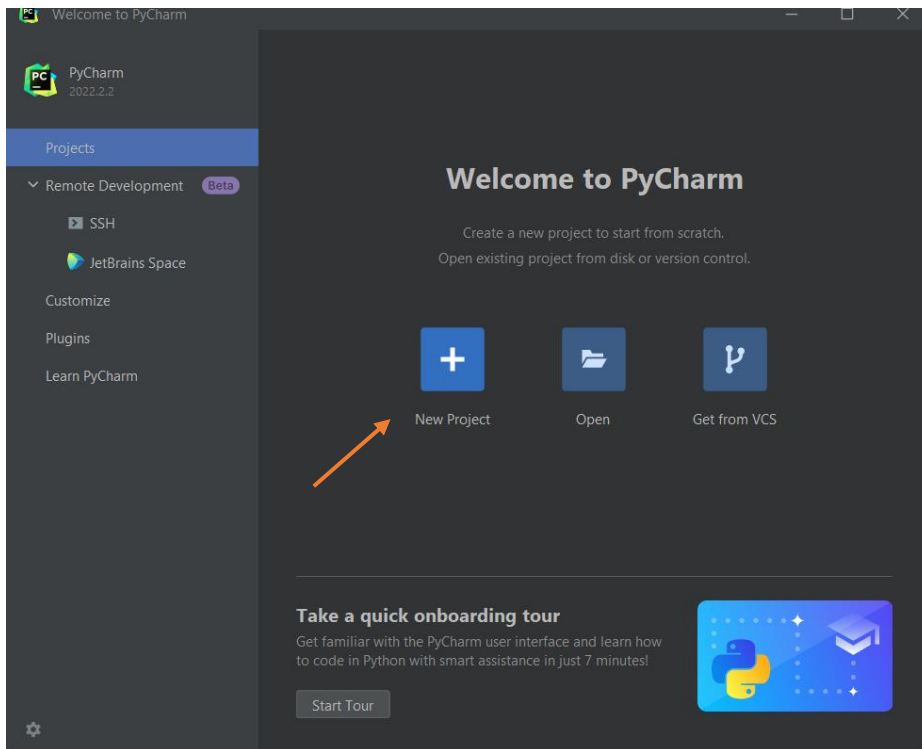
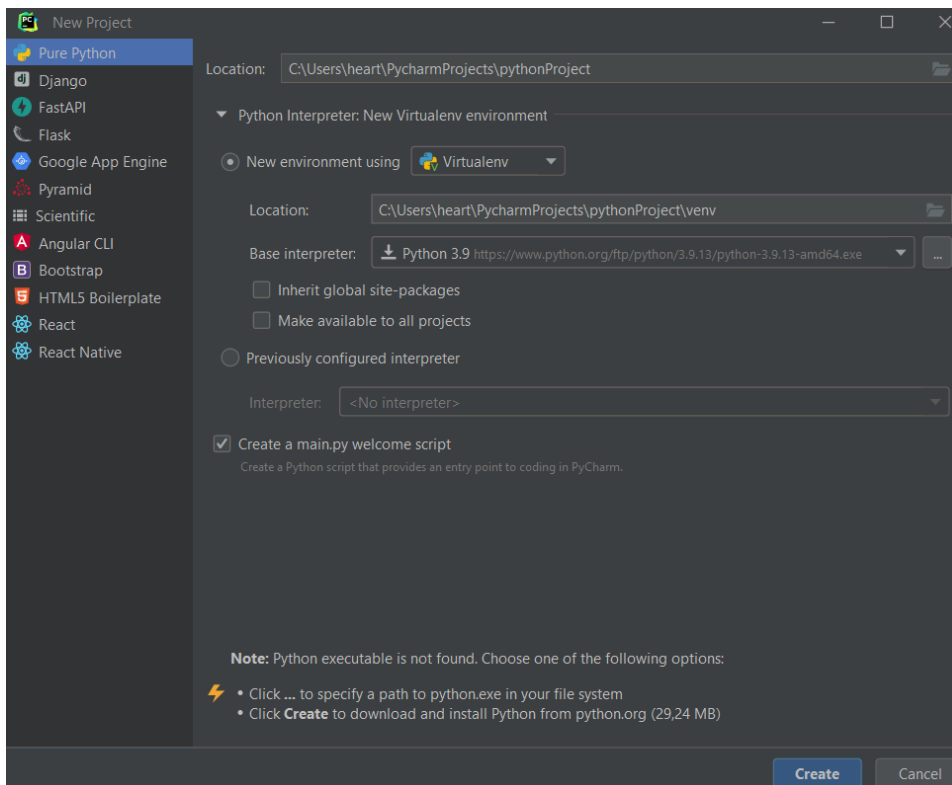


