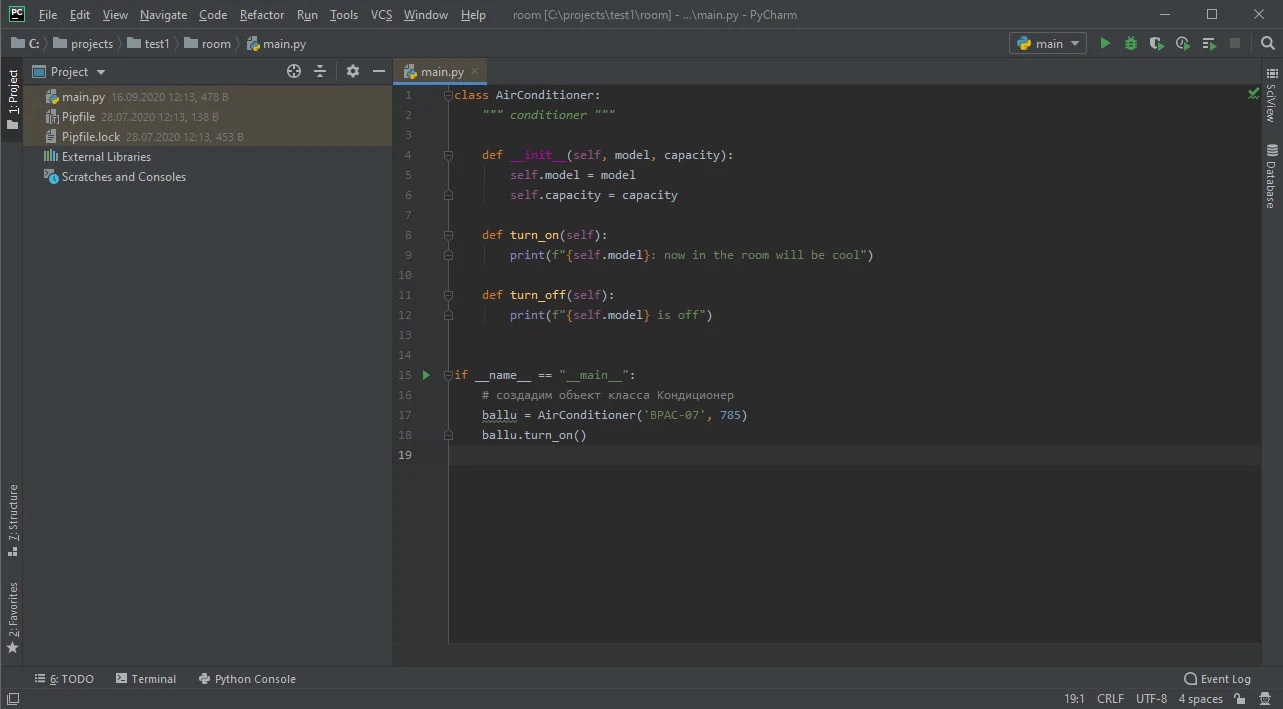
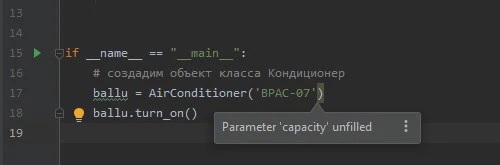
Установка и настройка среды Pycharm

PyCharm кроссплатформенна и встреча с Windows, Linux и MacOS. Она поддерживает, как вторую, так и третью версию Питона, кроме всего прочего, имеет красивый и функциональный интерфейс.



Инструменты и функции, предоставляющие интегрированную средой, позволяют Python-программистам быстро и эффективно писать код, синхронизироваться с системой контроля версий, использовать фреймворки и плагины, позволяя при этом настроить интерфейс так, как удобно самому пользователю и даже предоставить возможность дополнительного расширения IDE.

В PyCharm, конечно же, есть встроенный анализатор кода, который действительно помогает при его написании. Благодаря анализатору можно весьма комфортно перемещаться по проекту, использовать быстрый поиск, а также исправлять ошибки, которые PyCharm любезно подсвечивает и подробно описывает.



Тут реализованы системы автодополнения и контроля качества кода в со[ответ](https://sdo.eduprof.ru/mod/url/view.php?id=32771" \o "Ответ)ствии со стандартом PEP8. Всё для того, чтобы сделать текст ваших программ красивее, чище и структурированнее.

В интегрированной среде от JetBrains поддерживаются современные Python-фреймворки для веб-разработки, в ней можно работать с Jupyter-notebook, подключать Anaconda, а также прочие библиотеки для научных вычислений и Data Science.

