# Midterm-2022

```
- Name:Abhay Nimavat
- Student Number:N01530302
```

**Example1: (1 Mark)**

> 1. create a dictionary for the consumer items which contains prices of 3 some items and names ,
>    convert the above dictionary into a series.

```python
In [1]: import pandas as pd
        import numpy as np
        books = {1 : 'Games', 2 : 'Sports', 3 : 'History'}
        books
```

```
Out[1]: {1: 'Games', 2: 'Sports', 3: 'History'}
```

```python
In [2]: books_series = pd.Series(books)
        print(books_series)
```

```
1       Games
2      Sports
3     History
dtype: object
```

**Example2: (2 Mark)**

- 1.Create 2 - 3 X 3 Arrays and add them up.
- 2.create a 3rd 6x6 array and multiply the entire array with 5 and then slice it into half
  horizontally.

```python
In [11]:
        a1 = np.array([1,2,3,4,5,6,7,8,9])
        a2 = np.array([10,11,12,13,14,15,16,17,18])
        arr1 = a1.reshape(3,3)
        arr2= a2.reshape(3,3)
        arr1+arr2
```

```
Out[11]: array([[11, 13, 15],
               [17, 19, 21],
               [23, 25, 27]])
```

In [14]:
```python
arr3 = np.arange(36).reshape(6,6)
a3 = arr3*5
a3
```

Out[14]:
```
array([[  0,   5,  10,  15,  20,  25],
       [ 30,  35,  40,  45,  50,  55],
       [ 60,  65,  70,  75,  80,  85],
       [ 90,  95, 100, 105, 110, 115],
       [120, 125, 130, 135, 140, 145],
       [150, 155, 160, 165, 170, 175]])
```

**Example3: (3 Mark)**

- 1.Create 2 data frames one with a shape of 2*3 and other with a shape of 3*4
- 2.Add these two data frames using the normal addition operator.
- 3.Now replace those NaN values with 0.

In [19]:
```python
sh1 = np.arange(23)
sh2 = np.arange(34)
df1 = pd.DataFrame(sh1)
df2 = pd.DataFrame(sh2)
newdf = df1+df2
finaldf = newdf.replace(np.nan,0)
finaldf
```

Out[19]:

|    | 0    |
|----|------|
| 0  | 0.0  |
| 1  | 2.0  |
| 2  | 4.0  |
| 3  | 6.0  |
| 4  | 8.0  |
| 5  | 10.0 |
| 6  | 12.0 |
| 7  | 14.0 |
| 8  | 16.0 |
| 9  | 18.0 |
| 10 | 20.0 |
| 11 | 22.0 |
| 12 | 24.0 |
| 13 | 26.0 |
| 14 | 28.0 |
| 15 | 30.0 |
| 16 | 32.0 |
| 17 | 34.0 |
| 18 | 36.0 |
| 19 | 38.0 |
| 20 | 40.0 |
| 21 | 42.0 |
| 22 | 44.0 |
| 23 | 0.0  |
| 24 | 0.0  |
| 25 | 0.0  |
| 26 | 0.0  |
| 27 | 0.0  |
| 28 | 0.0  |
| 29 | 0.0  |

|     | **0** |
| --- | --- |
| **30** | 0.0 |
| **31** | 0.0 |
| **32** | 0.0 |
| **33** | 0.0 |

**Example4:(3 Mark)**

Create 3 subplots. Have a numpy array.

- 1.The first subplot should have a plot of array vs array simple blue line.
- 2.Second array vs array ^2 red.
- 3.Third array vs array ^3 green.