Создание проекта

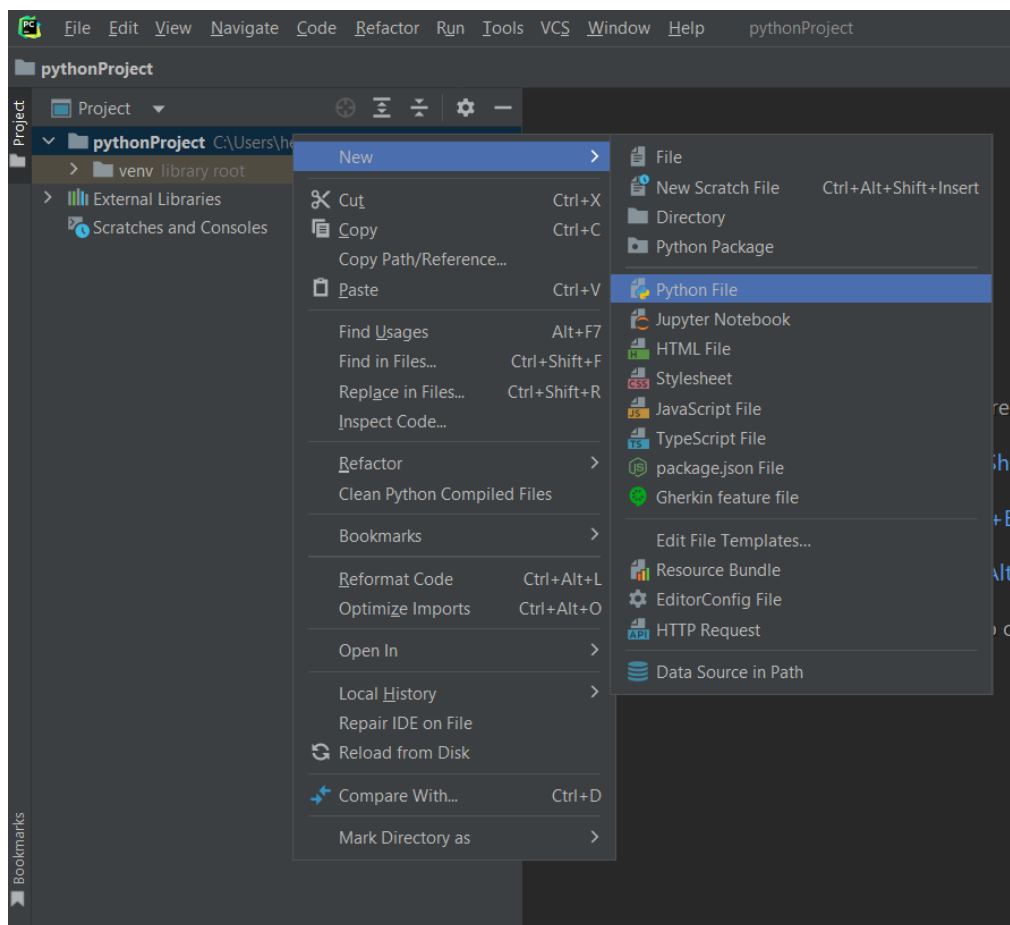
1. Нажать на кнопку новый проект



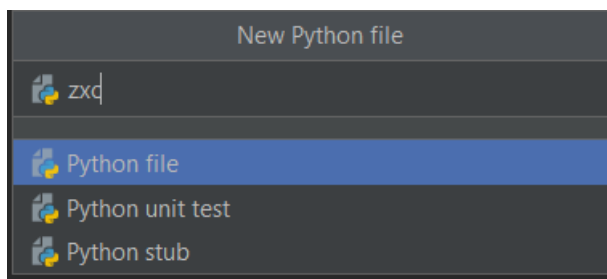
2. Выбираем папку для нашего проекта и тип виртуальной среды



