Python Programming Language Notes

Timothy J. Helton
720.641.8370
timothy.j.helton@gmail.com

June 16, 2016

Contents

List of Figures

List of Tables

Nomenclature

pip	Python Package Index

1 Built-In Functions

1.1 abs()

Input must be an **int** , **float** or **complex number** .

- Returns the absolute value if the argument is a float or int.
- Returns the magnitude if the argument is a complex number.

1.2 all()

Input must be an **iterable** .

- Returns True if:
  - all elements of the iterable are true
  - the iterable is **empty**

1.3 any()

Input must be an **iterable** .

- Returns True if:
  - any of the elements of the iterable are true
- Returns False if:
  - the iterable is **empty**

1.4 ascii()

Input is an **object** .

Use this function to display a printable representation of an object, similar to repr(), but use escapes for non-ascii characters.

1.5 bin()

Input must be an **int** or an **object with a __index__() method** that returns an int.

Convert an integer to a binary string.

2 numpy

2.1 Record Array

2.1.1 Create from list of tuples

- First create an ndarray from a list of tuples
  - Must be a list of tuples (list of lists will not work)
  - The data type is very import in this case. Each item in the dtype corresponds to the item in the tuple.
  - The last field of the dtype for each column is the size of the data. Without this argument the conversion to an array will create multiple empty columns in the array.
- After the ndarray is created make a view as a recarray.

```
arr = np.ndarray(list_of_tuples,
                 dtype=[(col1_name, dtype1, (1,)), (col2_name, dtype2, (1,))])
rec_arr = arr.view(np.recarray)
```

3 pandas

4 pip - Python Package Index

4.1 Install pip

4.1.1 Without Internet Connection

1. Download **get-pip.py**
2. On a computer with an internet connection create the following wheels.
   - pip
   - setuptools
3. Move the wheels to the computer without an internet connection.
4. Call the following command

```
python get-pip.py --no-index --find-links=WheelHouseDirectory
```

4.2 Find the Site Packages Installation Directory

```
python -c "from distutils.sysconfig
           import get_python_lib;
           print get_python_lib()"
```

4.3 Install Packages

4.3.1 Install a Single Package

```
pip install PackageName
```

4.3.2 Install Packages From requirments.txt File

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

4.3.3 Install a Package From Wheel

- If you do not want to use the cached wheel then add the following argument.

```
--no-cache-dir
```

```
pip install --use-wheel --no-index --find-links=WheelHouseDirectory PackageName
```

4.3.4 Install Package in Developer Mode

This option allows a package to be actively developed while being installed in a Python interpreter.

```
pip install -e .
```

4.4 List Outdated Modules

```
pip list -o
```

4.5 Change the Version of an Installed Package

4.5.1 Upgrade to the Latest Version

```
pip install --upgrade PackageName
```

4.5.2 Install a Previous Version

```
pip install --upgrade PackageName==Version
```

4.6 Create Wheel

```
pip wheel --wheel-dir=WheelHouseDirectory
```

4.7 Package Configuration

The package configuration may be maintained by a single text file, which is commonly called requirements.txt. Boolean operations may also be used in these configuration files. To create a snapshot of the current interpreter use the following command.

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

5 psycopg2

6 scipy

References