

Скрипт Python. Инструкция для macOS

Шаг 1: Проверка Python

Откройте Терминал и выполните:

```
python3 --version
```

Если видите версию Python, переходите к шагу 2.

Если пишет "command not found", скачайте Python с python.org.

Шаг 2: Создание папки проекта

```
cd ~/Desktop  
mkdir python_project  
cd python_project
```

Шаг 3: Создание файла установки

Создайте файл `install.py`:

```
nano install.py
```

Скопируйте этот код и вставьте в файл `install.py`:

```
import subprocess  
import sys  
import os  
import io  
  
# Принудительно установить UTF-8 для вывода  
sys.stdout = io.TextIOWrapper(sys.stdout.buffer, encoding='utf-8')  
  
print("Установка инструментов Python", flush=True)  
print("-----")
```

```
# Проверка Python
print("1. Проверка Python...", flush=True)
result = subprocess.run([sys.executable, "--version"],
capture_output=True, text=True)
if result.returncode == 0:
    print("    Python найден: " + result.stdout.strip(), flush=True)
else:
    print("    Ошибка: Python не найден", flush=True)
    print(result, flush=True)
    sys.exit(1)

# Проверяем установлен ли pip
result = subprocess.run([sys.executable, '-m', 'pip', '--version'],
stdout=subprocess.PIPE, stderr=subprocess.PIPE, text=True)
if result.returncode == 0:
    print("    Pip найден:", result.stdout.strip(), flush=True)
else:
    print("    Ошибка: Pip не найден", flush=True)
    sys.exit(1)

# Проверяем установлен ли venv
result = subprocess.run([sys.executable, "-m", "venv", "--help"],
stdout=subprocess.DEVNULL, stderr=subprocess.DEVNULL, timeout=10)
if result.returncode == 0:
    print("    venv найден:", result, flush=True)
else:
    print("    Ошибка: venv не найден", flush=True)
    sys.exit(1)

# Папка может называться 'venv', а может '.venv'
root_dir_name = "venv" # название папки виртуального окружения

# Определяем путь к pip
if os.name == "nt": # Windows
    venv_python = os.path.join("venv", "Scripts", "python.exe")
else: # Unix/Linux/macOS
    venv_python = os.path.join("venv", "bin", "python")

# Создание виртуального окружения
if not os.path.exists(root_dir_name):
    print("\n2. Создание виртуального окружения...", flush=True)
```

```

    result = subprocess.run(["python3", "-m", "venv",
root_dir_name], capture_output=True, text=True)
    if result.returncode == 0:
        print("    Виртуальное окружение", root_dir_name, "создано",
flush=True)
    else:
        print("    Ошибка создания", root_dir_name, flush=True)
        print(result, flush=True)
        print("\n" + "=" * 50, flush=True)
        print("!!! РЕКОМЕНДАЦИЯ: Если вы используете Linux, то установите
python3.12-venv (если ещё не установлен). Если у вас другая версия Python –
замените 3.12 на вашу (например, 3.10, 3.11).", flush=True)
        print("Команда для установки: sudo apt install python3.12-venv",
flush=True)
        print("=" * 50, flush=True)
        sys.exit(1)
else:
    print("Виртуальное окружение", root_dir_name, "уже создано", flush=True)

# Установка инструментов
print("\n3. Установка инструментов...\n", flush=True)
tools = ["mypy", "pylint", "flake8", "black", "isort"]
for tool in tools:
    print("    Установка " + tool + "...", flush=True)
    cmd = [venv_python, "-m", "pip", "install", tool]
    result = subprocess.run(cmd, capture_output=True, text=True)
    if result.returncode == 0:
        print("    " + tool + " установлен", flush=True)
    else:
        print("    Ошибка установки " + tool, flush=True)
    print(flush=True)

print("\n4. Проверка установленных пакетов...", flush=True)
check_cmd = [venv_python, "-m", "pip", "list"]
result = subprocess.run(check_cmd, capture_output=True, text=True)
print("    Установленные пакеты:", flush=True)
packages = result.stdout.strip().split('\n')
for package in packages:
    if any(tool in package.lower() for tool in tools):
        print("    " + package, flush=True)

print("\n" + "=" * 50, flush=True)
print("УСТАНОВКА ЗАВЕРШЕНА", flush=True)
print("=" * 50, flush=True)

```

Шаг 4: Запуск скрипта установки

```
python3 install.py
```

Шаг 5: Дополнительная проверка

После установки активируйте окружение и проверьте пакеты:

```
source venv/bin/activate  
pip list
```

Проверка каждого пакета отдельно:

```
pip show mypy  
pip show pylint  
pip show flake8  
pip show black  
pip show isort
```

Инструкция для Windows

Шаг 1: Проверка Python

Откройте Командную строку и выполните:

```
python3 --version
```

Если видите версию Python, переходите к шагу 2.

Если пишет "не является внутренней или внешней командой...", скачайте Python с python.org и **обязательно отметьте галочку "Add Python to PATH"** при установке.

Шаг 2: Создание папки проекта

```
cd Desktop  
mkdir python_project  
cd python_project
```

Шаг 3: Создание файла установки

Создайте файл `install.py` :

```
notepad install.py
```

Скопируйте туда **тот же самый код**, что указан выше для macOS (он универсален и работает на Windows без изменений).

Шаг 4: Запуск скрипта установки

```
python3 install.py
```

Шаг 5: Дополнительная проверка

После установки активируйте окружение и проверьте пакеты:

```
venv\Scripts\activate  
pip list
```

Проверка каждого пакета отдельно:

```
pip show mypy  
pip show pylint  
pip show flake8  
pip show black  
pip show isort
```

Для обеих систем

Если все инструменты установлены правильно, вы должны увидеть в выводе строки, содержащие:

- Версию `mypy`
- Версию `pylint`
- Версию `flake8`
- Версию `black`
- Версию `isort`

Если какой-то инструмент не появляется в списке — установите его вручную внутри активированного виртуального окружения:

```
pip install имя_инструмента
```

Работу выполнили: Тоц Л. А. и Калинин И. В.

Группа: ИВТ-2