

Федеральное государственное автономное образовательное учреждение
высшего образования «Северо-Кавказский федеральный университет»
Институт цифрового развития

ОТЧЕТ ПО ДИСЦИПЛИНЕ
ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ
Лабораторная работа №2

Выполнила:

Мирзаева Камилла Мирзаевна
2 курс, группа ПИЖ-б-о-20-1

Принял:

Воронкин Роман Александрович

Ставрополь, 2022 г.

ВЫПОЛНЕНИЕ

```
(base) PS C:\Users\student-09-525> conda create -n laba2_14
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.11.0

Please update conda by running

    $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\.conda\envs\laba2_14

Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
##
## To activate this environment, use
##
##     $ conda activate laba2_14
##
## To deactivate an active environment, use
##
##     $ conda deactivate
```

Рисунок 1 - Создание виртуального окружения

```
(base) PS C:\Users\student-09-525> conda activate laba2_14
(laba2_14) PS C:\Users\student-09-525>
```

Рисунок 2 - Активация виртуального окружения

```
(laba2_14) PS C:\Users\student-09-525> conda install -n laba2_14 pip
Collecting package metadata (current_repodata.json): done
Solving environment: done
```

```
=> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.11.0
```

Please update conda by running

```
$ conda update -n base -c defaults conda
```

Package Plan

environment location: C:\Users\student-09-525\.conda\envs\laba2_14

added / updated specs:
- pip

The following packages will be downloaded:

package	build	
ca-certificates-2022.2.1	haa95532_0	123 KB
certifi-2021.10.8	py39haa95532_2	152 KB
openssl-1.1.1m	h2bbff1b_0	4.8 MB
pip-21.2.4	py39haa95532_0	1.8 MB
python-3.9.7	h6244533_1	16.5 MB
setuptools-58.0.4	py39haa95532_0	778 KB
sqlite-3.37.2	h2bbff1b_0	799 KB
tzdata-2021e	hda174b7_0	112 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	py39haa95532_2	15 KB
Total:		26.1 MB

The following NEW packages will be INSTALLED:

ca-certificates	pkgs/main/win-64::ca-certificates-2022.2.1-haa95532_0
certifi	pkgs/main/win-64::certifi-2021.10.8-py39haa95532_2
openssl	pkgs/main/win-64::openssl-1.1.1m-h2bbff1b_0
pip	pkgs/main/win-64::pip-21.2.4-py39haa95532_0
python	pkgs/main/win-64::python-3.9.7-h6244533_1
setuptools	pkgs/main/win-64::setuptools-58.0.4-py39haa95532_0
sqlite	pkgs/main/win-64::sqlite-3.37.2-h2bbff1b_0
tzdata	pkgs/main/noarch::tzdata-2021e-hda174b7_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-py39haa95532_2

Рисунок 3 - Пример установки пакета pip

```

(laba2_14) PS C:\Users\student-09-525> conda install -n laba2_14 numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

```

```

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.11.0

```

Please update conda by running

```
$ conda update -n base -c defaults conda
```

Package Plan

environment location: C:\Users\student-09-525\.conda\envs\laba2_14

added / updated specs:
- numpy

The following packages will be downloaded:

package	build	
intel-openmp-2021.4.0	haa95532_3556	2.2 MB
mkl-2021.4.0	haa95532_640	114.9 MB
mkl-service-2.4.0	py39h2bbff1b_0	51 KB
mkl_fft-1.3.1	py39h277e83a_0	139 KB
mkl_random-1.2.2	py39hf11a4ad_0	225 KB
numpy-1.21.5	py39ha4e8547_0	9 KB
numpy-base-1.21.5	py39hc2deb75_0	4.4 MB
six-1.16.0	pyhd3eb1b0_1	18 KB
Total:		121.8 MB

