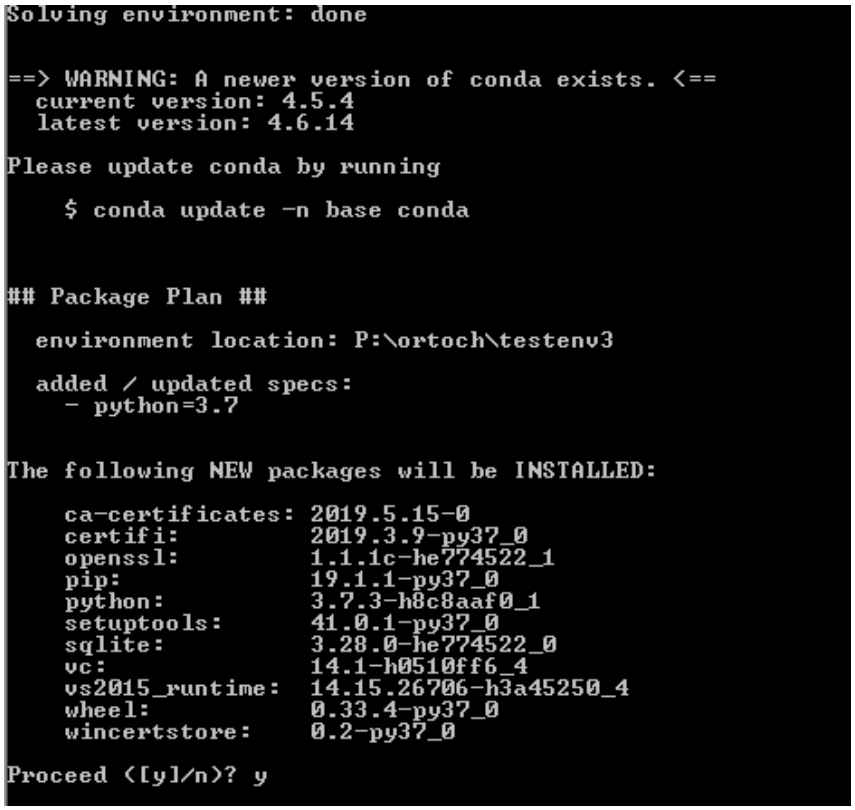
Step-by-step guide to Python package install in virtual environments

1. Open Anaconda Prompt (to manage existing environments outside of command line, Anaconda Navigator can also be used) – You will be presented with the view <base> C:\Users\username>
2. To create a new environment within your user folder on the P drive (hugely recommended) (change "ortoch" to your own username and "testenv3" to your own environment name):

The following will occur if successful:



3. 'y' to proceed with environment creation:
4. The following will occur if successful:

```
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate P:\ortoch\testenv3
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

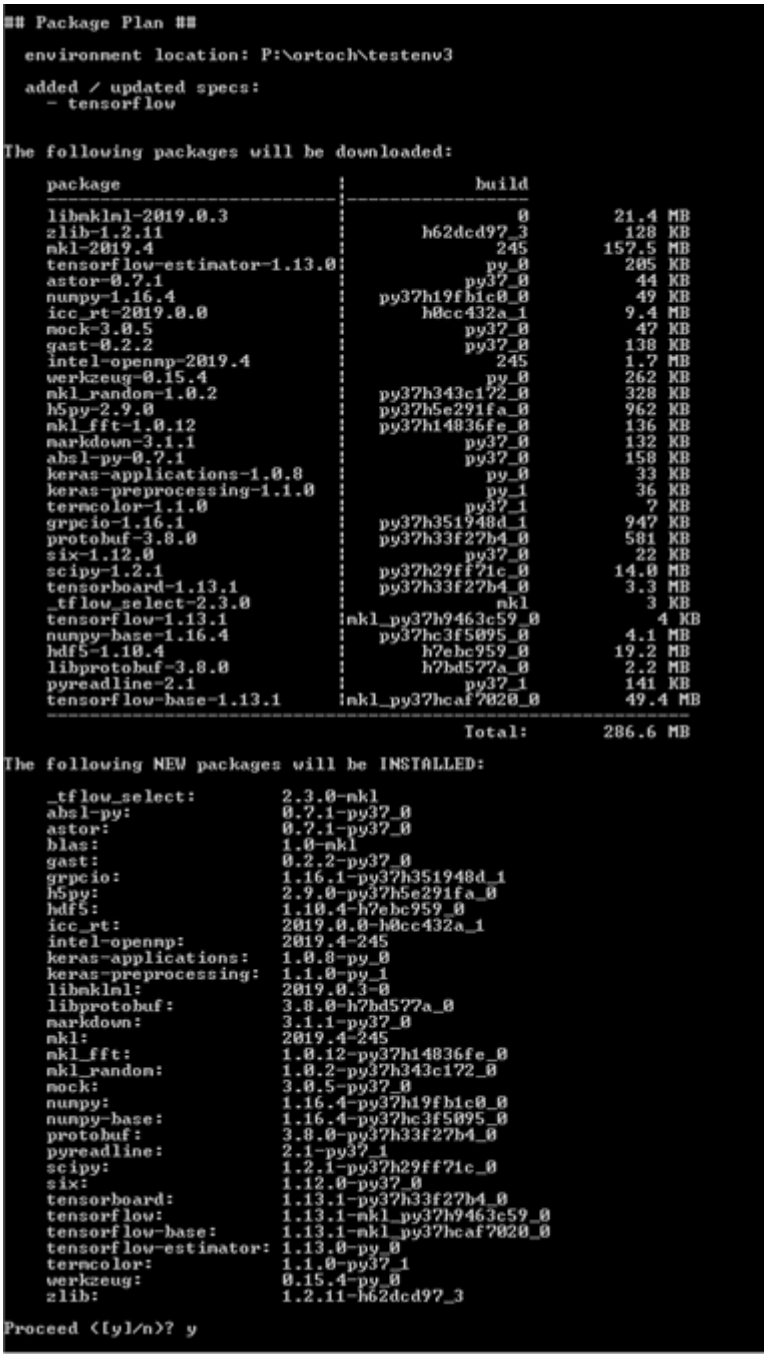
5. You can check that the package is in the correct path by running conda info –envs
6. Activate the created environment as follows (replace ortoch with own username and testenv3 with own environment name:

