

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

Институт цифрового развития  
Кафедра инфокоммуникаций

**«Установка пакетов в Python. Виртуальные  
окружения»**

**ОТЧЕТ**  
**по лабораторной работе №17**  
**дисциплины**  
**«Основы программной инженерии»**

Выполнил:

Сотников Андрей Александрович  
2 курс, группа ПИЖ-б-о-21-1,  
09.03.04 «Программная  
инженерия», направленность  
(профиль) «Разработка и  
сопровождение программного  
обеспечения», очная форма  
обучения

---

(подпись)

Проверил:

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

```
C:\lab-17>pip list
Package      Version
-----
pip          19.0.3
setuptools  40.8.0
You are using pip version 19.0.3, however version 23.0 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip'
command.
```

Рисунок 1 - пример использования pip

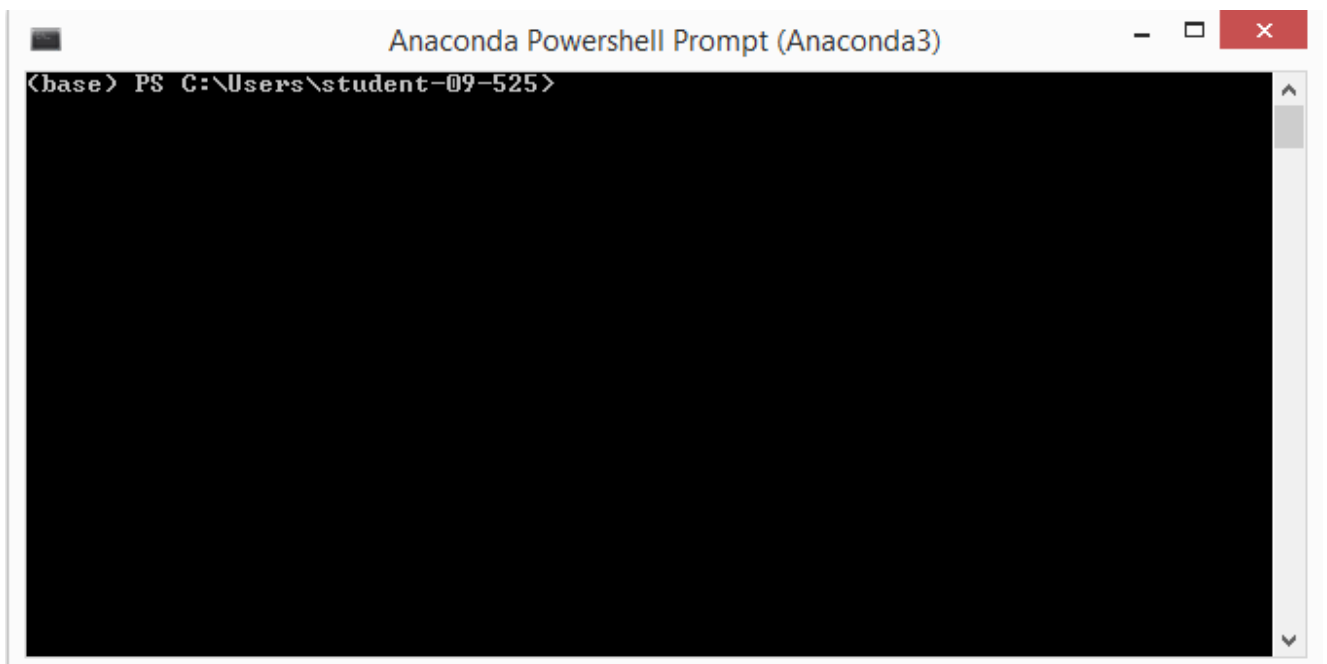


Рисунок 2 - окно Anaconda Powershell Prompt

```
Anaconda Powershell Prompt (Anaconda3)

Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 23.1.0
Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\.conda\envs\lab-17
  added / updated specs:
    - python=3.7

The following packages will be downloaded:

  package                                     | build
  -----
  ca-certificates-2023.01.10                | haa95532_0 121 KB
  certifi-2022.12.7                         | py37haa95532_0 149 KB
  openssl-1.1.1t                            | h2bbff1b_0 5.5 MB
  pip-22.3.1                                | py37haa95532_0 2.7 MB
  python-3.7.16                             | h6244533_0 17.2 MB
  setuptools-65.6.3                        | py37haa95532_0 1.1 MB
  sqlite-3.40.1                             | h2bbff1b_0 889 KB
  vc-14.2                                   | h21ff451_1 8 KB
  vs2015_runtime-14.27.29016               | h5e58377_2 1007 KB
