Numpy Library

- Processing N-Dimensional arrays

In [5]:

```
import numpy as np


li = [1,2,3,'z']

a = np.array(li)

b = np.arange(15)
b
```

Out[5]:

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

In [8]:

```
rn = np.random.randint(0,100, size=10)
rn
```

Out[8]:

```
array([66, 60, 69,  6, 21, 40, 93, 27,  0, 14])
```

In [15]:

```
m3 = np.random.randint(0,2,size=(3,3,3,3))
m3[2][2][2][1]

```

Out[15]:

```
0
```

In [20]:

```
m3.ndim

```

Out[20]:

```
4
```

In [21]:

```
m3.size

```

Out[21]:

```
81
```

In [22]:

```
m3.shape

```

Out[22]:

```
(3, 3, 3, 3)
```

In [23]:

```
m3.dtype

```

Out[23]:

```
dtype('int32')
```

In [24]:
```
m3.itemsize

```

Out[24]:
```
4
```

In [25]:
```
m3.nbytes
```

Out[25]:
```
324
```

In [29]:
```
print(b)
```

```
[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14]
```

In [31]:
```
b.reshape(5,3)
```

Out[31]:
```
array([[ 0,  1,  2],
       [ 3,  4,  5],
       [ 6,  7,  8],
       [ 9, 10, 11],
       [12, 13, 14]])
```

Pandas

Use Cases

- Data Cleaning
- Data Transformation
- Data Analysis

Notations

- Series (Row)
- DataFrames (Table)

In [5]:
```
import pandas as pd

internal1 = {'s1':22, 's2':18, 's3':24}

internal1 = pd.Series(internal1)

internal2 = {'s1':15, 's2':18, 's3':12}
internal2 = pd.Series(internal2)
internal2


```

Out[5]:
```
s1    15
s2    18
s3    12
dtype: int64
```

In [8]:

```
final = {'Internal1':internal1, 'Internal2':internal2}

final = pd.DataFrame(final)
final

```

Out[8]:

	Internal1	Internal2
s1	22	15
s2	18	18
s3	24	12

In [9]:

```
final.columns #### Names of all columns
```

Out[9]:

```
Index(['Internal1', 'Internal2'], dtype='object')
```

In [10]:

```
final.values ### Lists of all rows
```

Out[10]:

```
array([[22, 15],
       [18, 18],
       [24, 12]], dtype=int64)
```

In [11]:

```
final.values[2]
```

Out[11]:

```
array([24, 12], dtype=int64)
```

In [28]:

```
final.values[2,0] =25
final.values[2][0]
```

Out[28]:

```
25
```

In [21]:

```
for row in final.values:
    print('Internal - ',row[0],' Internal2 - ',row[1],)
```

```
Internal -  22  Internal2 -  15
Internal -  18  Internal2 -  18
Internal -  24  Internal2 -  12
```

In [23]:

```
final.loc[3] =[15,18]
final
```

Out[23]:

	Internal1	Internal2
s1	22	15
s2	18	18
s3	24	12
3	15	18

In [24]:

```
final.loc['s4'] = [11,32]
final
```

Out[24]:

	Internal1	Internal2
s1	22	15
s2	18	18
s3	24	12
3	15	18
s4	11	32

In [27]:

```
final.drop('s3')
```

Out[27]:

	Internal1	Internal2
s1	22	15
s2	18	18
3	15	18
s4	11	32

In [30]:

```
final.values[2] = [18,15]
final
```

Out[30]:

	Internal1	Internal2
s1	22	15
s2	18	18
s3	18	15
3	15	18
s4	11	32

In [1]:

```
##### Reading CSV file

import pandas as pd
filepath = 'DataFiles/income.csv'
incomedf = pd.read_csv(filepath)
incomedf
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-1-690172f162e8> in <module>
      3 import pandas as pd
      4 filepath = 'DataFiles/income.csv'
----> 5 incomedf = pd.read_csv(filepath)
      6 incomedf

~\Anaconda3\lib\site-packages\pandas\io\parsers.py in parser_f(filepath_or_buff
er, sep, delimiter, header, names, index_col, usecols, squeeze, prefix, mangle_
dupe_cols, dtype, engine, converters, true_values, false_values, skipinitialspa
ce, skiprows, skipfooter, nrows, na_values, keep_default_na, na_filter, verbos
e, skip_blank_lines, parse_dates, infer_datetime_format, keep_date_col, date_pa
rser, dayfirst, iterator, chunksize, compression, thousands, decimal, linetermi
nator, quotechar, quoting, doublequote, escapechar, comment, encoding, dialect,
tupleize_cols, error_bad_lines, warn_bad_lines, delim_whitespace, low_memory, m
emory_map, float_precision)
    700                     skip_blank_lines=skip_blank_lines)
    701 
--> 702         return _read(filepath_or_buffer, kwds)
    703 
    704     parser_f.__name__ = name

~\Anaconda3\lib\site-packages\pandas\io\parsers.py in _read(filepath_or_buffer,
kwds)
    427 
    428     # Create the parser.
--> 429     parser = TextFileReader(filepath_or_buffer, **kwds)
    430 
    431     if chunksize or iterator:

~\Anaconda3\lib\site-packages\pandas\io\parsers.py in __init__(self, f, engine,
**kwds)
    893             self.options['has_index_names'] = kwds['has_index_names']
    894 
--> 895         self._make_engine(self.engine)
    896 
    897     def close(self):

~\Anaconda3\lib\site-packages\pandas\io\parsers.py in _make_engine(self, engin
e)
   1120     def _make_engine(self, engine='c'):
   1121         if engine == 'c':
-> 1122             self._engine = CParserWrapper(self.f, **self.options)
   1123         else:
   1124             if engine == 'python':

~\Anaconda3\lib\site-packages\pandas\io\parsers.py in __init__(self, src, **kwd
s)
   1851         kwds['usecols'] = self.usecols
   1852
-> 1853         self._reader = parsers.TextReader(src, **kwds)
   1854         self.unnamed_cols = self._reader.unnamed_cols
   1855

pandas/_libs/parsers.pyx in pandas._libs.parsers.TextReader.__cinit__()

pandas/_libs/parsers.pyx in pandas._libs.parsers.TextReader._setup_parser_sourc
e()

FileNotFoundError: [Errno 2] File b'DataFiles/income.csv' does not exist: b'Dat
aFiles/income.csv'
```

```
In [ ]:
```