

Настройка рабочего окружения | Я.Шпора

Терминал

На Linux и macOS используйте встроенные приложения Bash и Zsh соответственно.

Для Windows установите приложение Git Bash и работайте с ним.

Основные команды для работы в терминале

Узнать текущую директорию: **pwd**

```
$ pwd
/c/Users/user # Вывод для Windows.
/home/user    # Вывод для Linux.
/Users/user   # Вывод для macOS.
```

Узнать содержимое директории: **ls**

```
$ ls
```

Навигация по папкам: **cd**

```
cd имя_директории
```

В качестве директорий можно указывать специальные сокращения:

- **~** — домашняя директория;
- **..** — директория на уровень выше.

```
cd ~ # Перейти в домашнюю директорию /home/<username>.
pwd  # Вернёт /home/<username>.
cd .. # Перейти на уровень выше из текущей папки, т.е. в /home.
```

Создать новую директорию в текущей: **mkdir**

```
mkdir имя_директории
```

Создать новый файл в текущей директории: **touch**

```
touch имя_файла.расширение
```

Удалить файл: **rm**

```
rm имя_файла
```

Удалить пустую директорию: **rmdir**

```
rmdir empty_dir
```

Удалить не пустую директорию

```
rm -r notempty_dir
```

Ключи

Ключи используются для настройки команд.

Получить список ключей:

```
<название_команды> --help
```

В одной команде можно комбинировать несколько ключей:

```
# Показать файлы, в том числе скрытые (-a), с подробной информацией (-l)
# и отсортировать их по времени изменения (-t).
$ ls -alt
```

Редактирование файлов через nano

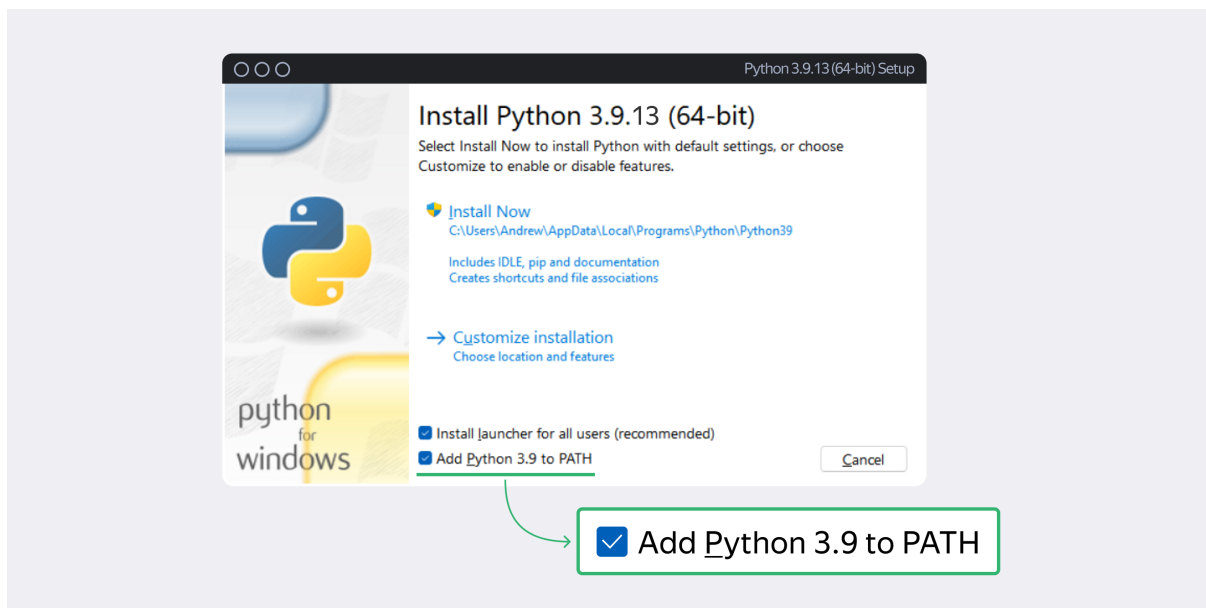
Открыть файл в редакторе *nano*:

`nano` имя_файла_с_расширением

Установка интерпретатора Python

На Windows

1. Скачайте версию интерпретатора python 3.9, выбрав разрядность своей ОС Windows.
2. Запустите установщик и на первом экране установите флажок для **Add Python 3.9 to Path**.



3. Нажмите **Install Now**.
4. Для проверки, что интерпретатор Python установлен и готов к работе, откройте терминал Git Bash и выполните команду:

```
python --version
```

В ответ должна вывестись версия только что установленного интерпретатора.