In [38]:
```python
import matplotlib.pyplot as plt
%matplotlib inline
x= np.arange(0,10)
y=np.arange(5,15)
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.plot(x,y,'b')
plt.xlabel('X Axis Title Here')
plt.ylabel('Y Axis Title Here')

plt.show()
```
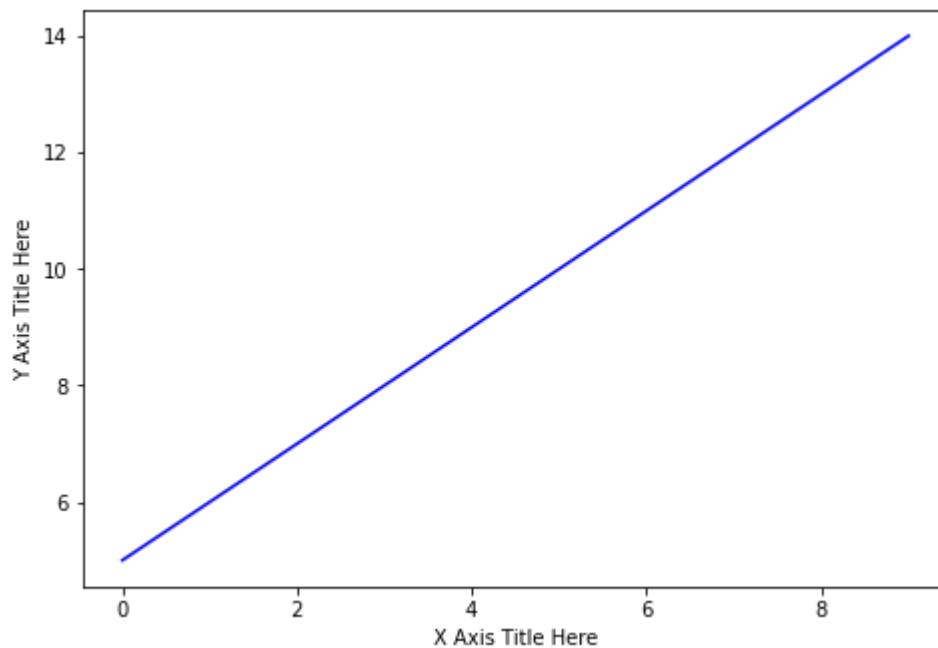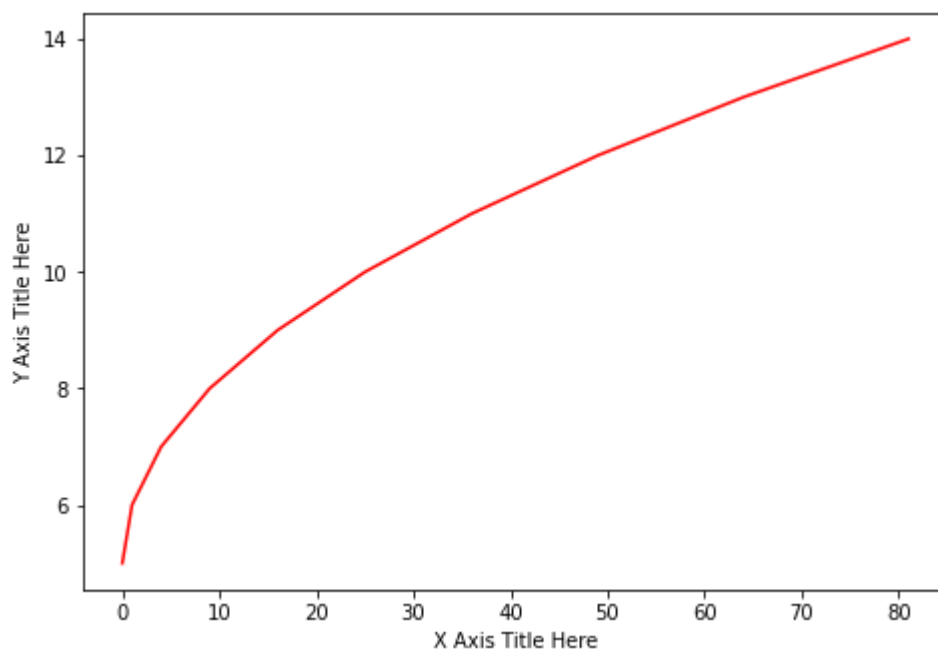
In [34]:

