

Инструкция

# Как настроить окружение в PyCharm на компьютере с Windows

## Что такое окружение

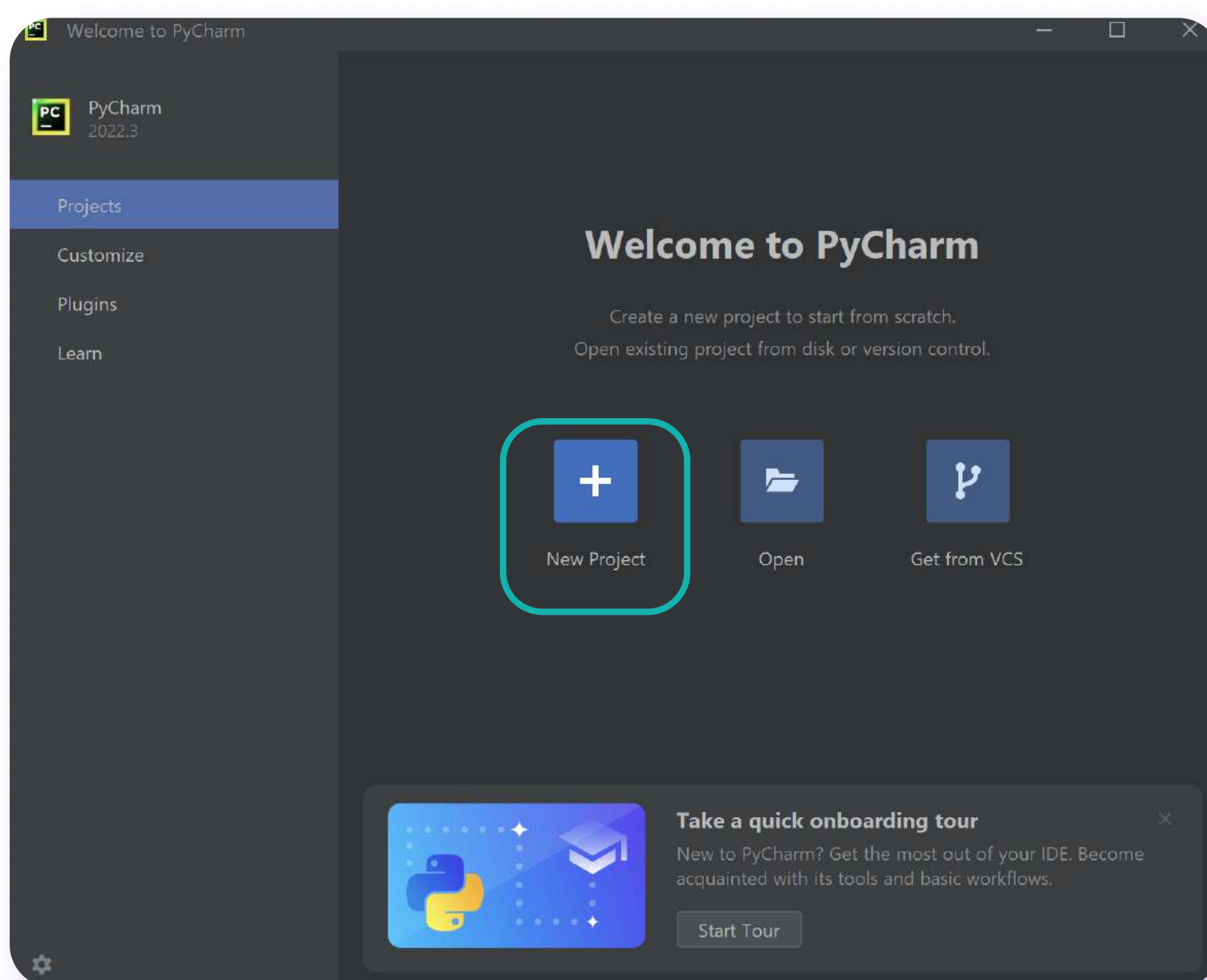
Окружение — это все внешние модули (библиотеки, пакеты), которые доступны для работы с кодом.

Скачайте модули и включите их в окружение, чтобы ссылаться на функции других разработчиков.