```
(base) C:\Users\ortoch>conda activate P:\ortoch\testenv3
```

7. Once activated, the path will now have your environment preceding the root, you can install a package by using: 'conda install packagename':

```
(P:\ortoch\testenv3) C:\Users\ortoch>conda install tensorflow
```

8. An example of the code that will appear if package installation is proceeding successfully; 'y' to proceed:



```
## Package Plan ##

  environment location: P:\ortoch\testenv3

  added / updated specs:
    - tensorflow


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    libmklml-2019.0.3          |                0        21.4 MB
    zlib-1.2.11                |        h62dcd97_3         128 KB
    mkl-2019.4                 |              245       157.5 MB
    tensorflow-estimator-1.13.0|             py_0         205 KB
    astor-0.7.1                |            py37_0          44 KB
    numpy-1.16.4               |   py37h19fb1c0_0          49 KB
    icc_rt-2019.0.0            |        h0cc432a_1         9.4 MB
    mock-3.0.5                 |            py37_0          47 KB
    gast-0.2.2                 |            py37_0         138 KB
    intel-openmp-2019.4        |              245         1.7 MB
    werkzeug-0.15.4            |             py_0         262 KB
    mkl_random-1.0.2           |   py37h343c172_0         328 KB
    h5py-2.9.0                 |   py37h5e291fa_0         962 KB
    mkl_fft-1.0.12             |   py37h14836fe_0         136 KB
    markdown-3.1.1             |            py37_0         132 KB
    absl-py-0.7.1              |            py37_0         158 KB
    keras-applications-1.0.8   |             py_0          33 KB
    keras-preprocessing-1.1.0  |             py_1          36 KB
    termcolor-1.1.0            |            py37_1           7 KB
    grpcio-1.16.1              |   py37h351948d_1         947 KB
    protobuf-3.8.0             |   py37h33f27b4_0         581 KB
    six-1.12.0                 |            py37_0          22 KB
    scipy-1.2.1                |   py37h29ff71c_0        14.0 MB
    tensorboard-1.13.1         |   py37h33f27b4_0         3.3 MB
    _tflow_select-2.3.0        |              mkl           3 KB
    tensorflow-1.13.1          |mkl_py37h9463c59_0           4 KB
    numpy-base-1.16.4          |   py37hc3f5095_0         4.1 MB
    hdf5-1.10.4                |        h7ebc959_0        19.2 MB
    libprotobuf-3.8.0          |        h7bd577a_0         2.2 MB
    pyreadline-2.1             |            py37_1         141 KB
    tensorflow-base-1.13.1     |mkl_py37hcaf7020_0        49.4 MB
    ------------------------------------------------------------
                                           Total:       286.6 MB

The following NEW packages will be INSTALLED:

    _tflow_select:        2.3.0-mkl
    absl-py:              0.7.1-py37_0
    astor:                0.7.1-py37_0
    blas:                 1.0-mkl
    gast:                 0.2.2-py37_0
    grpcio:               1.16.1-py37h351948d_1
    h5py:                 2.9.0-py37h5e291fa_0
    hdf5:                 1.10.4-h7ebc959_0
    icc_rt:               2019.0.0-h0cc432a_1
    intel-openmp:         2019.4-245
    keras-applications:   1.0.8-py_0
    keras-preprocessing:  1.1.0-py_1
    libmklml:             2019.0.3-0
    libprotobuf:          3.8.0-h7bd577a_0
    markdown:             3.1.1-py37_0
    mkl:                  2019.4-245
    mkl_fft:              1.0.12-py37h14836fe_0
    mkl_random:           1.0.2-py37h343c172_0
    mock:                 3.0.5-py37_0
    numpy:                1.16.4-py37h19fb1c0_0
    numpy-base:           1.16.4-py37hc3f5095_0
    protobuf:             3.8.0-py37h33f27b4_0
    pyreadline:           2.1-py37_1
    scipy:                1.2.1-py37h29ff71c_0
    six:                  1.12.0-py37_0
    tensorboard:          1.13.1-py37h33f27b4_0
    tensorflow:           1.13.1-mkl_py37h9463c59_0
    tensorflow-base:      1.13.1-mkl_py37hcaf7020_0
    tensorflow-estimator: 1.13.0-py_0
    termcolor:            1.1.0-py37_1
    werkzeug:             0.15.4-py_0
    zlib:                 1.2.11-h62dcd97_3

Proceed ([y]/n)? y
```

9. If appearing successful, before using an IDE such as Spyder/Jupyter Notebook to start a project, you can check libraries have been loaded correctly as follows in Anaconda Prompt:

python

>>> import tensorflow

If no errors return, the library is successfully installed.

10. You can also use Anaconda Navigator to view all packages in an environment – as shown here:

    You can use Anaconda Navigator to install a number of other associated software, such as glueviz, Jupyterlab, Jupyter Notebook, orange3, qtconsole, rstudio (already installed on VDI but can be integrated into Python), spyder, and vscode