  wheel-0.37.1                             | pyhd3eb1b0_0 33 KB
  wincertstore-0.2                         | py37haa95532_2 15 KB
  -----
  Total:                                     28.7 MB

The following NEW packages will be INSTALLED:

  ca-certificates pkgs/main/win-64::ca-certificates-2023.01.10-haa95532_0
  certifi         pkgs/main/win-64::certifi-2022.12.7-py37haa95532_0
  openssl         pkgs/main/win-64::openssl-1.1.1t-h2bbff1b_0
  pip            pkgs/main/win-64::pip-22.3.1-py37haa95532_0
  python         pkgs/main/win-64::python-3.7.16-h6244533_0
  setuptools     pkgs/main/win-64::setuptools-65.6.3-py37haa95532_0
  sqlite         pkgs/main/win-64::sqlite-3.40.1-h2bbff1b_0
  vc             pkgs/main/win-64::vc-14.2-h21ff451_1
  vs2015_runtime pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
  wheel          pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0
  wincertstore   pkgs/main/win-64::wincertstore-0.2-py37haa95532_2

Proceed [y]/n)? y

Downloading and Extracting Packages
ca-certificates-2023 | 121 KB | ##### | 100%
sqlite-3.40.1        | 889 KB | ##### | 100%
openssl-1.1.1t       | 5.5 MB | ##### | 100%
wincertstore-0.2     | 15 KB  | ##### | 100%
certifi-2022.12.7    | 149 KB | ##### | 100%
python-3.7.16        | 17.2 MB | ##### | 100%
vc-14.2              | 8 KB   | ##### | 100%
setuptools-65.6.3    | 1.1 MB | ##### | 100%
vs2015_runtime-14.27 | 1007 KB | ##### | 100%
pip-22.3.1           | 2.7 MB | ##### | 100%
wheel-0.37.1         | 33 KB  | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

# To activate this environment, use
#
#   $ conda activate lab-17
#
# To deactivate an active environment, use
#
#   $ conda deactivate
#

(base) PS C:\lab-17>
```

Рисунок 3 – создание виртуального окружение через conda

```
(base) PS C:\lab-17> conda activate lab-17
(lab-17) PS C:\lab-17>
```

