**<https://docs.microsoft.com/ru-ru/windows/python/beginners>**

**Начало работы с Python в Windows для начинающих**

* + [](https://github.com/olprod)

Это пошаговое руководство для начинающих работу с Python в Windows 10.

**Настройка среды разработки**

Для начинающих, которые не знакомы с Python, рекомендуется [установить Python из Microsoft Store](https://www.microsoft.com/p/python-37/9nj46sx7x90p?activetab=pivot:overviewtab). При установке из Microsoft Store используется базовый интерпретатор Python3, но в дополнение к автоматическому обновлению также настраиваются параметры пути для текущего пользователя (без необходимости доступа администратора). Это особенно полезно, если вы работаете из среды образовательного учреждения или являетесь частью организации, которая ограничивает разрешения или административный доступ на компьютере.

Если вы используете Python в Windows для **разработки веб-приложений**, мы рекомендуем настроить среду разработки другим образом. Вместо установки непосредственно в Windows рекомендуется установить и использовать Python через подсистему Windows для Linux. Справочные сведения см. в следующих статьях: [Начало работы с Python для разработки веб-приложений в Windows](https://docs.microsoft.com/ru-ru/windows/python/web-frameworks). Если вы заинтересованы в автоматизации общих задач в операционной системе, ознакомьтесь с нашим руководством: [Начало работы с Python в Windows для создания сценариев и автоматизации](https://docs.microsoft.com/ru-ru/windows/python/scripting) В некоторых сложных сценариях (например, при необходимости модификации или доступа к установленным файлам Python, создания копий двоичных файлов или непосредственного использования библиотек DLL Python) может потребоваться загрузить определенный выпуск Python непосредственно с сайта [python.org](https://www.python.org/downloads/) или установить [альтернативное средство](https://www.python.org/download/alternatives), например Anaconda, Jython, PyPy, WinPython, IronPython и т. д. Мы рекомендуем это только в том случае, если вы более продвинутый программист на Python и у вас есть конкретная причина выбрать альтернативную реализацию.

**Установка Python**

Чтобы установить Python с помощью Microsoft Store, сделайте следующее:

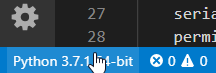
1. Перейдите в меню **Пуск** (значок Windows в нижнем левом углу), введите "Microsoft Store" и щелкните ссылку, чтобы открыть магазин.
2. Когда магазин откроется, выберите **Поиск** в верхнем правом меню и введите "Python". Выберите "Python 3.9" из результатов в разделе приложений. Щелкните **Получить**.
3. После того как Python завершит процесс загрузки и установки, откройте Windows PowerShell, используя меню **Пуск** (значок Windows в нижнем левом углу). После открытия PowerShell введите Python --version, чтобы убедиться, что Python 3 установлен на компьютере.
4. Установка Python из Microsoft Store содержит стандартный диспетчер пакетов **pip**. Pip позволяет устанавливать дополнительные пакеты, которые не входят в стандартную библиотеку Python, и управлять ими. Чтобы убедиться, что у вас есть pip, который можно использовать для установки пакетов и управления ими, введите pip --version.