The following NEW packages will be INSTALLED:

blas	pkgs/main/win-64::blas-1.0-mkl
intel-openmp	pkgs/main/win-64::intel-openmp-2021.4.0-haa95532_3556
mkl	pkgs/main/win-64::mkl-2021.4.0-haa95532_640
mkl-service	pkgs/main/win-64::mkl-service-2.4.0-py39h2bbff1b_0
mkl_fft	pkgs/main/win-64::mkl_fft-1.3.1-py39h277e83a_0
mkl_random	pkgs/main/win-64::mkl_random-1.2.2-py39hf11a4ad_0
numpy	pkgs/main/win-64::numpy-1.21.5-py39ha4e8547_0
numpy-base	pkgs/main/win-64::numpy-base-1.21.5-py39hc2deb75_0
six	pkgs/main/noarch::six-1.16.0-pyhd3eb1b0_1

Рисунок 4 - Установка пакета NumPy

```
(laba2_14) PS C:\Users\student-09-525> conda install -n laba2_14 pandas
Collecting package metadata (current_repodata.json): done
Solving environment: done
```

```
==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.11.0
```

Please update conda by running

```
$ conda update -n base -c defaults conda
```

Package Plan

environment location: C:\Users\student-09-525\.conda\envs\laba2_14

added / updated specs:

- pandas

The following packages will be downloaded:

package	build	
bottleneck-1.3.2	py39h7cc1a96_1	107 KB
numexpr-2.8.1	py39hb80d3ca_0	117 KB
packaging-21.3	pyhd3eb1b0_0	36 KB
pandas-1.4.1	py39hd77b12b_0	8.9 MB
pyparsing-3.0.4	pyhd3eb1b0_0	81 KB
python-dateutil-2.8.2	pyhd3eb1b0_0	233 KB
pytz-2021.3	pyhd3eb1b0_0	171 KB
Total:		9.6 MB

The following NEW packages will be INSTALLED:

bottleneck	pkgs/main/win-64::bottleneck-1.3.2-py39h7cc1a96_1
numexpr	pkgs/main/win-64::numexpr-2.8.1-py39hb80d3ca_0
packaging	pkgs/main/noarch::packaging-21.3-pyhd3eb1b0_0
pandas	pkgs/main/win-64::pandas-1.4.1-py39hd77b12b_0
pyparsing	pkgs/main/noarch::pyparsing-3.0.4-pyhd3eb1b0_0
python-dateutil	pkgs/main/noarch::python-dateutil-2.8.2-pyhd3eb1b0_0
pytz	pkgs/main/noarch::pytz-2021.3-pyhd3eb1b0_0

Рисунок 5 - Установка пакета Pandas

```

(laba2_14) PS C:\Users\student-09-525> conda install -n laba2_14 scipy
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.11.0

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\.conda\envs\laba2_14

  added / updated specs:
    - scipy

The following packages will be downloaded:



| package     | build          |         |
|-------------|----------------|---------|
| scipy-1.7.3 | py39h0a974cb_0 | 13.9 MB |
| Total:      |                | 13.9 MB |



The following NEW packages will be INSTALLED:

  icc_rt                pkgs/main/win-64::icc_rt-2019.0.0-h0cc432a_1
  scipy                 pkgs/main/win-64::scipy-1.7.3-py39h0a974cb_0