На macOS

Стандартный способ

1. Скачайте Python 3.9
2. Запустите загруженный файл и следуйте инструкциям по установке.

3. Проверьте версию Python через терминал командой `python3 --version`.

Альтернативный способ: с помощью пакетного менеджера `brew`

1. Установите `brew` по инструкции.
2. Добавьте `brew` в переменные окружения:

```
echo 'eval "$(/opt/homebrew/bin/brew shellenv)'" >> ~/.zprofile  
eval "$(/opt/homebrew/bin/brew shellenv)"
```

3. Установите Python 3.9 выполните команду:

```
brew install python@3.9
```

4. Проверьте версию Python через терминал командой `python3 --version`.
Вывод должен показать версию 3.9.

Если это не так, сделайте версию Python 3.9 версией по умолчанию с помощью команд:

```
# Отвяжите текущую версию.  
brew unlink python@3.текущая_версия  
# Установите новую привязку.  
brew link python@3.9
```

На Linux

На Ubuntu интерпретатор Python уже предустановлен. Чтобы узнать его версию, введите:

```
python3 --version
```

Если текущая версия 3.9, у вас всё готово к работе. Если нет, нужно её установить. Следуйте такому плану:

1. Обновите пакеты системы.
2. Установите нужную версию Python.
3. Убедитесь, что установка прошла успешно.

Откройте программу «Терминал» через палитру приложений или с помощью

сочетания клавиш **Ctrl+Alt+T**. Чтобы обновить пакеты системы, нужно использовать пакетный менеджер **apt**. Введите в терминал команду:

```
sudo apt update && sudo apt upgrade
```

Далее можно запускать установку нужной версии Python и пакетов для неё. Выполните последовательно команды:

```
sudo apt install python3.9 -y  
sudo apt install python3.9-venv
```

Убедитесь, что установка версии 3.9 прошла успешно. Выполните команду:

```
python3.9 --version
```

Установка VS Code

1. Скачайте установочный файл Visual Studio Code для вашей операционной системы и запустите его.
2. Откройте редактор кода, перейдите к пункту **Extensions** в боковом меню.
3. Установите два расширения:
 - **Python** — добавляет подсветку синтаксиса и ошибок, описание команд при наведении на них указателем мыши, расширенную поддержку языка Python для всех версий ≥ 3.7 и другие инструменты работы с кодом.
 - **Pylance** — отвечает за автодополнение кода и другие подсказки.

Настройка VS Code для работы с нужной версией Python

1. Откройте проект в VS Code и нажмите комбинацию горячих клавиш **Ctrl+Shift+P** (или **Command+Shift+P** для macOS). В появившейся строке начните печатать *python: sel*; выберите пункт **Python Select Interpreter**.
2. Далее выберите пункт **Select at workspace level**:
3. Появится список всех доступных версий Python. Выберите нужную версию, например Python 3.9.

Выбранная версия интерпретатора отображается в правой нижней части окна

Visual Studio Code.



Виртуальное окружение

Создать виртуальное окружение:

```
# Папка, в которой должна быть выполнена команда: .../Dev/first_project.
```

```
# Команда для Windows.
```

```
python -m venv venv
```

```
# Команда для Linux и macOS.
```

```
python3 -m venv venv
```

```
# Команда для Linux и macOS с указанием версии Python.
```

```
python3.9 -m venv venv
```

Активировать виртуальное окружение:

```
# Команда выполняется из директории с папкой venv.
```

```
# Команда для Windows:
```

```
source venv/Scripts/activate
```

```
# Для Linux и macOS:
```

```
source venv/bin/activate
```

Деактивировать виртуальное окружение:

```
deactivate
```

Установка модулей для Python

Установить сторонний модуль или библиотеку:

```
pip install <название модуля>
```

При установке можно указывать конкретную версию модуля:

```
pip install pytest==7.1.3
```

Обновить пакетный менеджер:

```
# Для Windows:
```

```
python -m pip install --upgrade pip
```

```
# Для Linux и macOS:
```

```
python3 -m pip install --upgrade pip
```

Посмотреть список модулей и библиотек в виртуальном окружении:

```
pip list
```

Файл с зависимостями

Зависимости проекта — это библиотеки и модули, которые необходимы для работы проекта.

Создать файл с зависимостями:

```
pip freeze > requirements.txt
```

Установить модули из файла *requirements.txt*:

```
pip install -r requirements.txt
```

Полезные ресурсы

Горячие клавиши VS Code: [Windows](#), [Linux](#), [macOS](#)

Шпаргалка — это карта,
помогающая мудрому
мореплавателю выбрать путь.

Ягдетто Там, испанский мореплаватель