

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра инфо коммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.14
Дисциплины «Программирование на Python»

Выполнил:

Липчанский Дмитрий Сергеевич
1 курс, группа ИТС-б-о-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», очная форма обучения

(подпись)

Руководитель практики: Воронкин Р.
А, канд. техн. наук, доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Установка пакетов в Python. Виртуальные окружения

Цель работы: приобретение навыков по работе с менеджером пакетов `pip` и виртуальными окружениями с помощью языка программирования Python версии 3.x.

Ход работы:

1. Решил задачи с помощью языка программирования Python3. И отправил их на GitHub.
2. Установил `pip`.

```
(prog) C:\Users\Admin\Desktop\4_lr\prog> python3 -m venv env
(prog) C:\Users\Admin\Desktop\4_lr\prog> env\Scripts\activate
(env) (prog) C:\Users\Admin\Desktop\4_lr\prog> python3 -m venv /home/user/envs/project1_env
```

Рисунок 1 – Процесс создания виртуального окружения

3. Обновил `pip` с помощью команды `python -m pip install -Upip`

```
C:\Users\GO_PB>cd/d C:\Users\GO_PB\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Python 3.10
C:\Users\GO_PB\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Python 3.10>python -m pip install --upgrade pip
Requirement already satisfied: pip in c:\users\go_pb\appdata\local\programs\python\python310\lib\site-packages (22.0.4)
Collecting pip
  Downloading pip-22.3.1-py3-none-any.whl (2.1 MB)
    ----- 2.1/2.1 MB 3.0 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 22.0.4
    Uninstalling pip-22.0.4:
      Successfully uninstalled pip-22.0.4
Successfully installed pip-22.3.1

C:\Users\GO_PB\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Python 3.10>pip liat
ERROR: unknown command "liat" - maybe you meant "list"

C:\Users\GO_PB\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Python 3.10>pip list
Package            Version
-----
distlib             0.3.6
filelock            3.9.0
numpy               1.23.5
pandas              1.5.2
pip                 22.3.1
platformdirs        2.6.2
python-dateutil     2.8.2
pytz                2022.6
setuptools           58.1.0
```

Рисунок 2 – Обновление `pip`

```
(prog) C:\Users\Admin\Desktop\4_lr\prog>conda install pip, NumPy, Pandas, SciPy.  
Collecting package metadata (current_repodata.json): done  
Solving environment: failed with initial frozen solve. Retrying with flexible solve.  
Collecting package metadata (repodata.json): done  
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
```

Рисунок 3 – Установка в виртуальное окружение следующие пакеты: pip, NumPy, Pandas, SciPy

```
(prog) C:\Users\Admin\Desktop\4_lr\prog>pip freeze > requirements.txt  
(prog) C:\Users\Admin\Desktop\4_lr\prog>conda env export > environment.yml
```

Рисунок 4 – Сформированы файлы requirements.txt и environment.yml

Контрольные вопросы:

1. Каким способом можно установить пакет Python, не входящий в стандартную библиотеку?

Существует так называемый PythonPackageIndex (PyPI) – это репозиторий, открытый для всех Python разработчиков, в нем вы можете найти пакеты для решения практически любых задач.

2. Как осуществить установку менеджера пакетов pip? python -m pip< аргументы>

3. Откуда менеджер пакетов pip по умолчанию устанавливает пакеты?

4. Как установить последнюю версию пакета с помощью pip?python -m pip install -U pip

5. Как установить заданную версию пакета с помощью pip? pipinstallProjectName==3.2

6. Как установить пакет из git репозитория (в том числе GitHub) с помощью pip?

pip install -e git+https://gitrepo.com/ProjectName.git

7. Как установить пакет из локальной директории с помощью pip? pip install ./dist/ProjectName.tar.gz

8. Как удалить установленный пакет с помощью pip? pipuninstallProjectName

9. Как обновить установленный пакет с помощью pip? pipinstall -- upgradeProjectName

10. Как отобразить список установленных пакетов с помощью `pip`? `pip list`

11. Каковы причины появления виртуальных окружений в языке Python?