**Установка Visual Studio Code**

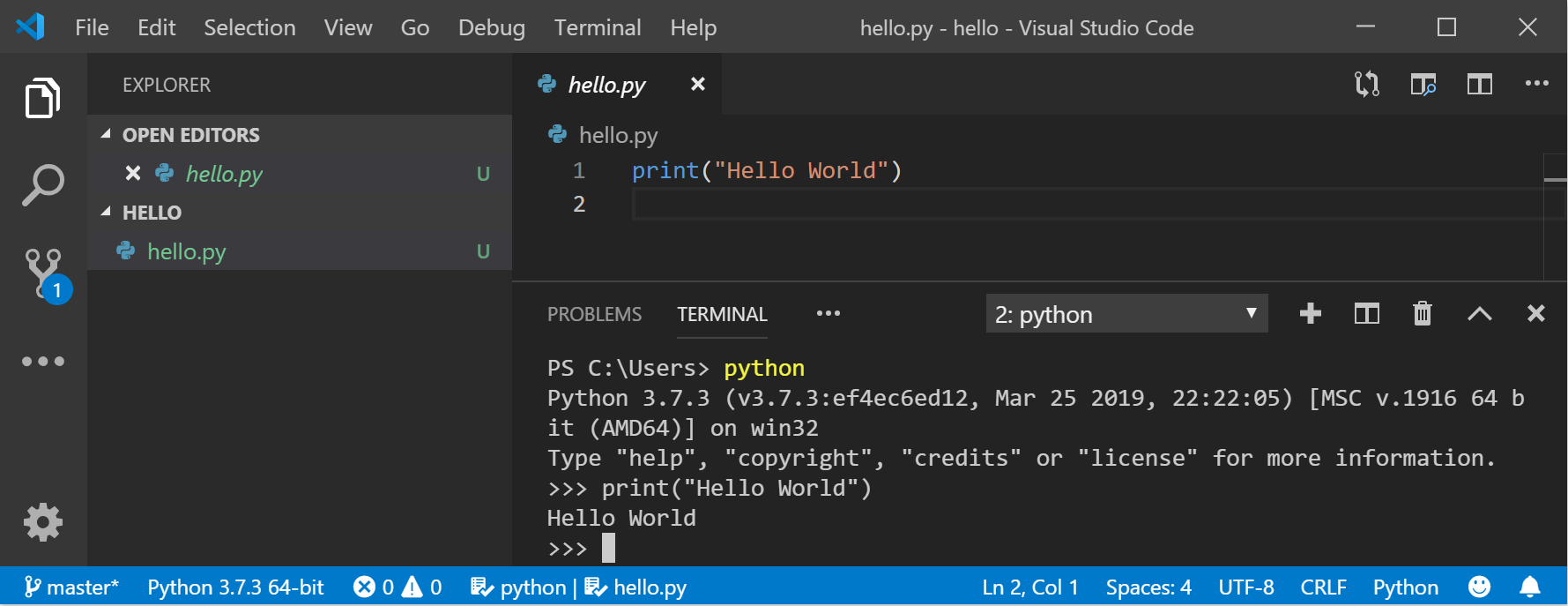
При использовании VS Code в качестве текстового редактора или интегрированной среды разработки (IDE) вам доступны [IntelliSense](https://code.visualstudio.com/docs/editor/intellisense) (помощь в завершении кода), [анализ кода](https://code.visualstudio.com/docs/python/linting) (помогает избежать ошибок в коде), [поддержка отладки](https://code.visualstudio.com/docs/python/debugging) (помогает находить ошибки в коде после запуска), [фрагменты кода](https://code.visualstudio.com/docs/editor/userdefinedsnippets) (шаблоны для небольших повторно используемых блоков кода) и [модульное тестирование](https://code.visualstudio.com/docs/python/unit-testing) (тестирование интерфейса кода с различными типами входных данных).

VS Code также содержит [встроенный терминал](https://code.visualstudio.com/docs/editor/integrated-terminal), который позволяет открывать командную строку Python с помощью командной строки Windows, PowerShell или любой другой, создавая простой рабочий процесс между редактором кода и командной строкой.

1. Чтобы установить VS Code, скачайте VS Code для Windows: [https://code.visualstudio.com](https://code.visualstudio.com/).
2. Установив VS Code, необходимо также установить расширение Python. Для установки расширения Python можно выбрать ссылку на [VS Code в Marketplace](https://marketplace.visualstudio.com/items?itemName=ms-python.python) или открыть VS Code и выполнить поиск по фразе **Python** в меню расширений (Ctrl+Shift+X).
3. Python — интерпретируемый язык, и для выполнения кода Python необходимо указать VS Code, какой интерпретатор нужно использовать. Мы советуем использовать Python 3.7, если только у вас нет конкретной причины для выбора другой программы. После установки расширения Python выберите интерпретатор Python 3, открыв **палитру команд** (CTRL+SHIFT+P), и начните вводить команду **Python: Select Interpreter** (Python: выбор интерпретатора) для поиска, а затем выберите появившуюся команду. Вы также можете использовать параметр **Select Python Environment** (Выбрать среду Python) в нижней строке состояния, если она доступна (возможно, уже отображается выбранный интерпретатор). Команда предоставляет список доступных интерпретаторов, которые VS Code может найти автоматически, включая виртуальные среды. Если нужный интерпретатор не отображается, перейдите к статье о [настройке сред Python](https://code.visualstudio.com/docs/python/environments).



1. Чтобы открыть терминал в VS Code, выберите **Просмотр** > **Терминал** или используйте клавиши **CTRL+`** (символ обратного апострофа). Терминалом по умолчанию является PowerShell.
2. В окне терминала VS Code откройте Python, просто введя команду: python
3. Попробуйте использовать интерпретатор Python, введя: print("Hello World"). Python вернет фразу "Hello World".