Процесс установки вкратце описан на [официальном сайте](https://www.jetbrains.com/ru-ru/pycharm/) для каждой из поддерживаемых ОС:

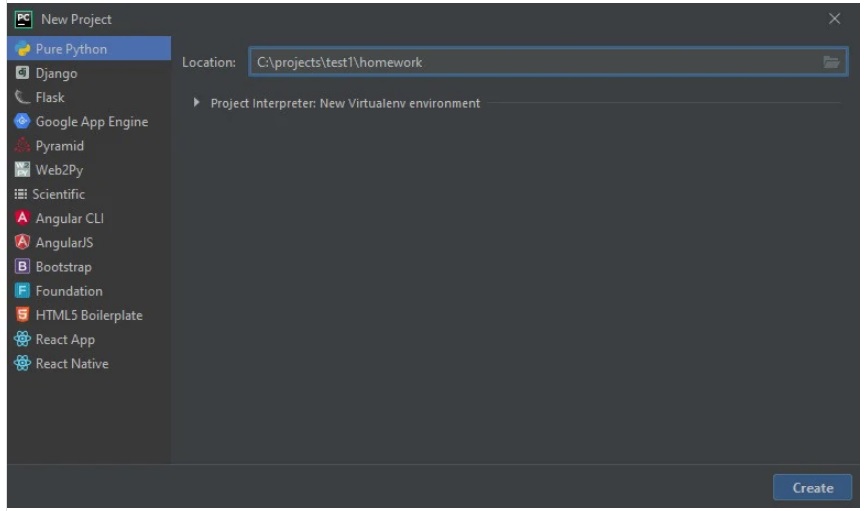
На Windows (Ниже представлено видео установки)

1. Запускаете файл  pycharm.exe.
2. Следуете инструкциям инсталл-мастера.
3. Выбираете нужные параметры установки.
4. Готово.

**Подготовка к работе и настройка**

Перед тем, как приступить к разработке в PyCharm, вам нужно создать проект.

Чтобы начать новый проект, нажмите Новый проект и дайте имя. Теперь можно приступить к настройке.



Настройки интерфейса

Зато вы можете изменить сами шрифты и их размеры, пройдя по пути:

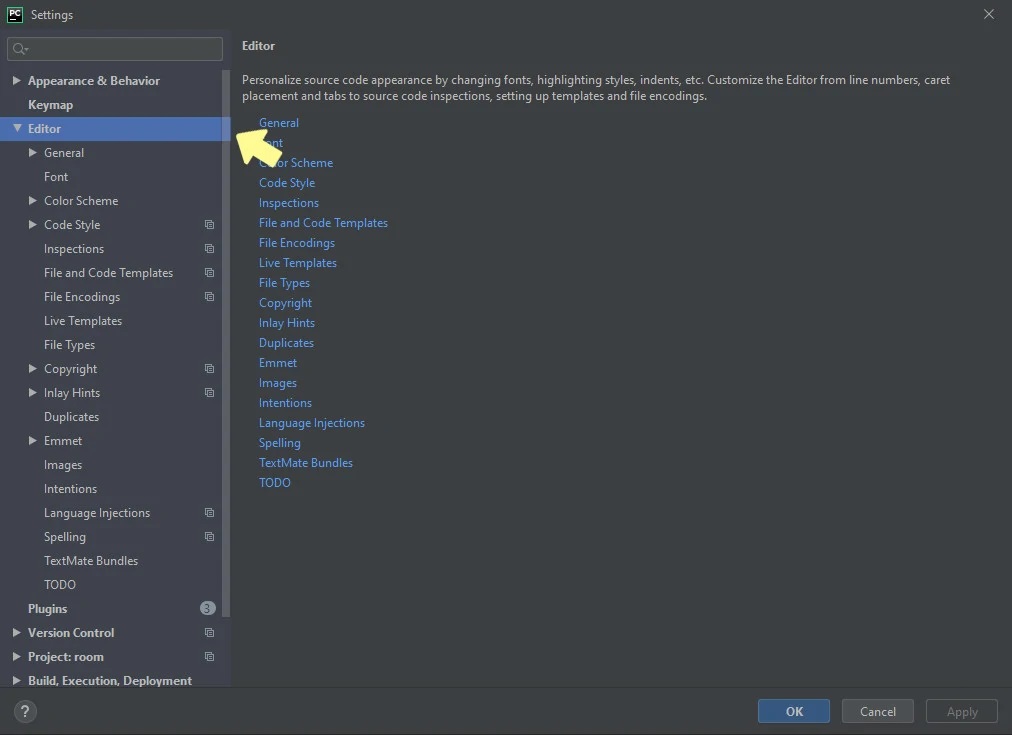
File -> Settings -> editor -> font

Поменять цветовую схему UI:

File -> Settings -> editor -> color scheme

И сконфигурировать различные другие настройки, вроде общих настроек, конфигураций, настроек каждого доступного языка и так далее. Для всего этого есть редактор:

file -> settings -> editor

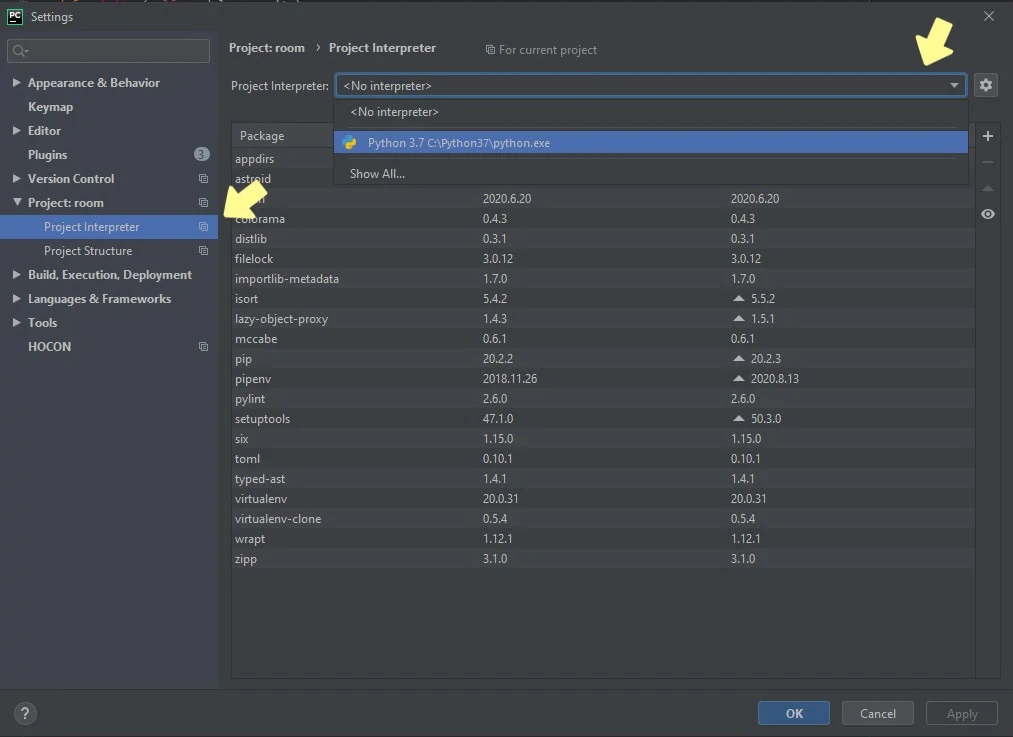


**Настройка интерпретатора**

Скачать интерпретатор вы можете с [официального сайта Python](https://www.python.org/downloads/) .

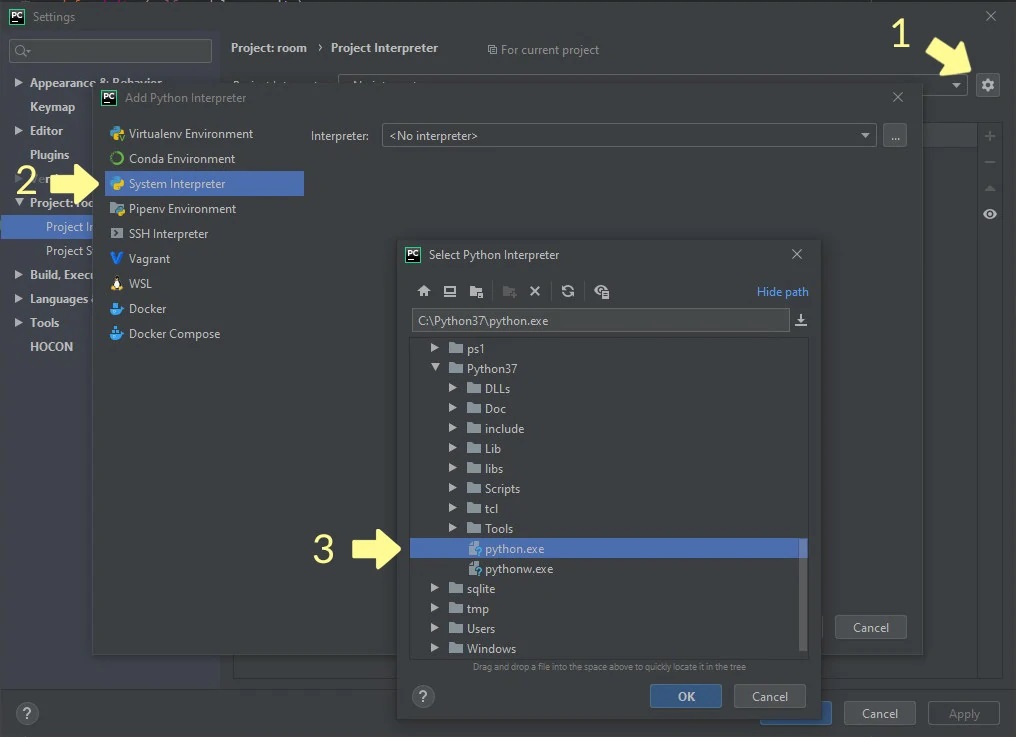
Для использования установленного интерпретатора, выберите его в настройках:

File -> Settings -> Project:<name> -> Project Interpreter



Если PyСharm не видит интерпретатор, попробуйте добавить его вручную. Для этого:

1. Нажмите на шестеренку в верхнем правом углу, выберите " Добавить ... ".
2. Далее выберите " Системный интерпретатор ";
3. Нажмите на 3 точки " ... " справа от поля в выборе интерпретатора;
4. Укажите до интерпретатора.



**Настройка виртуального окружения**

Если вам приходится работать с несколькими проектами, то определенно не помешает настройка и подключение виртуального окружения.

Venv – это директория, содержащая ссылку на интерпретатор и набор установленных библиотек. Виртуальное окружение “изолирует” ваши проекты, и помогает, в частности, не запутаться в разных версиях библиотек, адаптированных для каждого из них по отдельности.

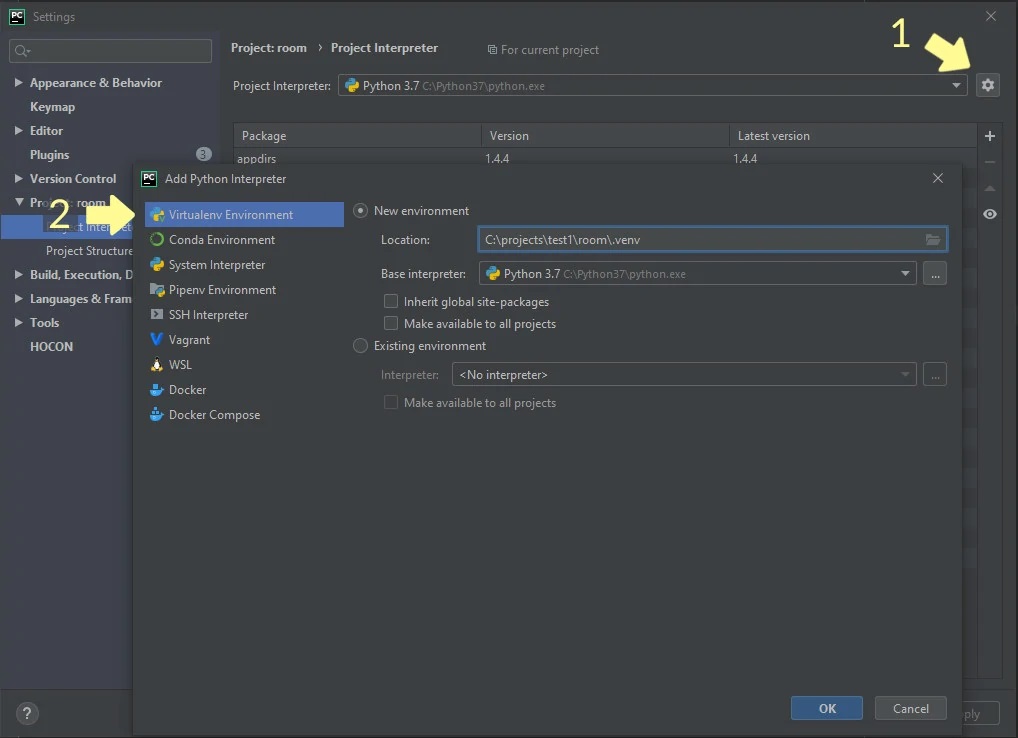
Подробнее про виртуальное окружение можно прочитать на странице ["Виртуальное окружение Python(venv)"](https://sdo.eduprof.ru/mod/page/$@PAGEVIEWBYID*20113@$)

Если вы настроили виртуальное окружение заранее, PyCharm покажет уведомление, в котором предложит использовать его в проекте. Если уведомления нет, вы всегда можете настроить и изменить его вручную в настройках:

File -> Settings -> Project:<name> -> Project Interpreter

Также в этом меню можно создать виртуальные окружения с нуля. Для этого:

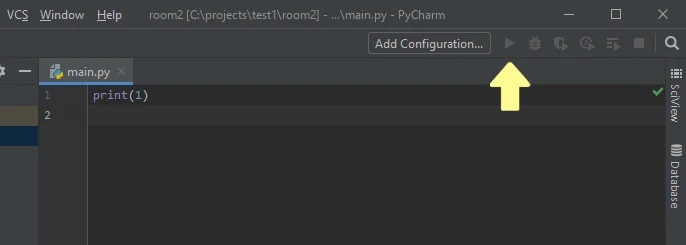
* Нажмите на шестеренку в верхнем правом углу, выберите "Add..".
* Выберите "Virual Enviroment" и задайте параметры.



Запуск скриптов в консоли

Чтобы запустить код в PyCharm, нужно выбрать интерпретатор в настройках .

Также, перед запуском своего кода в PyCharm необходимо добавить конфигурацию (чтобы IDE понимал, какой файл ему запускать и по каким правилам). Изначально, если проект новый, конфигурации нет, и кнопка запуска не активна.

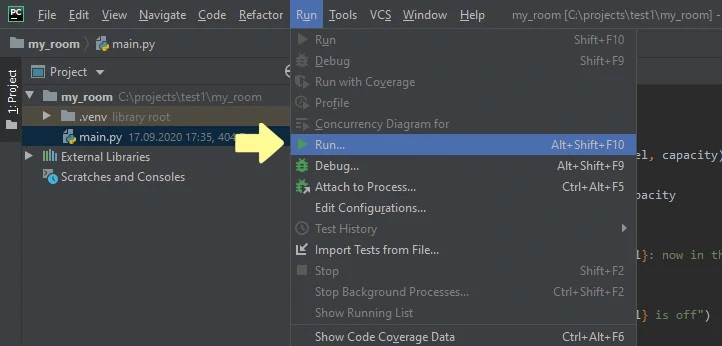


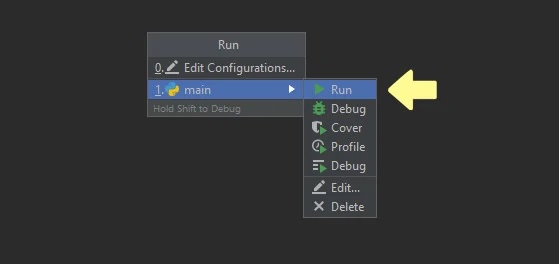
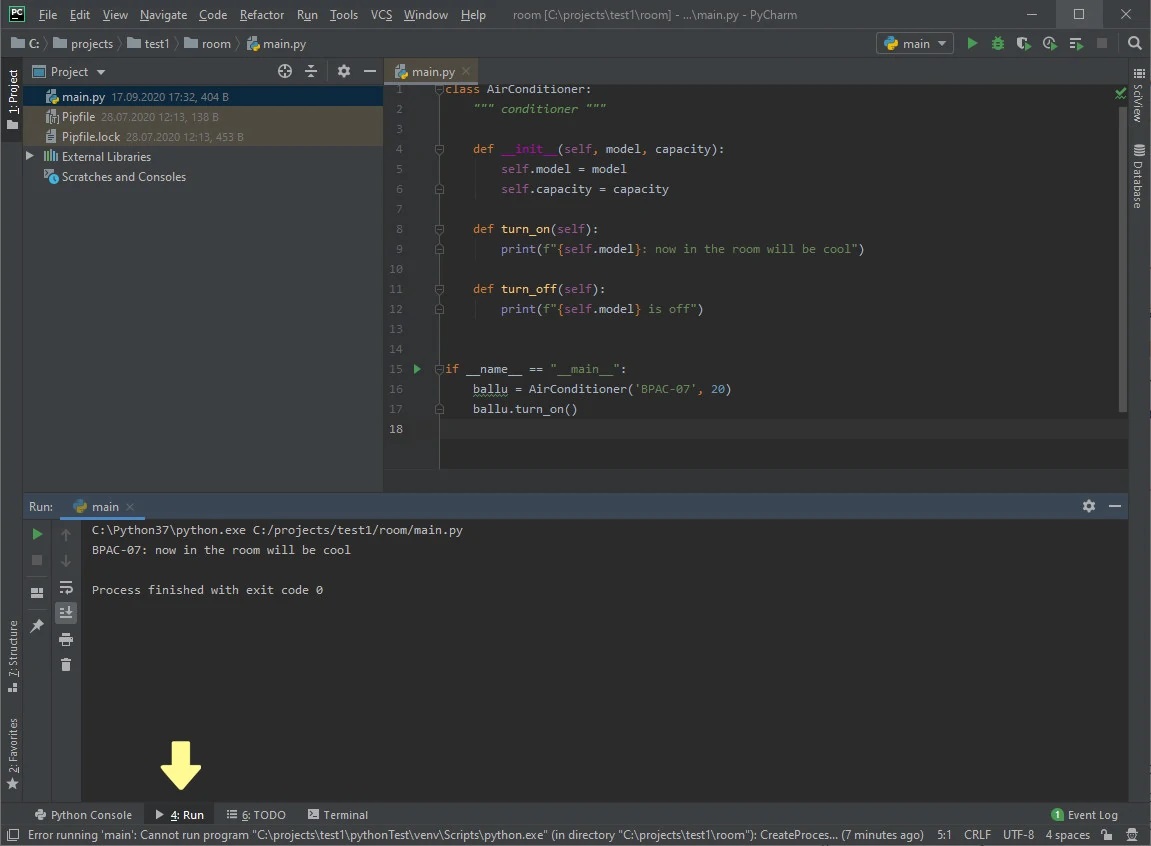
Чтобы запустить код в новом проекте, зайдите в