Рисунок 4 – переход в виртуальное окружение

```

(base) PS C:\lab-17> conda activate lab-17
(lab-17) PS C:\lab-17> conda install django, pandas
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible
Solving environment: failed with repodata from current_repodata.json, will ret
with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 23.1.0

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\conda\envs\lab-17

  added / updated specs:
    - django
    - pandas

The following packages will be downloaded:



| package                 | build          |          |
|-------------------------|----------------|----------|
| asgiref-3.5.2           | py37haa95532_0 | 39 KB    |
| bottleneck-1.3.5        | py37h080aedc_0 | 105 KB   |
| django-3.2.15           | py37haa95532_0 | 4.1 MB   |
| flit-core-3.6.0         | pyhd3eb1b0_0   | 42 KB    |
| intel-openmp-2021.4.0   | haa95532_3556  | 2.2 MB   |
| krb5-1.19.4             | h5b6d351_0     | 786 KB   |
| libpq-12.9              | hb652d5d_3     | 2.7 MB   |
| mkl-2021.4.0            | haa95532_640   | 114.9 MB |
| mkl-service-2.4.0       | py37h2bbff1b_0 | 49 KB    |
| mkl_fft-1.3.1           | py37h277e83a_0 | 135 KB   |
| mkl_random-1.2.2        | py37hf11a4ad_0 | 216 KB   |
| numexpr-2.8.4           | py37h5b0cc5e_0 | 127 KB   |
| numpy-1.21.5            | py37h7a0a035_3 | 25 KB    |
| numpy-base-1.21.5       | py37hca35cd5_3 | 4.4 MB   |
| packaging-22.0          | py37haa95532_0 | 67 KB    |
| pandas-1.3.5            | py37h6214cd6_0 | 8.4 MB   |
| psycopy2-2.9.3          | py37hcd4344a_0 | 145 KB   |
| python-dateutil-2.8.2   | pyhd3eb1b0_0   | 233 KB   |
| pytz-2022.7             | py37haa95532_0 | 210 KB   |
| six-1.16.0              | pyhd3eb1b0_1   | 18 KB    |
| sqlparse-0.4.3          | py37haa95532_0 | 89 KB    |
| typing_extensions-4.4.0 | py37haa95532_0 | 46 KB    |
| zlib-1.2.13             | h8cc25b3_0     | 113 KB   |
| Total:                  |                | 139.0 MB |



The following NEW packages will be INSTALLED:

  asgiref                pkgs/main/win-64::asgiref-3.5.2-py37haa95532_0
  blas                   pkgs/main/win-64::blas-1.0-mkl

```

Рисунок 5 – установка пакетов Django и Pandas в виртуальное окружение lab-17

```
<lab-17> PS C:\lab-17> conda env export > environment.yml
C:\lab-17\environment.yml - Notepad+
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
environment.yml
1 name: lab-17
2 channels:
3   - defaults
4 dependencies:
5   - asgiref=3.5.2=py37haa95532_0
6   - blas=1.0=mkl
7   - bottleneck=1.3.5=py37h080aedc_0
8   - ca-certificates=2023.01.10=haa95532_0
9   - certifi=2022.12.7=py37haa95532_0
10  - django=3.2.15=py37haa95532_0
11  - flit-core=3.6.0=pyhd3eb1b0_0
12  - intel-openmp=2021.4.0=haa95532_3556
13  - krb5=1.19.4=h5b6d351_0
14  - libpq=12.9=hb652d5d_3
15  - mkl=2021.4.0=haa95532_640
16  - mkl-service=2.4.0=py37h2bbff1b_0
17  - mkl_fft=1.3.1=py37h277e83a_0
18  - mkl_random=1.2.2=py37hf11a4ad_0
19  - numexpr=2.8.4=py37h5b0cc5e_0
20  - numpy=1.21.5=np37h7a0a035_3
```