**Установка Git (необязательно)**

Если вы планируете совместно работать над кодом Python с другими пользователями или размещать проект на сайте с открытым исходным кодом (например, GitHub), примите во внимание, что VS Code поддерживает [управление версиями с помощью Git](https://code.visualstudio.com/docs/editor/versioncontrol#_git-support). Вкладка системы управления версиями в VS Code отслеживает все изменения и содержит общие команды Git (добавление, фиксация, принудительная отправка, извлечение) прямо в пользовательском интерфейсе. Сначала необходимо установить Git для включения панели управления версиями.

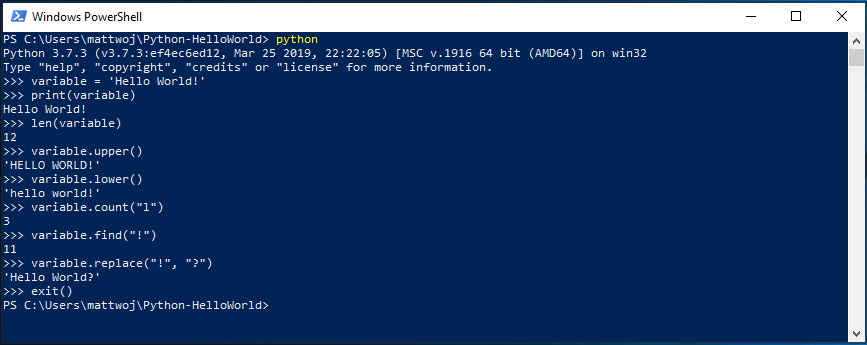
1. Скачайте и установите Git для Windows с [веб-сайта git-scm](https://git-scm.com/download/win).
2. В комплект входит мастер установки, который задает вам ряд вопросов о параметрах установки Git. Рекомендуется использовать все параметры по умолчанию, если у вас нет конкретной причины изменить какой-либо из них.
3. Если вы никогда не использовали Git, обратитесь к [руководствам по GitHub](https://guides.github.com/). Они помогут вам приступить к работе.

**Учебник по некоторым основам работы с Python (на примере Hello World)**

Python, согласно его создателю Гвидо ван Россуму, — это "язык программирования высокого уровня, и его основная философия проектирования — это удобочитаемость кода и синтаксис, позволяющий программистам выразить концепции в нескольких строках кода".

Python — интерпретируемый язык. В отличие от скомпилированных языков, в которых написанный код необходимо перевести в машинный код для выполнения процессором компьютера, код Python передается непосредственно интерпретатору и запускается напрямую. Просто введите код и запустите его. Попробуем сделать это!

1. Откройте командную строку PowerShell и введите python, чтобы запустить интерпретатор Python 3. (В некоторых инструкциях указано использовать команду py или python3, которые также подойдут.) Если вы делаете все правильно, появится командная строка с тремя символами "больше, чем" (>>>).
2. Существует несколько встроенных методов, позволяющих вносить изменения в строки в Python. Создайте переменную с помощью команды variable = 'Hello World!'. Нажмите клавишу ВВОД для создания новой строки.
3. Выведите переменную с помощью команды print(variable). Отобразится текст "Hello World!".
4. Выясните, сколько символов используется для переменной строки, с помощью команды len(variable). Будет показано, что используется 12 символов. (Обратите внимание, что пробел учитывается как символ в общей длине.)
5. Преобразуйте строковую переменную в буквы верхнего регистра: variable.upper(). Теперь преобразуйте строковую переменную в буквы нижнего регистра: variable.lower().
6. Подсчитайте, сколько раз буква "l" используется в строковой переменной: variable.count("l").
7. Найдите определенный символ в вашей строковой переменной. Давайте найдем восклицательный знак с помощью команды variable.find("!"). Будет показано, что восклицательный знак находится в позиции 11 строки.
8. Замените восклицательный знак на вопросительный знак: variable.replace("!", "?").
9. Чтобы выйти из Python, введите exit(), quit() или нажмите клавиши CTRL+Z.



Надеемся, вам понравилось использовать некоторые из встроенных в Python методов модификации строк. Теперь попробуйте создать файл программы Python и запустить его с помощью VS Code.