Run -> Run

или выполните комбинацию:

Alt + Shift + F10

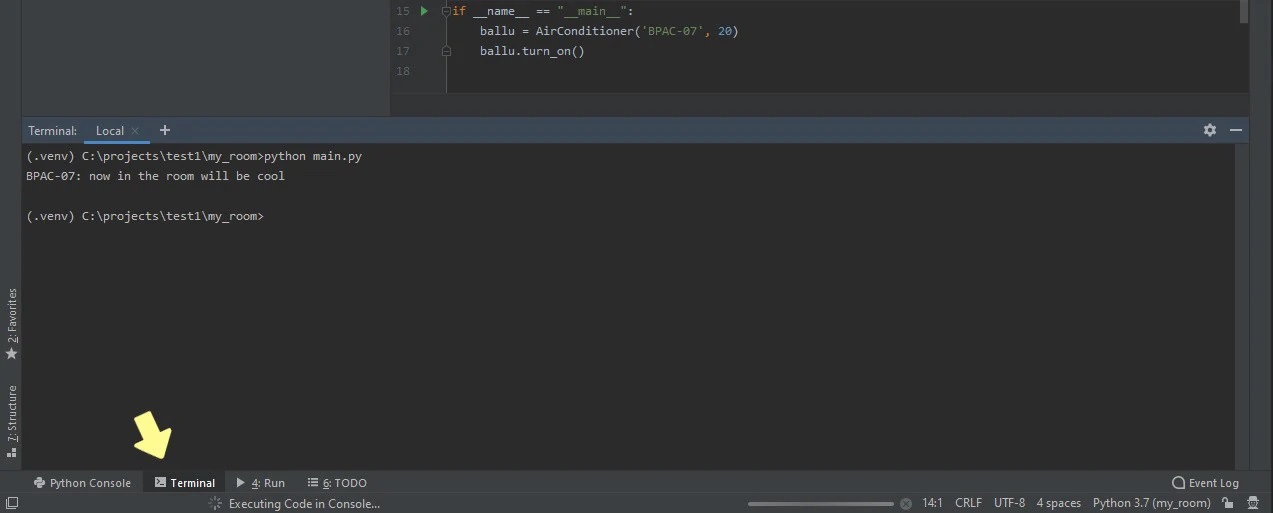


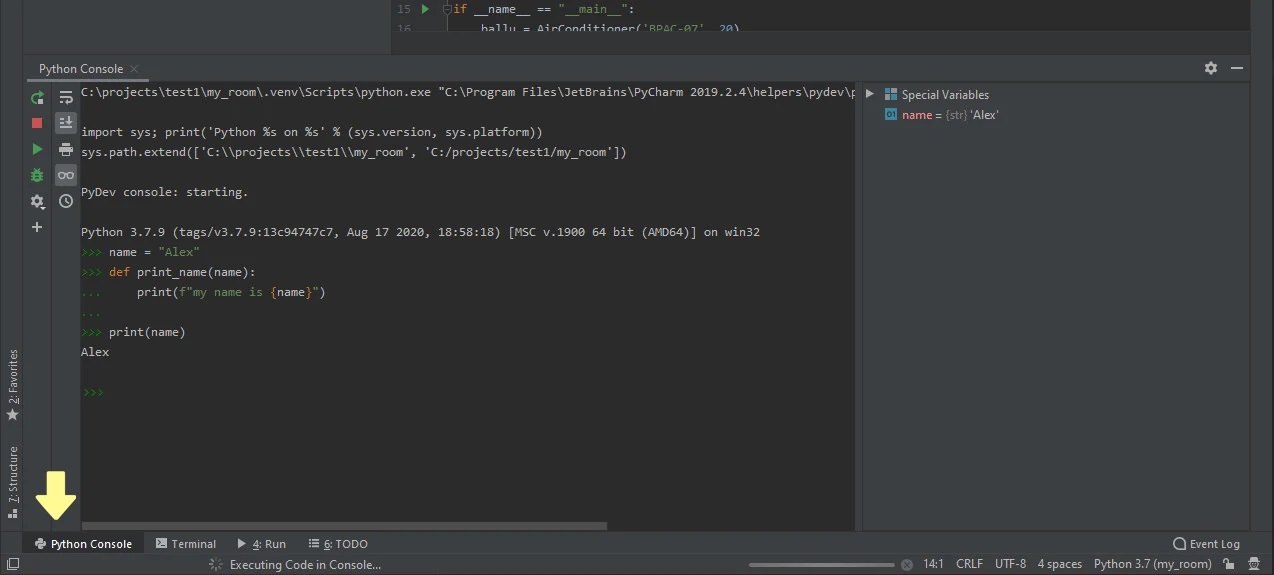
Откроется диалоговое окно в выбором файла, который необходимо запустить.  
  