3. Создаём “Python file” в корневой папке

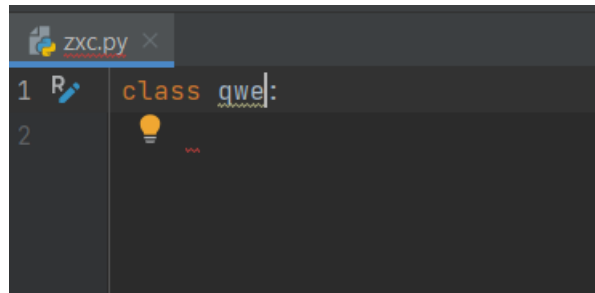


4. Даём имя файлу

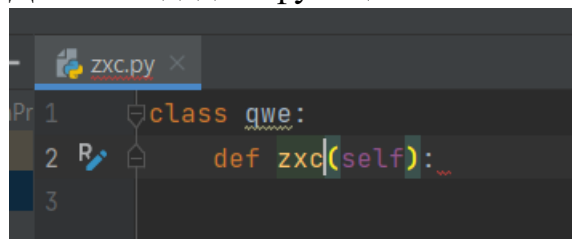


Кодирование

1. Начнём с объявления класса (class) и введём его имя



2. Далее создадим функцию def



Если при редактировании кода вы заметили какие-либо предупреждения о проверке, щелкните символ лампочки, чтобы просмотреть список возможных исправлений и рекомендуемых действий

Также PyCharm анализирует код в реальном времени результаты сразу отображаются на индикаторе проверки в правом верхнем углу редактора. Этот индикатор проверки работает как светофор: когда он горит зеленым, все в порядке, и вы можете продолжать свой код; желтый индикатор означает некоторые незначительные проблемы, которые, однако, не повлияют на компиляцию; но когда горит красный, это означает, что у вас есть серьезные ошибки. Щелкните его, чтобы просмотреть подробности в окне инструмента "Проблемы".

Форматирование кода

Переформатировать фрагмент кода

1. В редакторе выберите фрагмент кода, который вы хотите переформатировать.

Перед переформатированием вы можете ознакомиться с настройками стиля кода, которые применяются к выбранному коду: нажмите **Alt+Enter** и щелкните Настроить параметры стиля кода.

2. В главном меню выберите Код / Переформатировать код или нажмите **Ctrl+Alt+L**.

Переформатировать файл

1. Либо откройте файл в редакторе и нажмите **Ctrl+Alt+Shift+L** или в окне инструментов проекта щелкните файл правой кнопкой мыши и выберите Переформатировать код.
2. В открывшемся диалоговом окне переформатирования файла, если вам нужно, выберите следующие параметры переформатирования:
3. Нажмите Выполнить.

Если вы хотите увидеть точные изменения, внесенные в ваш код во время переформатирования, используйте функцию локальной истории.

Переформатировать отступы строк

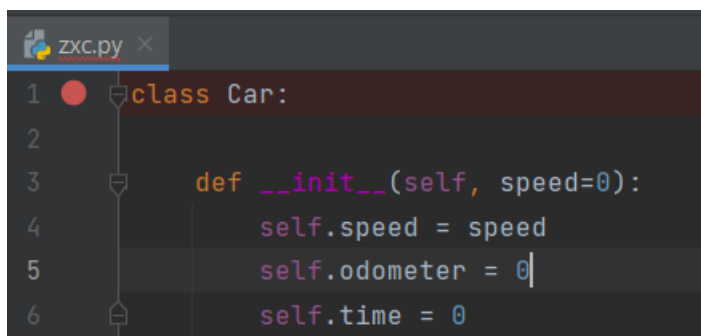
1. Находясь в редакторе, выберите необходимый фрагмент кода и нажмите **Ctrl+Alt+I**.
2. Если вам нужно настроить параметры отступов, в диалоговом окне Настройки / Настройки (**Ctrl+Alt+S**) перейдите в Редактор | Стилль кода.
3. На странице соответствующего языка на вкладке Вкладки и отступы укажите соответствующие параметры отступов и нажмите кнопку ОК.