**Учебник по использованию Python с VS Code (на примере Hello World)**

Команда VS Code составила отличный учебник по [началу работы с Python](https://code.visualstudio.com/docs/python/python-tutorial#_start-vs-code-in-a-project-workspace-folder) с пошаговым руководством по созданию программы Hello World с помощью Python, запуску программного файла, настройке и запуску отладчика, а также установке пакетов, таких как *matplotlib* и *numpy*, для создания графического изображения в виртуальной среде.

1. Откройте PowerShell и создайте пустую папку с именем hello, перейдите в эту папку и откройте ее в VS Code:

КонсольКопировать

mkdir hello

cd hello

code .

1. После открытия среды VS Code, где показана новая папка *hello* в левом окне **обозревателя**, откройте окно командной строки в нижней панели VS Code, нажав **CTRL+`** (символ обратного апострофа) или выбрав **Просмотр** > **Терминал**. После запуска VS Code в папке эта папка станет вашей рабочей областью. VS Code хранит параметры, относящиеся к этой рабочей области, в файле .vscode/settings.json. Они отделены от параметров пользователя, которые хранятся глобально.
2. Продолжайте работу с учебником в документации для VS Code: [Сведения о создании файла исходного кода для Hello World на Python](https://code.visualstudio.com/docs/python/python-tutorial#_create-a-python-hello-world-source-code-file).

**Создание простой игры с помощью Pygame**



Pygame — это популярный пакет Python для создания игр, который позволяет учащимся изучать программирование увлекательным способом. Pygame отображает графические изображения в новом окне, поэтому метод "только командной строки" WSL не подойдет. Но если вы установили Python с помощью Microsoft Store, как описано в этом учебнике, все получится.

1. После установки Python установите Pygame из командной строки (или терминала в VS Code), введя python -m pip install -U pygame --user.
2. Протестируйте установку, запустив пример игры: python -m pygame.examples.aliens
3. Если все в порядке, откроется окно игры. По завершении игры закройте окно.

Вот как начать написание кода собственной игры:

1. Откройте PowerShell (или командную строку Windows) и создайте пустую папку с именем bounce. Перейдите к этой папке и создайте файл с именем bounce.py. Откройте папку в VS Code:

PowerShellКопировать

mkdir bounce

cd bounce

new-item bounce.py

code .

1. С помощью VS Code введите следующий код Python (или скопируйте и вставьте его):

PythonКопировать

import sys, pygame

pygame.init()

size = width, height = 640, 480

dx = 1

dy = 1

x= 163

y = 120

black = (0,0,0)

white = (255,255,255)

screen = pygame.display.set\_mode(size)

while 1:

for event in pygame.event.get():

if event.type == pygame.QUIT: sys.exit()

x += dx

y += dy

if x < 0 or x > width:

dx = -dx

if y < 0 or y > height:

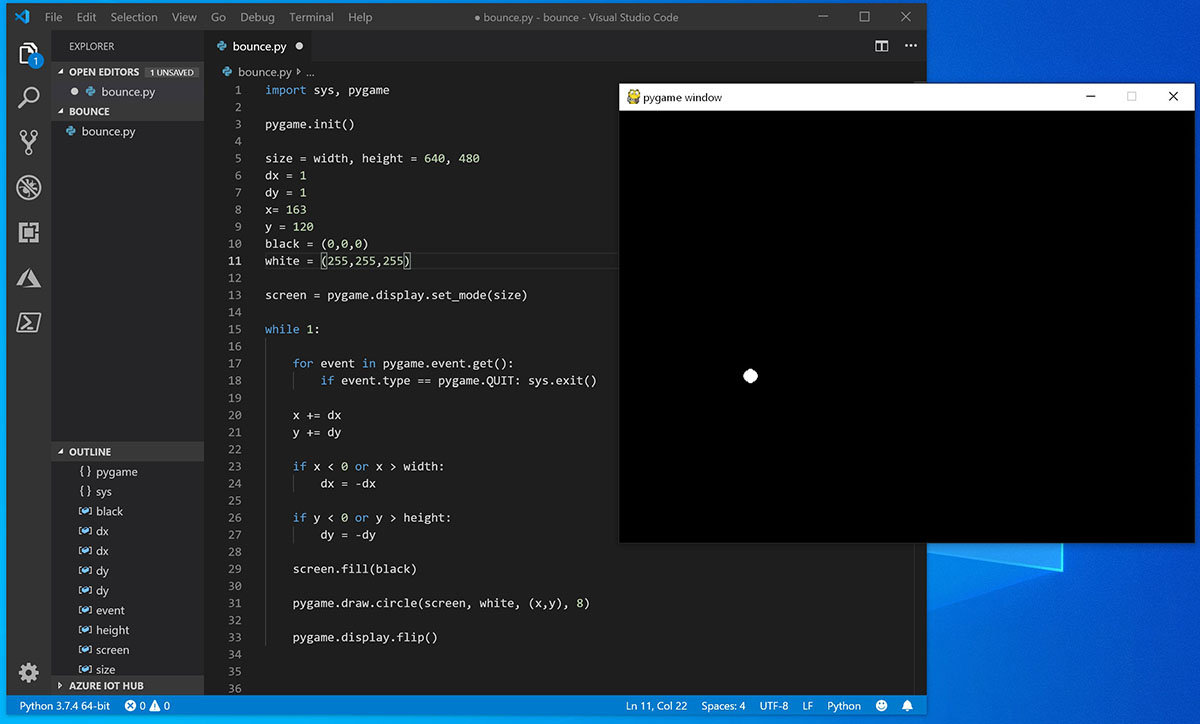
dy = -dy

screen.fill(black)

pygame.draw.circle(screen, white, (x,y), 8)

pygame.display.flip()

1. Сохраните его как: bounce.py.
2. Запустите его в терминале PowerShell, введя: python bounce.py.



Попробуйте изменить некоторые из чисел, чтобы увидеть, как они влияют на прыгающий шарик.

Дополнительные сведения о создании игр с помощью Pygame см. на сайте [pygame.org](http://www.pygame.org/).

**Материалы для непрерывного обучения**

Мы рекомендуем использовать следующие ресурсы, чтобы продолжить изучение разработки на Python в Windows.

**Онлайн-курсы для изучения Python**

* [Введение в Python на Microsoft Learn](https://docs.microsoft.com/ru-ru/learn/modules/intro-to-python/). Попробуйте интерактивную платформу Microsoft Learn и получите навыки выполнения этого модуля, охватывающего основы написания базового кода Python, объявления переменных и работы с входными и выходными данными консоли. Интерактивная среда песочницы предоставляет эту отличную возможность начать пользователям, у которых еще не настроена среда разработки Python.
* [Python на Pluralsight: 8 курсов, 29 часов](https://app.pluralsight.com/paths/skills/python). Схема обучения Python на Pluralsight предлагает онлайн-курсы, охватывающие различные темы, связанные с Python, включая средство для измерения навыков и поиска пробелов в знаниях.
* [Учебники на сайте LearnPython.org](https://www.learnpython.org/). Приступите к изучению Python без необходимости дополнительной установки или настройки с помощью этих бесплатных интерактивных учебников по Python от DataCamp.
* [Учебники на сайте Python.org](https://docs.python.org/3/tutorial/index.html). Предоставляют читателям неформальное описание основных понятий и функций языка и системы Python.
* [Изучение Python на сайте Lynda.com](https://www.lynda.com/Python-tutorials/Learning-Python/661773-2.html). Основные сведения о Python.

**Работа с Python в VS Code**

* [Редактирование Python в VS Code](https://code.visualstudio.com/docs/python/editing). Узнайте больше о том, как воспользоваться преимуществами автозаполнения VS Code и поддержкой IntelliSense для Python, включая их настройку или отключение.
* [Анализ кода Python](https://code.visualstudio.com/docs/python/linting). Анализ кода — это процесс запуска программы, которая будет анализировать код на наличие возможных ошибок. Узнайте о различных формах поддержки анализа кода VS Code для Python и о том, как выполнить его настройку.
* [Отладка Python](https://code.visualstudio.com/docs/python/debugging). Отладка — это процесс обнаружения и удаления ошибок из компьютерной программы. В статье по этой ссылке описывается инициализация и настройка отладки для Python с помощью VS Code, установка и проверка точек останова, присоединение локального скрипта, выполнение отладки для различных типов приложений или на удаленном компьютере, а также некоторые основные способы устранения неполадок.
* [Модульное тестирование Python](https://code.visualstudio.com/docs/python/unit-testing). В статье по этой ссылке содержатся некоторые основные сведения о модульном тестировании, включении платформы тестирования, создании и выполнении тестов, отладке тестов и параметрах конфигурации теста, а также приведено пошаговое руководство с примером.