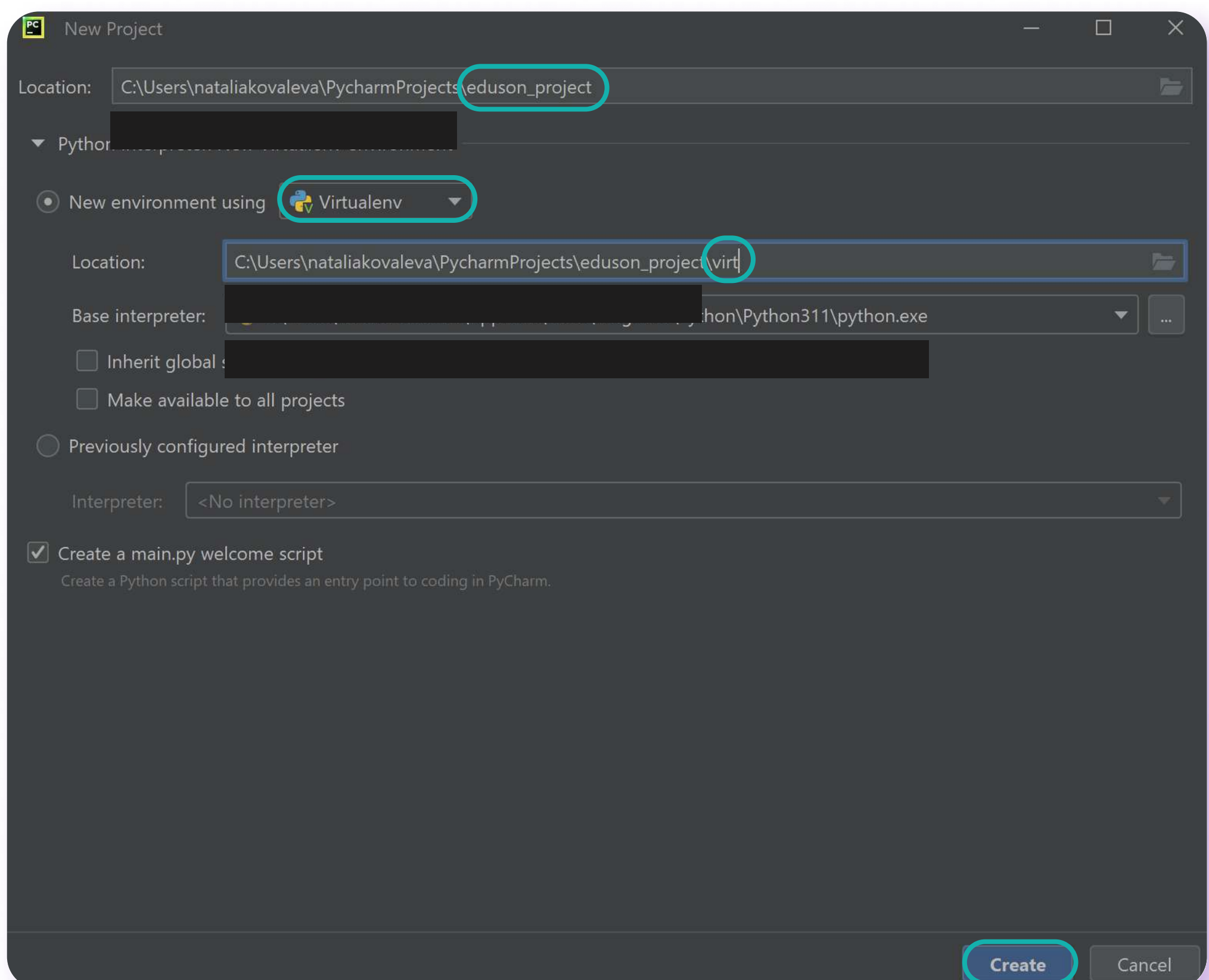
Виртуальное окружение хранит внешние модули для проекта, но исключает доступ к ним из других проектов. Это помогает избегать проблем с совместимостью версий.

## Как настроить виртуальное окружение для нового проекта

1. Откройте PyCharm и нажмите на New Project, чтобы создать новый проект.



2. Прodelайте следующие шаги на странице New Project, чтобы создать окружение:
- в поле Location выберите, куда должен сохраниться проект, и введите название (например, eduson\_project);
  - выберите, какой инструмент для виртуального окружения вы хотите использовать (например, Virtualenv);
  - замените название окружения в поле Location под выбранным инструментом (например, virt);
  - нажмите Create внизу страницы, чтобы создать проект.



После того, как вы проделаете эти действия, программа откроет новое рабочее пространство.