Рисунок 6 – экспорт параметров окружения в файл environment.yml

```
Anaconda Powershell Prompt (Anaconda3)
(base) PS C:\Users\student-09-525> cd /lab-17
(base) PS C:\lab-17> conda activate lab-17
(lab-17) PS C:\lab-17> pip freeze > requirements.txt
(lab-17) PS C:\lab-17>

requirements.txt — Блокнот
Файл Правка Формат Вид Справка
asgiref @ file:///C:/Users/BUILDE~1/AppData/Local/Temp/abs_cchykh5sgr/croot/
Bottleneck @ file:///C:/Windows/Temp/abs_3198ca53-903d-42fd-87b4-03e6d03a8:
certifi @ file:///C:/b/abs_85o_6fm0se/croot/certifi_1671487778835/work/cert
Django @ file:///C:/Users/BUILDE~1/AppData/Local/Temp/abs_e2vdu8snoo/croot:
flit_core @ file:///opt/conda/conda-bld/flit-core_1644941570762/work/source
mkl-fft==1.3.1
mkl-random @ file:///C:/ci/mkl_random_1626186163140/work
mkl-service==2.4.0
numexpr @ file:///C:/b/abs_a7kbak88hk/croot/numexpr_1668713882979/work
numpy @ file:///C:/ci/numpy_and_numpy_base_1653574840943/work
packaging @ file:///C:/b/abs_cfsup8ur87/croot/packaging_1671697442297/work
pandas @ file:///C:/ci/pandas_1641461312795/work
psycopg2 @ file:///C:/b/abs_6cdgqr8yu7/croot/psycopg2_1666817457048/work
python-dateutil @ file:///tmp/build/80754af9/python-dateutil_1626374649649,
pytz @ file:///C:/b/abs_22fofvpn1x/croot/pytz_1671698059864/work
six @ file:///tmp/build/80754af9/six_1644875935023/work
sqlparse @ file:///C:/b/abs_83jn6owfpc/croot/sqlparse_1668793499314/work
typing_extensions @ file:///C:/b/abs_89eui86zuq/croot/typing_extensions_166
wincertstore==0.2
```

Рисунок 7 – создание файла requirements.txt

```

(lab-17) PS C:\lab-17> conda install tensorflow
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible
lve.
Solving environment: failed with repodata from current_repodata.json, will re
with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 23.1.0

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\.conda\envs\lab-17

  added / updated specs:
    - tensorflow

The following packages will be downloaded:


```

package	build	
_tfflow_select-2.2.0	eigen	3 KB
absl-py-1.3.0	py37haa95532_0	170 KB
aiohttp-3.8.3	py37h2bbff1b_0	411 KB
aiosignal-1.2.0	pyhd3eb1b0_0	12 KB
astunparse-1.6.3	py_0	17 KB
async-timeout-4.0.2	py37haa95532_0	12 KB
asynctest-0.13.0	py_0	26 KB
attrs-22.1.0	py37haa95532_0	84 KB
blinker-1.4	py37haa95532_0	23 KB
brotlipy-0.7.0	py37h2bbff1b_1003	337 KB
cachetools-4.2.2	pyhd3eb1b0_0	13 KB
cffi-1.15.1	py37h2bbff1b_3	236 KB
charset-normalizer-2.0.4	pyhd3eb1b0_0	35 KB
click-8.0.4	py37haa95532_0	153 KB
colorama-0.4.6	py37haa95532_0	32 KB
cryptography-38.0.4	py37h21b164f_0	1.0 MB
fftw-3.3.9	h2bbff1b_1	672 KB
flatbuffers-2.0.0	h6c2663c_0	1.4 MB
frozenlist-1.3.3	py37h2bbff1b_0	40 KB
gast-0.4.0	pyhd3eb1b0_0	13 KB
giflib-5.2.1	h8cc25b3_1	81 KB
google-auth-2.6.0	pyhd3eb1b0_0	83 KB
google-auth-oauthlib-0.4.4	pyhd3eb1b0_0	18 KB
google-pasta-0.2.0	pyhd3eb1b0_0	46 KB
grpcio-1.42.0	py37hc60d5dd_0	1.8 MB
h5py-3.7.0	py37h3de5c98_0	800 KB
hdf5-1.10.6	h1756f20_1	7.9 MB
icc_rt-2022.1.0	h6049295_2	6.5 MB
idna-3.4	py37haa95532_0	92 KB
importlib-metadata-4.11.3	py37haa95532_0	40 KB
jpeg-9e	h2bbff1b_0	292 KB
keras-2.10.0	py37haa95532_0	1.5 MB
keras-preprocessing-1.1.2	pyhd3eb1b0_0	35 KB
libcurl-7.87.0	h86230a5_0	324 KB
libprotobuf-3.20.3	h23ce68f_0	2.2 MB
libssh2-1.10.0	hcd4344a_0	222 KB
markdown-3.4.1	py37haa95532_0	148 KB
markupsafe-2.1.1	py37h2bbff1b_0	26 KB
multidict-6.0.2	py37h2bbff1b_0	45 KB
oauthlib-3.2.1	py37haa95532_0	193 KB
opt_einsum-3.3.0	pyhd3eb1b0_1	57 KB
protobuf-3.20.3	py37hd77b12b_0	231 KB
pyasn1-0.4.8	pyhd3eb1b0_0	54 KB
pyasn1-modules-0.2.8	py_0	72 KB
pycparser-2.21	pyhd3eb1b0_0	94 KB
pyjwt-2.4.0	py37haa95532_0	38 KB
pyopenssl-22.0.0	pyhd3eb1b0_0	50 KB
pysocks-1.7.1	py37_1	28 KB