  
После запуска, откроется терминал, в котором вы увидите результат выполнения своего   
скрипта и сообщение о том, что процесс завершился с кодом 0 (0 означает успешное завершение).  


Далее, когда конфигурация создана и сохранена, запускать свой код на выполнения можно комбинацией:  
Shift + F10  
☝️ Если скрипт не запускается, возможно вы не сохранили конфигурацию или PyCharm не видит интерпретатор.  
Настроить или создать конфигурацию можно в меню "Edit Configurations..", которое находится сверху справа,  
около кнопки "Run".

Есть еще один способ запуска кода:

Внизу окна программы кликните на "Terminal".  
Напишите в терминале python <name>.py или python3 <name>.py.

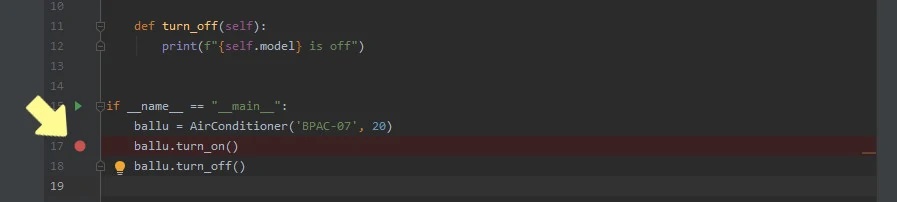
  
В целях обучение, вам может понадобиться быстро писать и проверять Python-инструкции. Для этого в PyCharm-е реализован интерактивный режим (по аналогии с IDLE). Чтобы его вызвать, достаточно нажать на "Python Conole" внизу слева.



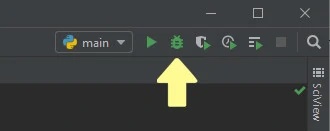
Отладчик (debugger) в Pycharm

Отладка – чрезвычайно полезный инструмент. С помощью него можно пошагово исполнять программный код. Интерпретатор переходит в особый режим, фиксируя на каждом этапе выполнения текущее состояние программы. Т.е. для каждой строки кода вам будет представлен мини-отчёт, в котором можно посмотреть актуальные значения всех переменных и проследить таким образом весь процесс их изменения.

Отладчик сокращает время поиска ошибок на порядки.