## Зачем использовать сторонние пакеты, библиотеки и фреймворки при работе с Python

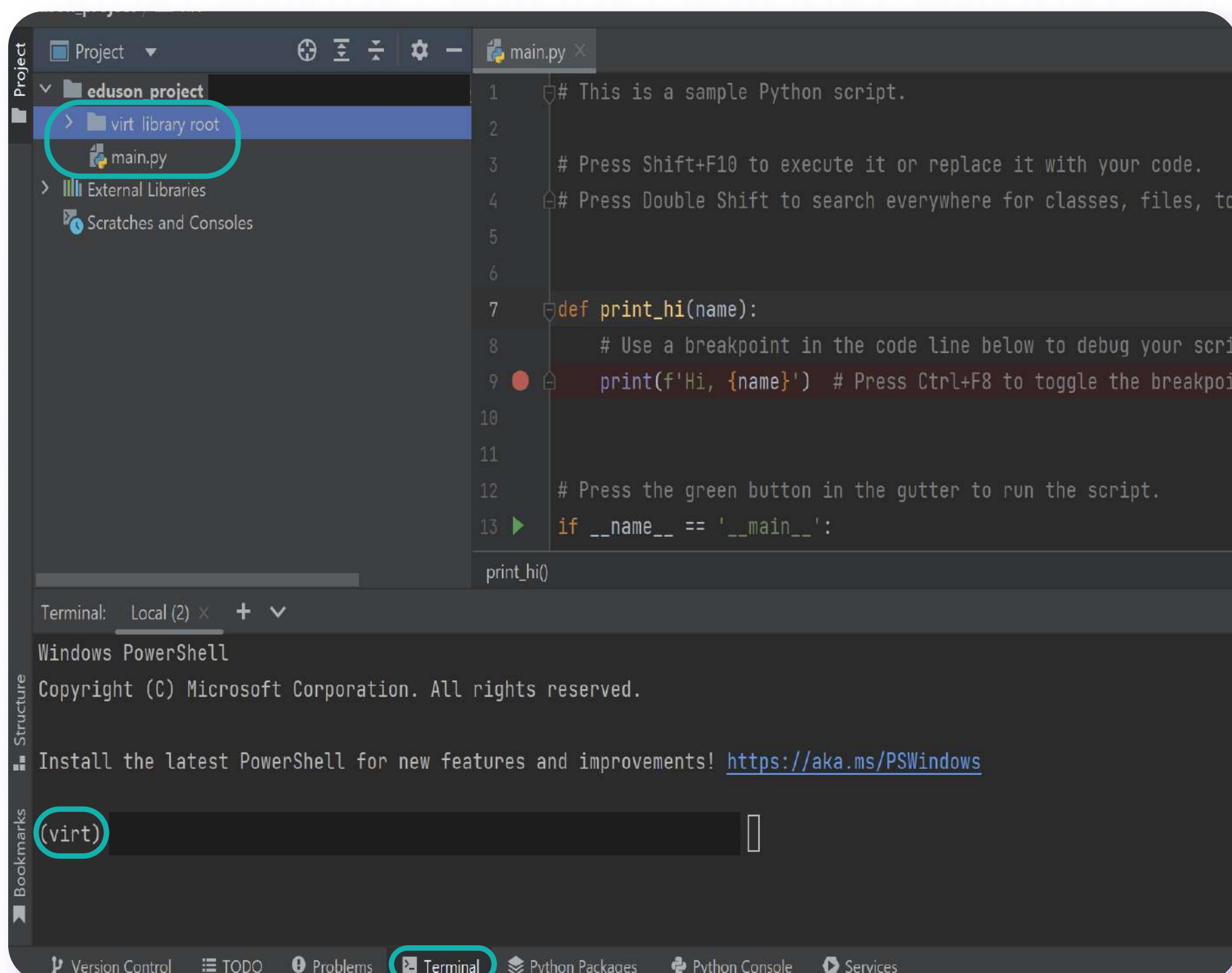
Сторонние пакеты, библиотеки и фреймворки позволяют ускорить разработку. Они хранятся в центральном репозитории Python — PyPI.

Библиотеки можно установить с помощью инструмента PIP. PIP устанавливается по умолчанию вместе с Python версии 3.4 и выше.

Скачайте библиотеки и фреймворки через Terminal внутри виртуального окружения в PyCharm, чтобы они не влияли на другие проекты.

## Как скачать модули внутри виртуального пространства в PyCharm

1. Нажмите на опцию Terminal на нижней панели PyCharm, чтобы открыть встроенный в IDE терминал. Название рабочего виртуального окружения отобразится в круглых скобках в строке для ввода команд.





2. Чтобы проверить номер версии Python, введите команду в терминале:

```
python --version
```

Номер версии появится на следующей строке. Если версия Python ниже 3.4, обновите ее, чтобы перейти к следующему шагу.

```
(virt) python --version  
Python 3.11.1
```

3. Чтобы установить библиотеку, введите следующую команду. Вместо квадратных скобок укажите название внешней библиотеки, которую хотите скачать:

```
pip install [название пакета]
```

На экране отобразится прогресс установки. Когда она завершится, на экране появится сообщение:

```
Successfully installed [пакет-номер версии]
```


Когда вы устанавливаете некоторые модули, программа может автоматически скачать дополнительные необходимые пакеты. Их названия и версии также отобразятся после успешной установки.

```
(virt) pip install django  
Collecting django  
  Downloading Django-4.1.4-py3-none-any.whl (8.1 MB)  
    |████████████████████████████████████████| 8.1 MB 2.2 MB/s  
Collecting tzdata  
  |████████████████████████████████████████| 42 kB ...  
Collecting asgiref<4,>=3.5.2  
  Downloading asgiref-3.6.0-py3-none-any.whl (23 kB)  
Installing collected packages: tzdata, sqlparse, asgiref, django  
Successfully installed asgiref-3.6.0 django-4.1.4 sqlparse-0.4.3 tzdata-2022.7
```

4. Чтобы проверить, какие библиотеки и пакеты уже установлены в окружении, введите команду:

```
pip list
```

Установленные пакеты отобразятся в списке.



```
(virt) pip list
Package Version
-----
asgiref 3.6.0
Django 4.1.4
pip 21.3.1
setuptools 60.2.0
sqlparse 0.4.3
tzdata 2022.7
wheel 0.37.1
```

Настройте виртуальное окружение для каждого проекта и скачайте нужные модули внутри него, чтобы избежать проблем с совместимостью в будущем.