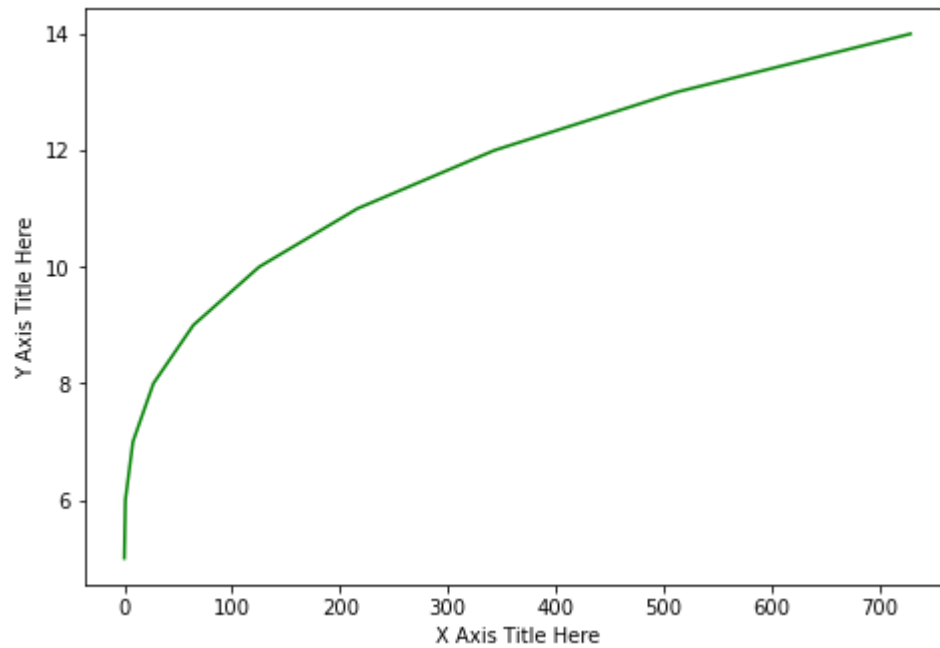Out[34]: array([ 0,  1,  4,  9, 16, 25, 36, 49, 64, 81], dtype=int32)

In [40]:
```python
z= x**2
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.plot(z,y,'r')
plt.xlabel('X Axis Title Here')
plt.ylabel('Y Axis Title Here')
plt.show()
```

In [41]:
```python
z= x**3
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.plot(z,y,'g')
plt.xlabel('X Axis Title Here')
plt.ylabel('Y Axis Title Here')

plt.show()
```



**Example5: (5 Mark) 1. Read the dataset and create dataframe**

1. Read the dataset and create dataframe
2. Create a new column called 'Total' that contains the Sum of 4 columns (Mini_Exam1, Mini_Exam2, Mini_Exam3 and Final)
3. use the scatter plot and find correlation between all numerical features.
4. write your analysis on the relationship with different features?
5. Use pie plot to display information of Grade column.
6. use the heatmap for numerical data.

In [61]:
```python
grade1 = pd.read_csv('Downloads/Grade (1).csv')
grade1.head()
```

Out[61]:

|   | Name | Mini_Exam1 | Mini_Exam2 | Participation | Mini_Exam3 | Final | Grade |
|---|------|-----------|-----------|---------------|-----------|-------|-------|
| 0 | Jake | 19.5 | 20.0 | 1 | 10.0 | 33.0 | A |
| 1 | Joe | 20.0 | 16.0 | 1 | 14.0 | 32.0 | A |
| 2 | Susan | 19.0 | 19.0 | 1 | 10.5 | 33.0 | A- |
| 3 | Sol | 22.0 | 13.0 | 1 | 13.0 | 34.0 | A |
| 4 | Chris | 19.0 | 17.0 | 1 | 12.5 | 33.5 | A |