Рисунок 8 – установка TensorFlow через conda

```

(lab-17) PS C:\lab-17> pip install tensorflow
Requirement already satisfied: tensorflow in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (2.10.0)
Requirement already satisfied: keras-preprocessing>=1.1.1 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (1.1.2)
Requirement already satisfied: keras<2.11,>=2.10.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (2.10.0)
Requirement already satisfied: tensorboard<2.11,>=2.10 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (2.10.0)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (0.4.0)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (1.3.0)
Requirement already satisfied: h5py>=2.9.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (3.7.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (1.6.3)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (1.42.0)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (4.4.0)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (0.2.0)
Requirement already satisfied: flatbuffers>=2.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (2.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (3.3.0)
Requirement already satisfied: setuptools in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (65.6.3)
Collecting protobuf<3.20,>=3.9.2
  Downloading protobuf-3.19.6-cp37-cp37m-win_amd64.whl (896 kB)
      896.6/896.6 kB 2.6 MB/s eta 0:00:00
Requirement already satisfied: six>=1.12.0 in c:\users\student-09-525\conda\envs\lab-17\lib\site-packages (from tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\student-09-525\cond

```

Рисунок 9 – установка TensorFlow через pip

## Контрольные вопросы

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

Для этого необходимо использовать команду `pip install <название_пакета>`

### 2. Как осуществить установку менеджера пакетов `pip`?

Начиная с версии Python 3.4, менеджер пакетов `pip` устанавливается автоматически вместе с Python. Если по какой либо причине у вас стоит более старая версия Python, то установить `pip` вы сможете через скрипт `get-pip.py`, командой `python get-pip.py`, предварительно скачав скрипт из интернета

### 3. Откуда менеджер пакетов `pip` по умолчанию устанавливает пакеты?

По умолчанию менеджер пакетов `pip` устанавливает пакеты из основного индекса PyPI.

### 4. Как установить последнюю версию пакета с помощью `pip`?

Команда `pip install <название_пакета>` по умолчанию скачивает последнюю версию требуемого пакета.

### 5. Как установить заданную версию пакета с помощью `pip`?

Команда `pip install <название_пакета>==3.2` устанавливает требуемый пакет с версий 3.2

### 6. Как установить пакет из `git`-репозитория с помощью `pip`?

Для этого используется команда  
`pip install -e git+https://gitrepo.com/ProjectName.git`

### 7. Как установить пакет из локальной директории с помощью `pip`?

Для этого используется команда `pip install ./dist/ProjectName.tar.gz`

### 8. Как удалить установленный пакет с помощью `pip`?

`pip uninstall <название_пакета>`

### 9. Как обновить установленный пакет с помощью `pip`?

`Pip install --upgrade ProjectName`

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

`Pip list`

### 11. Каковы причины появления виртуальных окружений в языке Python?

Это проблема обратной совместимости и коллективной разработки, виртуальные окружения решают эти проблемы

### 12. Каковы основные этапы работы с виртуальными окружениями?

Создание виртуального окружения, его активация, деактивация

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

Работа происходит через командную строку. После создания виртуальной



машины появляется папка `env`, с установленными пакетами

**14. Как осуществляется работа с виртуальными окружениями с помощью `virtualenv`?**

Зачем нам нужно уметь работать с утилитой `virtualenv`? Ведь мы уже научились работать со стандартным модулем Python `venv`. Просто он очень распространён и поддерживает большее число вариантов и версий интерпретатора Python, например, PyPy и CPython.