```

Рисунок 6 - Установка пакета Scipy

```

(laba2_14) PS C:\Users\student-09-525> conda list
# packages in environment at C:\Users\student-09-525\.conda\envs\laba2_14:
#
# Name                   Version           Build    Channel
blas                     1.0               mkl
bottleneck               1.3.2             py39h7cc1a96_1
ca-certificates          2022.2.1          haa95532_0
certifi                  2021.10.8         py39h0a974cb_0
icc_rt                   2019.0.0          h0cc432a_1
intel-openmp             2021.4.0          haa95532_3556
mkl                      2021.4.0          haa95532_640
mkl-service              2.4.0             py39h2bbff1b_0
mkl_fft                  1.3.1             py39h277e83a_0
mkl_random               1.2.2             py39hf11a4ad_0
numexpr                  2.8.1             py39hb80d3ca_0
numpy                    1.21.5            py39ha4e8547_0
numpy-base               1.21.5            py39hc2deb75_0
openssl                  1.1.1m            h2bbff1b_0
packaging                 21.3              pyhd3eb1b0_0
pandas                   1.4.1             py39hd77b12b_0
pip                       21.2.4            py39h0a974cb_0
pyparsing                3.0.4             pyhd3eb1b0_0
python                   3.9.7             h6244533_1
python-dateutil          2.8.2             pyhd3eb1b0_0
pytz                     2021.3            pyhd3eb1b0_0
scipy                    1.7.3             py39h0a974cb_0
setuptools               58.0.4            py39h0a974cb_0
six                       1.16.0            pyhd3eb1b0_1
sqlite                   3.37.2            h2bbff1b_0
tzdata                   2021e             hda174b7_0
vc                       14.2              h21ff451_1
vs2015_runtime           14.27.29016       h5e58377_2
wheel                    0.37.1            pyhd3eb1b0_0
wincertstore             0.2               py39h0a974cb_0

```

Рисунок 7 - Список пакетов

```

(laba2_14) PS C:\Users\student-09-525> conda install -n laba2_14 tensorflow
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry 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: 4.11.0

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\student-09-525\.conda\envs\laba2_14

  added / updated specs:
    - tensorflow

The following packages will be downloaded:



| package                  | build             | size   |
|--------------------------|-------------------|--------|
| _tflow_select-2.3.0      | mk1               | 3 KB   |
| abseil-cpp-20210324.2    | hd77b12b_0        | 1.6 MB |
| absl-py-0.15.0           | pyhd3eb1b0_0      | 103 KB |
| aiohttp-3.8.1            | py39h2bbff1b_1    | 496 KB |
| aiohttp-3.8.1            | pyhd3eb1b0_0      | 12 KB  |
| astor-0.8.1              | py39haa95532_0    | 47 KB  |
| astunparse-1.6.3         | py_0              | 17 KB  |
| async-timeout-4.0.1      | pyhd3eb1b0_0      | 10 KB  |
| attrs-21.4.0             | pyhd3eb1b0_0      | 51 KB  |
| blinker-1.4              | py39haa95532_0    | 23 KB  |
| brotlipy-0.7.0           | py39h2bbff1b_1003 | 411 KB |
| cachetools-4.2.2         | pyhd3eb1b0_0      | 13 KB  |
| cffi-1.15.0              | py39h2bbff1b_1    | 224 KB |
| charset-normalizer-2.0.4 | pyhd3eb1b0_0      | 35 KB  |
| click-8.0.4              | py39haa95532_0    | 155 KB |
| colorama-0.4.4           | pyhd3eb1b0_0      | 21 KB  |
| cryptography-3.4.8       | py39h71e12ea_0    | 638 KB |
| dataclasses-0.8          | pyh6d0b6a4_7      | 8 KB   |
| flatbuffers-2.0.0        | h6c2663c_0        | 1.4 MB |
| frozenlist-1.2.0         | py39h2bbff1b_0    | 77 KB  |
| gast-0.4.0               | pyhd3eb1b0_0      | 13 KB  |
| giflib-5.2.1             | h62dc97_0         | 81 KB  |
| google-auth-2.6.0        | pyhd3eb1b0_0      | 83 KB  |


```

Рисунок 8 - Установка пакета Tensorflow

```

(laba2_14) PS C:\Users\student-09-525> python
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow

```

Рисунок 9 - Запуск Python

```

(laba2_14) PS C:\Users\student-09-525\.conda\envs\laba2_14> pip freeze
absl-py @ file:///opt/conda/conda-bld/absl-py_1639803114343/work
aiohttp @ file:///C:/ci/aiohttp_1646806572557/work
aiosignal @ file:///tmp/build/80754af9/aiosignal_1637843061372/work
astor==0.8.1
astunparse==1.6.3
async-timeout @ file:///tmp/build/80754af9/async-timeout_1637851218186/work
attrs @ file:///opt/conda/conda-bld/attrs_1642510447205/work
blinker==1.4
Bottleneck @ file:///C:/ci/bottleneck_1607557040328/work
brotlipy==0.7.0
cachetools @ file:///tmp/build/80754af9/cachetools_1619597386817/work
certifi==2021.10.8
cffi @ file:///C:/ci_310/cffi_1642682485096/work
charset-normalizer @ file:///tmp/build/80754af9/charset-normalizer_163000322965/work
click @ file:///C:/ci/click_1646038595831/work
colorama @ file:///tmp/build/80754af9/colorama_1607707115595/work
cryptography @ file:///C:/ci/cryptography_1633520531101/work
flatbuffers @ file:///tmp/build/80754af9/python-flatbuffers_1614345733764/work
frozenlist @ file:///C:/ci/frozenlist_1637767271796/work
gast @ file:///Users/ktietz/demo/mc3/conda-bld/gast_1628588903283/work
google-auth @ file:///opt/conda/conda-bld/google-auth_1646735974934/work
google-auth-oauthlib==0.4.1
google-pasta @ file:///Users/ktietz/demo/mc3/conda-bld/google-pasta_163057799134/work
grpcio @ file:///C:/ci/grpcio_1637590978642/work
h5py @ file:///C:/ci/h5py_1637120894255/work
idna @ file:///tmp/build/80754af9/idna_1637925883363/work
importlib-metadata @ file:///C:/ci/importlib-metadata_1638543082309/work
Keras-Preprocessing @ file:///tmp/build/80754af9/keras-preprocessing_161228364096/work
Markdown @ file:///C:/ci/markdown_1614364082838/work
mkl-fft==1.3.1
mkl-random @ file:///C:/ci/mkl_random_1626186184308/work
mkl-service==2.4.0
multidict @ file:///C:/ci/multidict_1607349747897/work
numexpr @ file:///C:/ci/numexpr_1640704337920/work
numpy @ file:///C:/ci_310/numpy_and_numpy_base_1643798589088/work
oauthlib @ file:///opt/conda/conda-bld/oauthlib_1644332107998/work
opt-einsum @ file:///tmp/build/80754af9/opt_einsum_1621500238896/work
packaging @ file:///tmp/build/80754af9/packaging_1637314298585/work
pandas @ file:///C:/ci/pandas_1644984307868/work
protobuf==3.14.0
pyasn1 @ file:///Users/ktietz/demo/mc3/conda-bld/pyasn1_1629708007385/work
pyasn1-modules==0.2.8
pyparsing @ file:///tmp/build/80754af9/pyparsing_1636541352034/work
PyJWT @ file:///C:/ci/pyjwt_1619682721924/work
pyOpenSSL @ file:///tmp/build/80754af9/pyopenssl_1635333100036/work
pyparsing @ file:///tmp/build/80754af9/pyparsing_1635766073266/work
pyreadline==2.1
PySocks @ file:///C:/ci/pysocks_1605307512533/work
python-dateutil @ file:///tmp/build/80754af9/python-dateutil_1626374649649/work
pytz==2021.3
requests @ file:///opt/conda/conda-bld/requests_1641824580448/work
requests-oauthlib==1.3.0
rsa @ file:///tmp/build/80754af9/rsa_1614366226499/work
scipy @ file:///C:/ci/scipy_1641555170412/work
six @ file:///tmp/build/80754af9/six_1644875935023/work
tensorboard @ file:///tmp/build/80754af9/tensorboard_1633093581375/work/tensorb

```

Рисунок 10 - Создание файла requirements.txt

```

(laba2_14) PS C:\Users\student-09-525\.conda\envs\laba2_14> pip freeze > requirements.txt
(laba2_14) PS C:\Users\student-09-525\.conda\envs\laba2_14>

```

Рисунок 11 - Создание файла requirements.txt

```

(laba2_14) PS C:\Users\student-09-525\.conda\envs\laba2_14> conda env export > enviromants.yml