In [66]:
```python
column_names = ['Mini_Exam1','Mini_Exam2','Mini_Exam3','Final']
grade1['Total'] = grade1[column_names].sum(axis=1)
grade1.head()
```
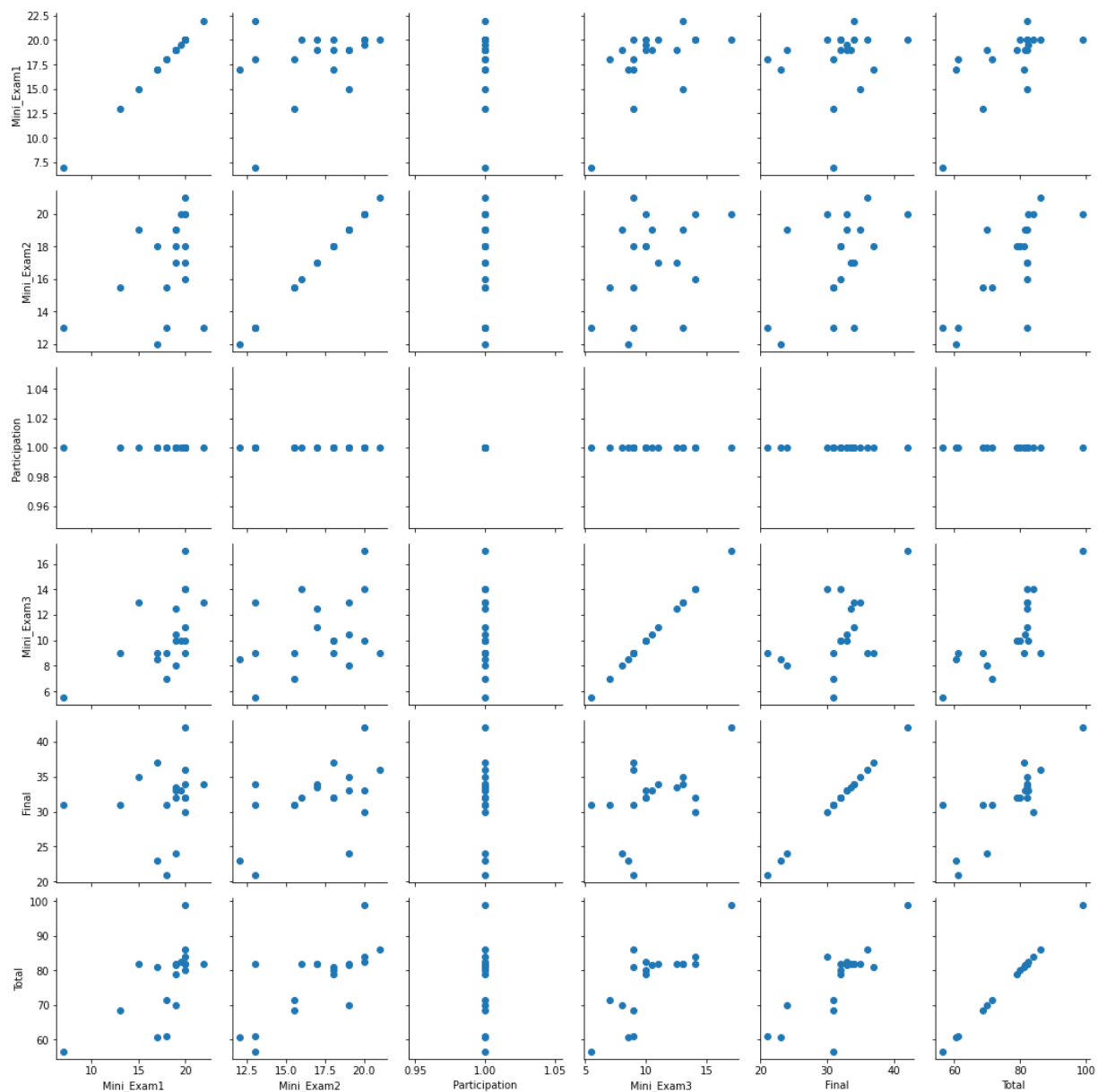
Out[66]:

|   | Name | Mini_Exam1 | Mini_Exam2 | Participation | Mini_Exam3 | Final | Grade | Total |
|---|------|-----------|-----------|---------------|-----------|-------|-------|-------|
| 0 | Jake | 19.5 | 20.0 | 1 | 10.0 | 33.0 | A | 82.5 |
| 1 | Joe | 20.0 | 16.0 | 1 | 14.0 | 32.0 | A | 82.0 |
| 2 | Susan | 19.0 | 19.0 | 1 | 10.5 | 33.0 | A- | 81.5 |
| 3 | Sol | 22.0 | 13.0 | 1 | 13.0 | 34.0 | A | 82.0 |
| 4 | Chris | 19.0 | 17.0 | 1 | 12.5 | 33.5 | A | 82.0 |

In [64]:
```python
import seaborn as sns
g = sns.PairGrid(grade1)
g.map(plt.scatter)
```

Out[64]: <seaborn.axisgrid.PairGrid at 0x19e9e3a6820>

In [ ]: |

In this scatter plot it is clear that those who have higher marks in all exams have total marks more than those who scored less in any of the exams.

**Example6: (1Mark)**

A national random sample of 11 ACT scores from 2010 is listed below. Calculate the sample mean and standard deviation.

- (29, 26, 13, 23, 23, 25, 17, 22, 17, 19, 12)

In [ ]: |