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

Как `pip`, только удобнее.

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

Само виртуальное окружение никуда переносить не нужно. Требуется возможность формирования и развертывания пакетных зависимостей. Для формирования и развертывания пакетных зависимостей используется утилита `pip`.

Просмотреть список зависимостей мы можем командой:

```
pip freeze
```

Что бы его сохранить, нужно перенаправить вывод команды в файл:

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

Имя файла хранения зависимостей `requirements.txt` выбрано не зря. Оно является стандартной договоренностью и используется некоторыми утилитами автоматически.

Установка пакетов из файла зависимостей в новом виртуальном окружении так же выполняется одной командой:

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

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

Основная проблема заключается в том, что `pip`, `easy_install` и `virtualenv` ориентированы на Python. Эти инструменты игнорируют библиотеки зависимостей, реализованные с использованием других языков. Например, XSLT, HDF5, MKL и другие, которые не имеют `setup.py` в исходном коде и не устанавливают файлы в директорию `site-packages`. 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`). Существуют также некоторые различия, если вы заинтересованы в

создании собственных пакетов. Например, `pip` создан на основе `setuptools`, тогда как `conda` использует свой собственный формат, который имеет некоторые преимущества (например, статическая компиляция пакета).

## 18. В какие дистрибутивы Python входит пакетный менеджер conda?

Все чаще среди Python-разработчиков заходит речь о менеджере пакетов `conda`, включенный в состав дистрибутивов `Anaconda` и `Miniconda`. `JetBrains` включил этот инструмент в состав `PyCharm`.

## 19. Как создать виртуальное окружение conda?

Для Windows, если используется дистрибутив `Anaconda`, то необходимо вначале запустить консоль `Anaconda Powershell Prompt`. Делается это из системного меню, посредством выбора следующих пунктов: *Пуск → Anaconda3 (64-bit) → Anaconda Powershell Prompt (Anaconda3)*. В результате будет отображено окно консоли, показанное на рисунке.

Обратите на имя виртуального окружения по умолчанию, которым в данном случае является `base`. В этом окне необходимо ввести следующую последовательность команд:

```
mkdir %PROJ_NAME%  
cd %PROJ_NAME%  
copy NUL > main.py
```

2. Создайте чистое `conda`-окружение с таким же именем, как директория проекта, и затем активируйте его. Для Linux это последовательность команд:

```
source deactivate  
conda create -n $PROJ_NAME python=3.7  
source activate $PROJ_NAME
```

Тогда как для Windows эта последовательность будет несколько иной:

```
conda create -n %PROJ_NAME% python=3.7  
conda activate %PROJ_NAME%
```

После выполнения этих команд в приглашении ввода должно отобразиться имя созданного виртуального окружения.

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

3. Установите пакеты, необходимые для реализации проекта.

```
conda install django, pandas
```

4. Периодически экспортируйте параметры окружения. Экспортируйте после установки, перед каждым большим или маленьким коммитом:

```
conda env export > environment.yml
```

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

Для Windows необходимо использовать следующую команду:

```
conda deactivate
```

Если вы хотите удалить только что созданное окружение, выполните:

```
conda remove -n $PROJ_NAME
```

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

Файл environment.yml позволит воссоздать окружение в любой нужный момент.

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

6. Файл `environment.yml` позволит воссоздать окружение в любой нужный момент.  
Достаточно набрать:

```
conda env create -f environment.yml
```

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

[Установка Anaconda + интеграция с Pycharm - Русские Блоги \(russianblogs.com\)](http://russianblogs.com)

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

Эти файлы дают возможность восстановить виртуальное окружение на другом устройстве/компьютере.