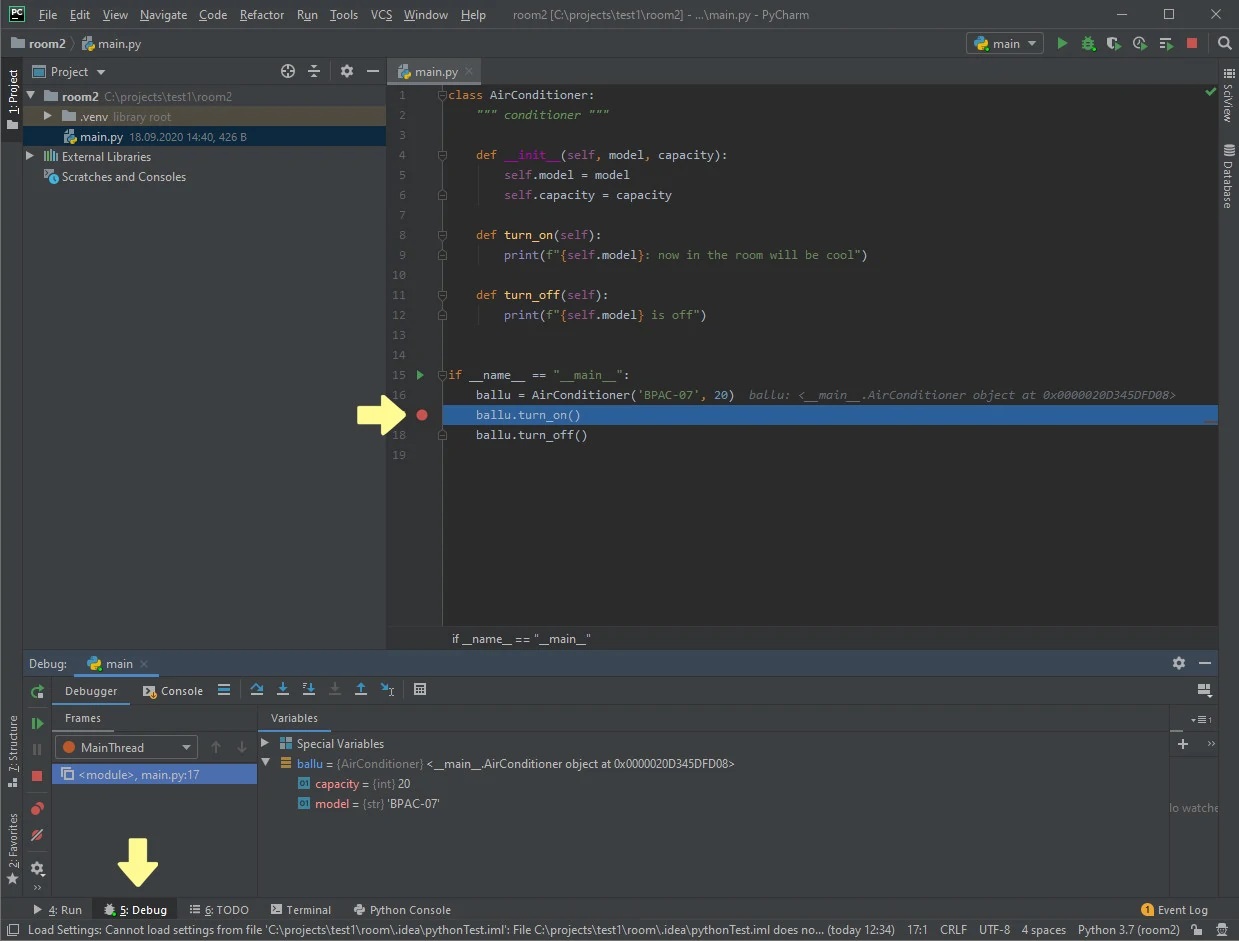
Для старта отладки, нужно поставить так называемую точку останова. Это красный кружок напротив строки кода.



Или выполните комбинацию:

Shift + F9

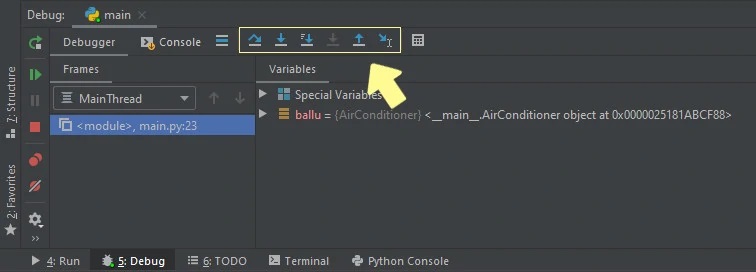
Отладка началась.



Теперь, нажимая F8, можно последовательно шагать по строчкам кода и смотреть текущее состояние всей программы.

"Шагать" можно и через F7, но в таком случае дебаггер будет совершать "Шаг с заходом". Т.е. при встрече вызова функции он будет заходить в её описание и последовательно проходить по инструкциям.

Все варианты навигации в режиме отладки можно посмотреть в окне "Debugger":



Для каждой кнопки существует свой hotkey – чтобы увидеть его, достаточно навести курсор на нужную кнопку.

Что делать, если PyCharm не видит модули Python?

Если у вас возникла подобная проблема, то в первую очередь стоит проверить настройки интерпретатора.

Вариант №1 – сброс настроек и выборе новой версии интерпретатора. Настройки находятся тут:

Settings -> Project:<project\_name> -> Project Interpreter

Вариант №2 – По умолчанию PyCharm ищет модули начиная от корня проекта. Подскажите PyCharm, где находится директория с вашими модулями. Для этого в проводнике слева щелкните по директории (правой кнопкой мыши), и в контекстном меню выберите Mark Directory As -> Sources Root.

Последнее изменение: Суббота, 13 Февраль 2021, 16:23