Numpy

- The general purpose of the numpy library

  We need to import the numpy package so that the array or the data can work with them to use the function in numpy package.

- What numpy arrays are and what kinds of things they can (and cant contain)

  Numpy arrays are a scientific data structure that can be used to store data as a grid, or a matrix, the numpy array can be one-dimensional (list), two-dimensional (matrix) and mutil-dimensional.

  The data type in numpy array can be **int, float, str.** In a numpy array, the data type should be all the same type.

- Various approaches for how to slice a numpy array (e.g. grabbing out a row, column, range of values)

  - For one-dimensional numpy arrays, you only need to specify one index value, which is the position of the element in the numpy array, e.g. arrayname[index].

    Range of values: [starting_value, ending_value]

  - For two-dimensional numpy arrays, you need to specify both a row index and a column index for the element, e.g. [row_index, column_index].

    Grabbing a column: arrayname[**:** , colum_index] (selecting the column and including all rows)

    Grabbing a row: arrayname[row_index , **:**] (selecting the row and including all column)

    Range of values: [start_row_index:end_row_index, start_column_index:end_column_index] (Note: index structure for both the row and column range is inclusive of the first index, but not the second index).

- How to create a numpy array from scratch (please show at least 2 options)

  We can use some functions such as **np.zero()**, the array with all values 0; **np.ones()**, the array with all values 1; **np.arrange(10)**, the array with values from 0 to 9; **np.random.rand(4,5),** 4x5 array with random float between 0-1, **np.random.randint(10, size=(3,3))**, 3x3 array with random ints between 0-4.

We can also import the .txt and .csv files formats to the numpy array.

- List 5-6 helpful numpy functions and what they do

  np.reshape(3,4) - Reshape the array to 3 rows and 4 columns without changing data.

  np.append(arrayname,values) – Appends values to the end of array.

  np.mean(), np.max(), np.min(), np.quantile() – Calculate the mean, max min and the quantile of the data. When the data is missing in array we need to add `nan` e.g. np.nanmean(), np.nanmax() etc.

  np.sum(), np.std() – Calculate the sum of the elements and the standard deviation of the array.

  np.concentration(), np.hsplit() – combining / splitting the array.

Pandas

- The general purpose of the Pandas library

  Pandas is also a package used for data analysis and it is more convenient for us to slice the data we want and more functions in pandas library can raise efficiency when analyzing data.

- What makes a pandas data frame is different from a numpy array

Numpy Array	Pandas Dataframe
Array only has same type data	Can have different type data (int, float, str)
Have no row and column name	Have rows and columns name
Slice data: [start:stop;start:stop]	Slice data: [start:stop;start:stop], e.g. iloc; index using names e.g. loc
Can not handle the missing data	Can handle the missing data as blank

- An explanation of what the index of a dataframe and why its different from other columns

  Index is an optional argument, the DataFrame has an index attribute that gives access to the index labels, other columns are more like the data in a numpy array. And we can slice the data we want by using the index of different rows.

- How to setup a pandas dataframe by reading a file
  We need to ` **import pandas as pd**` first.

  Using the `**pd.read_csv()**` or `**pd.read_table()**` function from the pandas package, you can import tabular data from CSV or TXT files into pandas dataframe by specifying a parameter value for the file name

- How to set the index of a pandas dataframe

  We can use `df.index` function to know what index of the dataframe and can use df.set_index('column') to change the index.

- How to slice a pandas dataframe:
  - using loc and iloc to get rows

    iloc is the same way as we do in numpy array, **loc** is searching the data by the index and column name.

  - grabbing out columns by name or number

    Select data in a pandas dataframe based on specific values within a column using: **dataframe[dataframe["column"] == value]**

- 5-6 helpful pandas functions or methods that you can use to inspect your dataframe (list each and explain what it does)

  Info – Get the details like size and names.

  Describe – Provides summary statistics on all numeric columns.

  Groupby - Group by one of the columns and then you can do some operations from there.

  Df1.append(df2) – Add the rows in df1 to the end of df2 and they should have the same columns.

  Df.head(n), df.tail(n) – Seclet the first/last n rows.