```

Рисунок 12 - Создание файла enviroments.yml

```
abs1-py @ file:///opt/conda/conda-bld/abs1-py_1639803114343/work
aiohttp @ file:///C:/ci/aiohttp_1646806572557/work
aiosignal @ file:///tmp/build/80754af9/aiosignal_1637843061372/work
astor==0.8.1
astunparse==1.6.3
async-timeout @ file:///tmp/build/80754af9/async-timeout_1637851218186/work
attrs @ file:///opt/conda/conda-bld/attrs_1642510447205/work
blinker==1.4
Bottleneck @ file:///C:/ci/bottleneck_1607557040328/work
brotlipy==0.7.0
cachetools @ file:///tmp/build/80754af9/cachetools_1619597386817/work
certifi==2021.10.8
cffi @ file:///C:/ci_310/cffi_1642682485096/work
charset-normalizer @ file:///tmp/build/80754af9/charset-normalizer_1630003229654/work
click @ file:///C:/ci/click_1646038595831/work
colorama @ file:///tmp/build/80754af9/colorama_1607707115595/work
cryptography @ file:///C:/ci/cryptography_1633520531101/work
flatbuffers @ file:///tmp/build/80754af9/python-flatbuffers_1614345733764/work
frozenlist @ file:///C:/ci/frozenlist_1637767271796/work
gast @ file:///Users/ktietz/demo/mc3/conda-bld/gast_1628588903283/work
google-auth @ file:///opt/conda/conda-bld/google-auth_1646735974934/work
google-auth-oauthlib==0.4.1
google-pasta @ file:///Users/ktietz/demo/mc3/conda-bld/google-pasta_1630577991354/work
grpcio @ file:///C:/ci/grpcio_1637590978642/work
h5py @ file:///C:/ci/h5py_1637120894255/work
idna @ file:///tmp/build/80754af9/idna_1637925883363/work
importlib-metadata @ file:///C:/ci/importlib-metadata_1638543082309/work
Keras-Preprocessing @ file:///tmp/build/80754af9/keras-preprocessing_1612283640596/work
Markdown @ file:///C:/ci/markdown_1614364082838/work
mkl-fft==1.3.1
mkl-random @ file:///C:/ci/mkl_random_1626186184308/work
mkl-service==2.4.0
multidict @ file:///C:/ci/multidict_1607349747897/work
numexpr @ file:///C:/ci/numexpr_1640704337920/work
numpy @ file:///C:/ci_310/numpy_and_numpy_base_1643798589088/work
oauthlib @ file:///opt/conda/conda-bld/oauthlib_1644332107998/work
opt-einsum @ file:///tmp/build/80754af9/opt_einsum_1621500238896/work
packaging @ file:///tmp/build/80754af9/packaging_1637314298585/work
pandas @ file:///C:/ci/pandas_1644984307868/work
protobuf==3.14.0
pyasn1 @ file:///Users/ktietz/demo/mc3/conda-bld/pyasn1_1629708007385/work
pyasn1-modules==0.2.8
pyparsing @ file:///tmp/build/80754af9/pyparsing_1636541352034/work
PyJWT @ file:///C:/ci/pyjwt_1619682721924/work
pyOpenSSL @ file:///tmp/build/80754af9/pyopenssl_1635333100036/work
pyparsing @ file:///tmp/build/80754af9/pyparsing_1635766073266/work
pyreadline==2.1
PySocks @ file:///C:/ci/pysocks_1605307512533/work
python-dateutil @ file:///tmp/build/80754af9/python-dateutil_1626374649649/work
pytz==2021.3
requests @ file:///opt/conda/conda-bld/requests_1641824580448/work
requests-oauthlib==1.3.0
```