Автоматическое переформатирование кода при сохранении

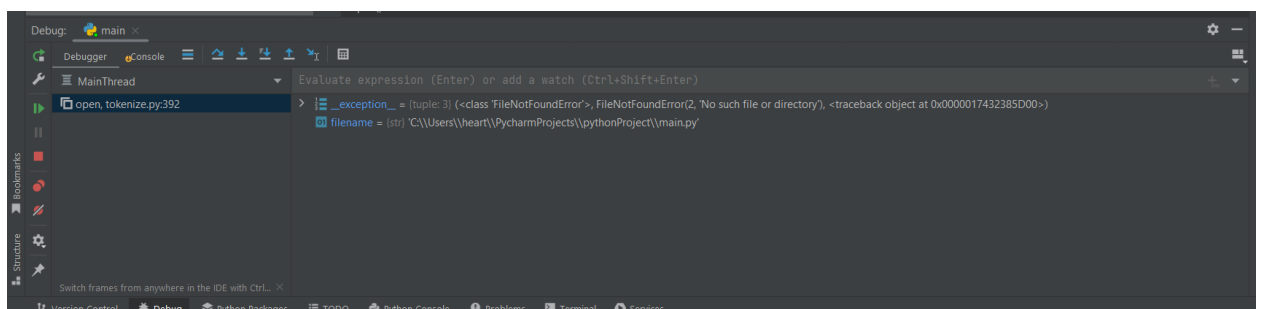
1. Нажмите Ctrl+Alt+S, чтобы открыть настройки IDE, и выберите Инструменты | Действия при сохранении.
2. Включите параметр Переформатировать код.
3. Кроме того, вы можете настроить способ, которым IDE будет переформатировать ваш код

Отладка и запуск

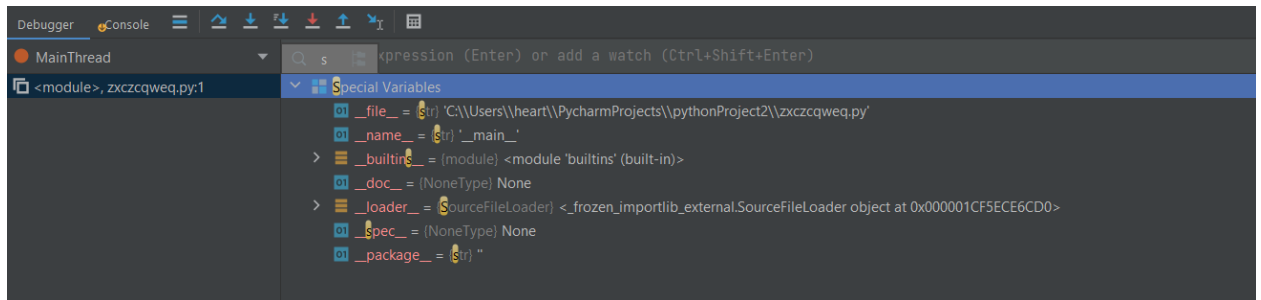
Чтобы подготовить PyCharm к отладке, необходимо указать так называемую точку останова. Это та строка, после которой запустится отладчик и выполнение кода станет пошаговым. Поскольку программа у нас небольшая, то мы скажем отладчику начать с самого начала. Чтобы поставить точку остановки, нужно щелкнуть мышью слева от желаемой строки и справа от ее номера. На этом месте появится красный кружок, говорящий о том, что в данном месте произойдет пауза. Чтобы убрать точку остановки, нужно нажать на красный кружок еще раз.



Чтобы просто запустить программу после первого запуска, достаточно нажать Shift + F10. А чтобы запустить программу в режиме отладки, нужно нажать Shift + F9. Причем все изменения в коде сохраняются автоматически, в отличие от IDLE. Запускаем отладку для нашей программы и смотрим в нижнюю часть экрана.



Мы теперь видим две вкладки: Отладчик и Консоль. В консоли мы видим все, что программа вывела на данный момент. Поскольку мы остановили выполнение в самом начале, то в консоли пока будет пусто. Для нас интерес представляет именно окно отладчика, поскольку здесь мы можем отслеживать изменение всех переменных. Как видите, у нас уже есть список Special Variables (Специальные переменные), в котором вы можете посмотреть значения таких служебных переменных, как `__name__`



Чтобы начать отслеживание новой переменной, нажмем знак с плюсом и введем туда имя нашей переменной `i`. Сначала мы увидим надпись `name 'i' is not defined`, а это означает, что наша переменная пока еще даже не создана. Давайте тогда скажем отладчику обработать следующую строку кода. Тут важно заметить, что существует 2 основных способа пошагового выполнения: шаг с заходом и шаг с обходом.