В отдельной папке создаётся неполная копия выбранной установки Python. Эта копия является просто набором файлов (например, интерпретатора или ссылки на него), утилит для работы с собой и нескольких пакетов (в том числе `pip`). Стандартные пакеты при этом не копируются.

12. Каковы основные этапы работы с виртуальными окружениями?

1) Создаём через утилиту новое виртуальное окружение в отдельной папке для выбранной

2) версии интерпретатора Python.

3) Активируем ранее созданное виртуального окружения для работы.

4) Работаем в виртуальном окружении, а именно управляем пакетами используя `pip` и

5) запускаем выполнение кода.

6) Деактивируем после окончания работы виртуальное окружение.

7) Удаляем папку с виртуальным окружением, если оно нам больше не нужно.

13. Как осуществляется работа с виртуальными окружениями с помощью `venv`?

Создав виртуальное окружение в папке проекта. После её выполнения создастся папка `env` с виртуальным окружением. После активации приглашение консоли изменится. В его начале в круглых скобках будет отображаться имя папки с виртуальным окружением.

14. Как осуществляется работа с виртуальными окружениями спомощью `virtualenv`?

Создание в текущей папке виртуального окружения для интерпретатора доступного через команду `python3` с названием папки окружения `env`.

freeze - команда, используемая для получения всех установленных пакетов в формате требований. Таким образом, все пакеты, которые вы установили перед выполнением команды и предположительно использовали в каком-либо проекте, будут перечислены в файле с именем

«requirements.txt». Кроме того, будут указаны их точные версии

15. Изучите работу с виртуальными окружениями `pipenv`. Как осуществляется работа с виртуальными окружениями `pipenv`?

При запуске проект с `Pipenv`, он автоматически создает виртуальную среду для текущего проекта, даже если вы еще не используете ее. `Pipenv` управляет зависимостями, отказавшись от привычного `requirements.txt`, и заменяя его на новый документ под названием `Pipfile`.

16. Каково назначение файла `requirements.txt`? Как создать этот файл? Какой он имеет формат?

Способ хранения списка внешних зависимостей проекта. `pipfreeze>`; `requirements.txt . < requirementspecifier>`

17. В чем преимущества пакетного менеджера `conda` по сравнению с пакетным менеджером `pip`?

`Conda` же способна управлять пакетами как для `Python`, так и для `C/ C++`, `R`, `Ruby`, `Lua`, `Scala` и других. `Conda` устанавливает двоичные файлы, поэтому работу по компиляции пакета самостоятельно выполнять не требуется (по сравнению с `pip`).

18. В какие дистрибутивы `Python` входит пакетный менеджер `conda`? `Anaconda`

19. Как создать виртуальное окружение `conda`? `conda create -n %PROJ_NAME% python=3.7 condaactivate %PROJ_NAME%`

20. Как активировать и установить пакеты в виртуальное окружение `conda`?

`condaactivateenv`

21. Как деактивировать и удалить виртуальное окружение `conda`?

`conda deactivate`

22. Каково назначение файла `environment.yml`? Как создать этот файл?

23. Как создать виртуальное окружение `conda` с помощью файла `environment.yml`?

Создайте окружение из `environment.yml` файла. Первая строка `yml` файла задает имя новой среды. Активируйте новую среду: `conda activate my env`. Убедитесь, что новая среда установлена правильно: `conda env list`

24. Самостоятельно изучите средства IDE PyCharm для работы с виртуальными окружениями `conda`. Опишите порядок работы с виртуальными окружениями `conda` в IDE PyCharm.

25. Почему файлы `requirements.txt` и `environment.yml` должны храниться в репозитории `git`?

Стандартные файлы виртуального окружения.

Вывод: приобретение навыков по работе с менеджером пакетов `pip` и виртуальными окружениями с помощью языка программирования Python версии 3.x