Рисунок 13 - Содержимое файла requirements.txt

```

name: laba2_14
channels:
  - defaults
dependencies:
  - _tflow_select=2.3.0=mk1
  - abseil-cpp=20210324.2=hd77b12b_0
  - absl-py=0.15.0=pyhd3eb1b0_0
  - aiohttp=3.8.1=py39h2bbff1b_1
  - aiosignal=1.2.0=pyhd3eb1b0_0
  - astor=0.8.1=py39haa95532_0
  - astunparse=1.6.3=py_0
  - async-timeout=4.0.1=pyhd3eb1b0_0
  - attrs=21.4.0=pyhd3eb1b0_0
  - blas=1.0=mk1
  - blinker=1.4=py39haa95532_0
  - bottleneck=1.3.2=py39h7cc1a96_1
  - brotli=1.0.7=py39h2bbff1b_1003
  - ca-certificates=2022.2.1=haa95532_0
  - cachetools=4.2.2=pyhd3eb1b0_0
  - certifi=2021.10.8=py39haa95532_2
  - cffi=1.15.0=py39h2bbff1b_1
  - charset-normalizer=2.0.4=pyhd3eb1b0_0
  - click=8.0.4=py39haa95532_0
  - colorama=0.4.4=pyhd3eb1b0_0
  - cryptography=3.4.8=py39h71e12ea_0
  - dataclasses=0.8=pyh6d0b6a4_7
  - flatbuffers=2.0.0=h6c2663c_0
  - frozenlist=1.2.0=py39h2bbff1b_0
  - gast=0.4.0=pyhd3eb1b0_0
  - giflib=5.2.1=h62dcd97_0
  - google-auth=2.6.0=pyhd3eb1b0_0
  - google-auth-oauthlib=0.4.1=py_2
  - google-pasta=0.2.0=pyhd3eb1b0_0
  - grpcio=1.42.0=py39hc60d5dd_0
  - h5py=3.6.0=py39h3de5c98_0
  - hdf5=1.10.6=h7ebc959_0
  - icc_rt=2019.0.0=h0cc432a_1
  - icu=68.1=h6c2663c_0
  - idna=3.3=pyhd3eb1b0_0
  - importlib-metadata=4.8.2=py39haa95532_0
  - intel-openmp=2021.4.0=haa95532_3556
  - jpeg=9d=h2bbff1b_0
  - keras-preprocessing=1.1.2=pyhd3eb1b0_0
  - libcurl=7.80.0=h86230a5_0
  - libpng=1.6.37=h2a8f88b_0
  - libprotobuf=3.14.0=h23ce68f_0
  - libssh2=1.9.0=h7a1dbc1_1
  - markdown=3.3.4=py39haa95532_0
  - mk1=2021.4.0=haa95532_640
  - mk1-service=2.4.0=py39h2bbff1b_0
  - mk1_fft=1.3.1=py39h277e83a_0
  - mk1_random=1.2.2=py39hf11a4ad_0

```

Рисунок 14 - Содержимое файла enviroments.yml

Ответы на вопросы:

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

Существует так называемый Python Package Index (PyPI) – это репозиторий, открытый для всех Python разработчиков, в нем вы можете найти пакеты для решения практически любых задач. Для скачивания и установки используется специальная утилита, которая называется `pip`.

2. Как осуществить установку менеджера пакетов `pip`?

Универсальный способ: скачать скрипт `get-pip.py`

```
$ curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py  
  
$ python get-pip.py
```

При этом, вместе с `pip` будут установлены `setuptools` и `wheels`. `Setuptools` – это набор инструментов для построения пакетов Python. `Wheels` – это формат дистрибутива для пакета Python.

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

По умолчанию в Linux `Pip` устанавливает пакеты в `/usr/local/lib/python2.7/dist-packages`. Использование `virtualenv` или `—user` во время установки изменит это местоположение по умолчанию. Важный момент: по умолчанию `pip` устанавливает пакеты глобально. Это может привести к конфликтам между версиями пакетов.

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

```
$ pip install ProjectName
```

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

```
$ pip install ProjectName==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`?