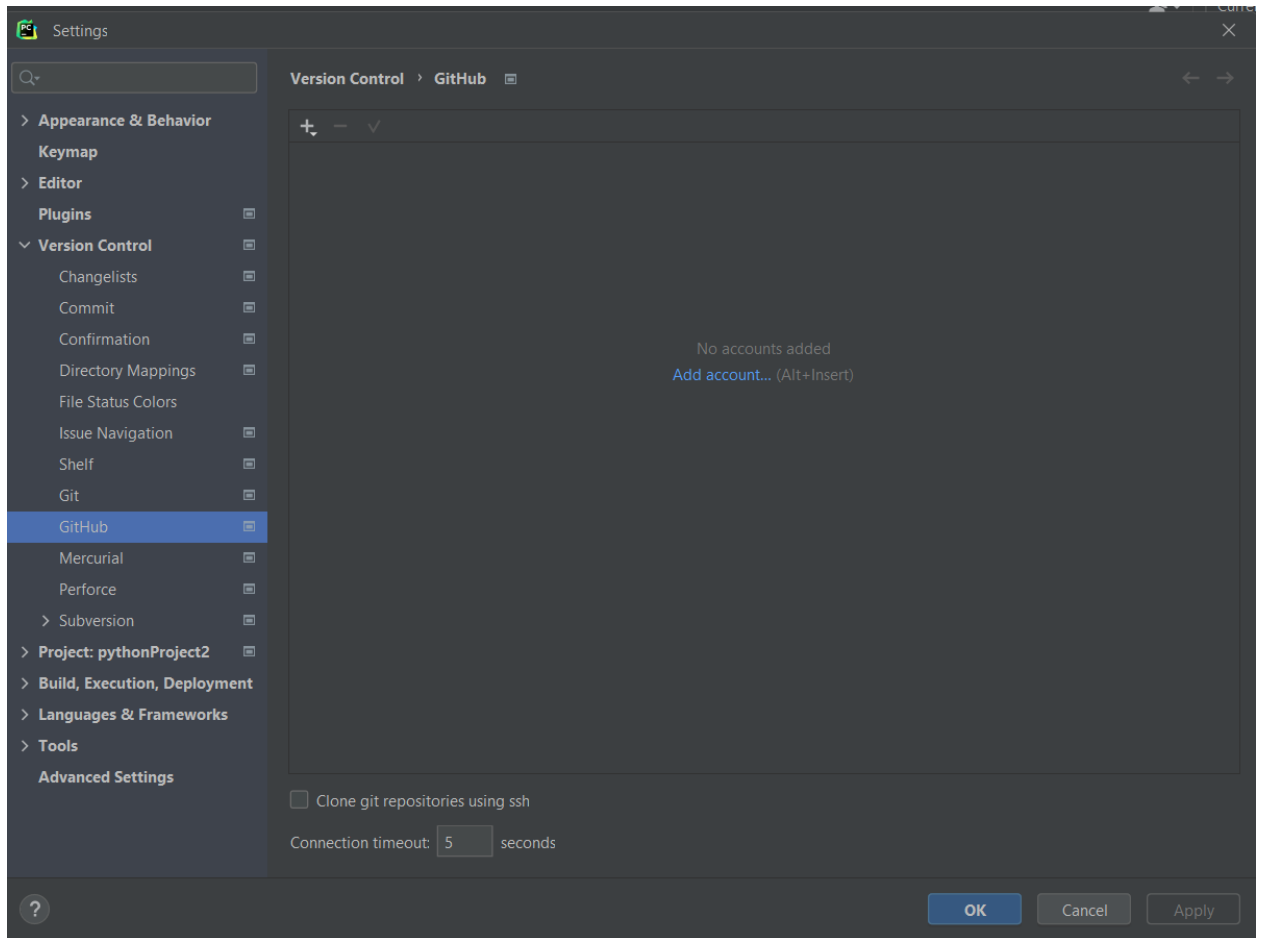
1. При шаге с заходом отладчик будет «заходить» внутрь всех функций и других конструкций, если их встретит. То есть, если отладчик дойдет до вызова некоторой функции `func()`, то он перейдет в ту часть кода, где описывается ее содержимое. Чтобы совершить один шаг с заходом, нажмите F7.
2. При шаге с обходом отладчик не будет «заходить» внутрь всех функций и других конструкций, а просто выполнит весь код функции без остановки и продолжит выполнение дальше. Чтобы совершить один шаг с обходом, нажмите F8.

В нашем примере без разницы что использовать, поскольку никаких функций у нас нет. Поэтому давайте, например, будем использовать шаг с обходом. При пошаговой обработке каждой строки мы 9 раз пройдем внутри цикла увидим, как 9 раз меняется переменная `i`.

Отладку можно как остановить вручную, так и дожидаться, пока отладчик обработает все строки в программе, что почти не встречается на практике. Также можно создавать несколько точек остановки.

Публикация в репозитории

1. Добавляем аккаунт в настройках и публикуем



2. Либо так

