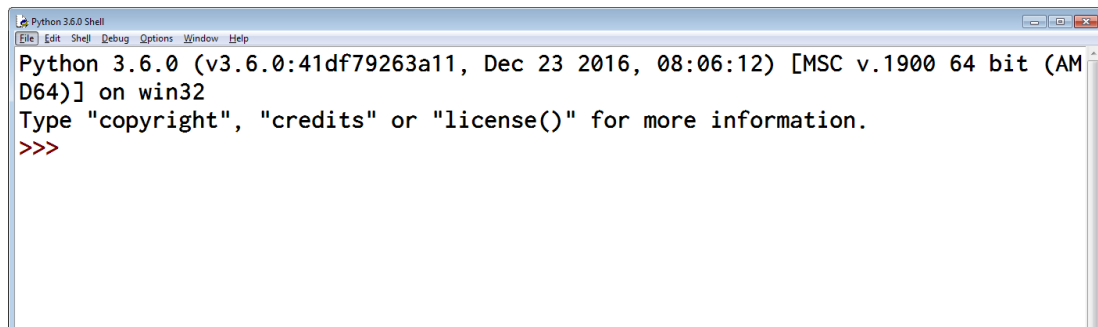
Шаг 3: Проверка работы IDLE

`IDLE` - интегрированная среда разработки(IDE) на языке `Python`, написанная на `Python` с использованием библиотеки `tkInter`. Отличается от обычного блокнота следующими возможностями:

- Встроенный интерпретатор
- Запуск программ
- Подсветка синтаксиса
- Управление отступами
- Автодополнение кода
- Подсказки по коду
- Создание точек остановки(`breakpoint`) для отладки кода
- Изменение размера шрифта

Как проверить корректность установки:

1. Открыть меню "Пуск".
2. В поиске набрать `IDLE`.
3. Нажать `Enter`.
4. Откроется окно *командного режима* IDLE:



Окно программы IDLE

5. В этом режиме можно давать команды на языке `Python`. Для выхода дать команду `exit()` или закрыть программу крестиком.

Удаление Python

Надо не только уметь устанавливать, но и удалять программы корректно. Чтобы удалить `Python`, `pip`, `IDLE` и все дополнительные пакеты нужно:

1. Открыть "Панель управления".
2. Перейти в раздел "Программы и компоненты".
3. Найти `Python 3.6.1 (32-bit)`, нажать `Удалить/Изменить`, в появившемся меню выбрать `Uninstall`, далее следовать инструкциям на экране.
4. В разделе "Программы и компоненты" найти `Python Launcher`, нажать `Удалить`, следовать инструкциям.

Примечание: для полного удаления *обязательно* удалять `Python Launcher`.

Примечание: удаление `Python` удалит также все дополнительные пакеты, установленные для этой версии языка.

Установка дополнительных модулей из интернета

На сайте <https://pypi.python.org/pypi> доступны для загрузки более 100 000 пакетов с открытым исходным кодом на языке `Python`. Они являются не только прекрасным материалом для самостоятельного изучения и усиления навыка программирования, но и хорошим подспорьем при создании своих собственных проектов практически в любой сфере. Например:

- `Panda3d` - *фреймворк*, набор *компонентов*(больших, чем функции) и *утилит*(маленьких программ) для построения трехмерных игр и симуляций.
- `Cocos2d` - высокоуровневый фреймворк для построения двумерных игр.
- `pygame` - низкоуровневый фреймворк для построения двумерных игр.
- `django` - *огромный*, как сам `Python`, фреймворк для построения сайтов.
- `flask` - в противоположность, маленький и легкий фреймворк для построения сайтов.
- `selenium` - фреймворк для автоматизации работы в интернете.
- `cryptography` - *библиотека*(набор функций) для шифрования.
- `Pillow` - библиотека для автоматизации работы с изображениями.
- `Sphinx` - фреймворк для построения сайтов с документацией(да, да, знаю, документация не нужна).
- `oauthlib` - библиотека для авторизации в интернете, чтобы затем обращаться к сайтам.
- `google-api-python-client` - библиотека для работы с Google-сервисами.
- `scipy` - *громадная* библиотека для физических расчетов(нахождение интегралов, решение дифференциальных уравнений).
- `matplotlib` - библиотека для рисования графиков(даже анимированных...)
- `poliastro` - библиотка для построения орбит тел(планет, спутников естественных и искусственных) в солнечной системе с использованием данных из NASA.
- `PyQt5` - *привязка*(библиотека для `Python`, использующая библиотеку на другом языке, или интернет сервис) к самому популярному открытому фреймворку для построения пользовательского интерфейса QT.
- `vk` - привязка к vk.com.
- и многие другие...

Вопрос: Для всех пользователей или для текущего пользователя?

Установив `Python` для всех пользователей может показаться, что устанавливать дополнительные пакеты тоже нужно устанавливать для всех пользователей(и с правами администратора). Однако это не так. `Python` и его стандартные библиотеки проходят строгий контроль качества перед релизом. С большой вероятностью в них не содержится критических ошибок, открывающих уязвимости в операционной системе. Даже если такие ошибки и попадают, они очень быстро исправляются. Этого нельзя сказать о пакетах из `PyPI`. Никто, кроме авторов пакета, не несет ответственности как за работоспособность, так и за безопасность пакета. Недобросовестные авторы могут встраивать вредоносный код в установщик пакета, заражая операционную систему во время установки с правами администратора. Для заражения практически всегда необходимы права администратора, поэтому установка пакетов *без прав администратора* является хорошей превентивной мерой для обеспечения безопасности компьютера. Поэтому в этом мануале нет инструкций для

установки пакетов с правами администратора. Единственный способ полностью обезопасить себя - прочитать исходный код пакета. Это возможно, но достаточно трудоемко.

Установка пакета из PyPI

1. Открыть командную строку(`cmd`).
2. Дать команду `pip install --user имя_пакета`.
3. Убедиться, что в результате команды нет красных строчек и результат заканчивается на `Successfully installed имя_пакета`.
4. Если это не так:
 1. Убедиться в наличии доступа к интернету.
 2. Убедиться в правильности написания имени пакета.
 3. Убедиться в совместимости вашей версии `Python` с пакетом на сайте `PyPI`.
 4. Попробовать найти другой пакет со схожей функциональностью.
 5. Подумать, стоит ли пробовать устанавливать пакет от имени администратора(практически всегда *нет*).
 6. Установить пакет от имени администратора.
 7. Если все это не помогло - плакать.

Примечание: если забыть написать аргумент `--user`, ничего страшного не случится, пакет просто не установится. Следует продолжить установку с пункта 2.

Проверка корректности установки пакета

1. Открыть командную строку.
2. Дать команду `python -c "import имя_пакета"`.
3. Убедиться в отсутствии слов `Error` и `Exception` в результате команды. Пустой результат с высокой вероятностью означает корректную установку.

Удаление пакета

1. Открыть командную строку.
2. Дать команду `pip uninstall имя_пакета`.
3. Подтвердить удаление.
4. Убедиться, что в конце результата команды есть `Successfully uninstalled имя_пакета`.