```
$ pip uninstall ProjectName
```

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

```
$ pip install --upgrade ProjectName
```

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

Для вывода списка всех установленных пакетов применяется команда `pip list`. Если необходимо получить более подробную информацию о

конкретном пакете `pip show ProjectName`

11. Каковы причины появления виртуальных окружений в языке Python?

Если разработчик работает над проектом не один, а с командой, ему нужно передавать и получать список зависимостей, а также обновлять их на своем компьютере таким образом, чтобы не нарушалась работа других его проектов. Значит нужен механизм, который вместе с обменом проектами быстро устанавливал бы локально и все необходимые для них пакеты, при этом не мешая работе других проектов. Попыток было несколько, но в основу PEP 405 легла утилита `virtualenv` Яна Бикинга. Были проанализированы возникающие при работе с ней проблемы. После этого в работу интерпретатора Python версии 3.3 добавили их решения. Так был создан встроенный в Python модуль `venv`, а утилита `virtualenv` теперь дополнительно использует в своей работе и его.

12. Каковы основные этапы работы с виртуальными окружениями?

- 1) Создаём через утилиту новое виртуальное окружение в отдельной папке для выбранной версии интерпретатора Python.
- 2) Активируем ранее созданное виртуального окружения для работы.
- 3) Работаем в виртуальном окружении, а именно управляем пакетами используя `pip` и запускаем выполнение кода.
- 4) Деактивируем после окончания работы виртуальное окружение.
- 5) Удаляем папку с виртуальным окружением, если оно нам больше не нужно.

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

Создадим виртуальное окружение в папке проекта. Для этого перейдём в корень любого проекта на Python ≥ 3.3 и дадим команду:

```
$ python3 -m venv env
```

Чтобы активировать виртуальное окружение под Windows нужно дать команду:

```
> env\Scripts\activate
```

После активации приглашение консоли изменится. В его начале в круглых скобках будет отображаться имя папки с виртуальным окружением.

При размещении виртуального окружения в папке проекта стоит позаботиться об его исключении из репозитория системы управления версиями.

```
$ python3 -m venv /home/user/envs/project1_env
```

Чтобы переключиться с одного окружения на другое нам нужно выполнить команду деактивации и команду активации другого виртуального окружения.

```
$ deactivate
$ source /home/user/envs/project1_env2/bin/activate
```

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

Для начала пакет нужно установить. Установку можно выполнить командой:

```
# Для python 3
python3 -m pip install virtualenv

# Для единственного python
python -m pip install virtualenv
```

Создание в текущей папке виртуального окружения для интерпретатора доступно через команду python3 с названием папки окружения env:

```
virtualenv -p python3 env
```

Активация и деактивация такая же, как у стандартной утилиты Python.

```
> env\\Scripts\\activate

(env) > deactivate
```

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

Грубо говоря, pipenv можно рассматривать как симбиоз утилит pip и venv (или virtualenv), которые работают вместе, пряча многие неудобные детали от конечного пользователя.

Pipenv умеет:

- автоматически находить интерпретатор Python нужной версии (находит даже интерпретаторы, установленные через pyenv и asdf!);
- запускать вспомогательные скрипты для разработки;
- загружать переменные окружения из файла .env;
- проверять зависимости на наличие известных уязвимостей.

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

Просмотреть список зависимостей можно командой: 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

Расширение: .txt

18. В какие дистрибутивы Python входит пакетный менеджер conda?

Anaconda, miniconda и PyCharm.

19. Как создать виртуальное окружение conda?

Начиная проект, создайте чистую директорию и дайте ей понятное короткое имя.

Для Windows необходимо вначале запустить консоль Anaconda Powershell Prompt..

В этом окне необходимо ввести следующую последовательность команд:

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

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

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

Установите пакеты, необходимые для реализации проекта. `conda install django, pandas.`

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?

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

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

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

```
conda env export > enviromant.yml
```

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

Создавать отдельное окружение Conda и устанавливать только нужные библиотеки для каждого проекта. PyCharm позволяет легко создавать и выбирать правильное окружение.

25. Почему файлы `requirements.txt` и `environment.yml` должны храниться в репозитории git? Предоставляет доступ другим пользователям к файлам.