Q1.

""" Matplotlib is a Python library used for creating static, animated, and interactive visualizations in 2D and 3D

Uses :

- Data Visualization
- Custom Plotting
- Scientific and Technical Plotting

Common types of plots :

- Line Plots
- Bar Plots
- Scatter Plots
- Histograms
- Pie Charts

"""

Q2.

""" A scatter plot is a type of data visualization that displays individual data points on a two-dimensional Cartesian plane """

In [2]:
```
import matplotlib.pyplot as plt
import numpy as np
```

In [2]:
```
np.random.seed(3)
x = 3+np.random.normal(0,2,50)
y = 3+np.random.normal(0,2,len(x))
```

In [3]:
```
x
```

Out[3]:
```
array([ 6.57725695,  3.8730197 ,  3.19299494, -0.72698541,  2.44522359,
        2.29048204,  2.83451704,  1.74599865,  2.91236366,  2.04556394,
        0.37227049,  4.76924476,  4.76263608,  6.41914613,  3.10006728,
        2.19064517,  1.9092801 , -0.09295463,  4.96473487,  0.79786474,
        0.62990695,  2.5887002 ,  5.97229671,  3.47343253,  0.95242972,
        1.5740136 ,  4.25048993,  2.67897327,  1.4623273 ,  2.53993856,
        4.49011253,  6.95222157,  0.51175334,  1.74716618,  1.39246781,
       -1.83816635,  1.15241596,  0.95224848,  5.24795592,  2.73617153,
       -0.24657089,  4.2933509 ,  2.28745848, -0.48628207,  1.80670072,
        1.82281124,  1.2522354 ,  3.05942763, -1.49651554,  2.46447627])
```

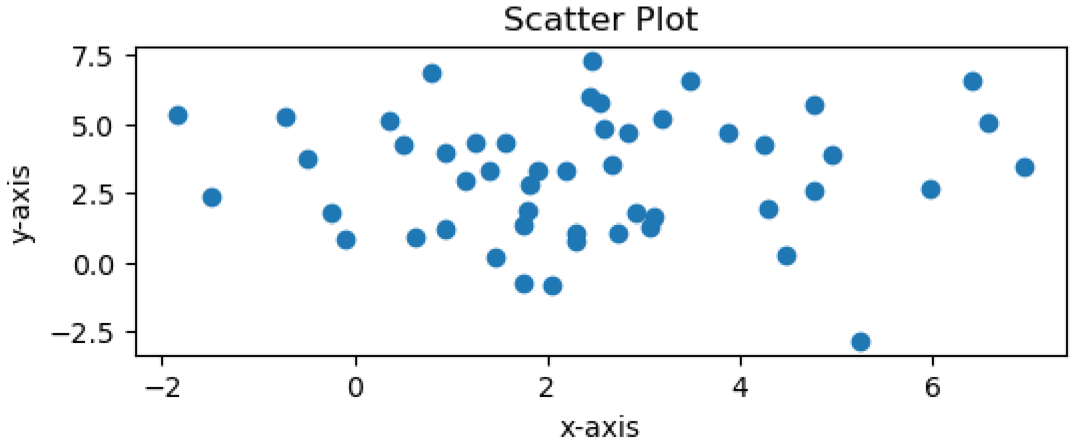
In [4]:
```
y
```

```
Out[4]: array([ 5.02636688,  4.70559568,  5.216375  ,  5.23878131,  5.97508626,
                0.76339863,  4.69166681, -0.72177906,  1.79422979, -0.82894409,
                5.09629502,  5.66747564,  2.60517064,  6.54929006,  1.65054498,
                3.30123373,  3.30589141,  0.87160945,  3.87589322,  6.87795692,
                0.95013825,  4.79867689,  2.6909863 ,  6.53925461,  3.9675767 ,
                4.3524328 ,  4.28632656,  3.49817341,  0.20847299,  5.78332582,
                0.25866197,  3.47712638,  4.22815418,  1.32417545,  3.29012643,
                5.33576457,  2.95179106,  1.22268516, -2.8314755 ,  1.05631899,
                1.81784252,  1.96716526,  1.08000764,  3.75459047,  1.85058316,
                2.78109133,  4.3581432 ,  1.28912566,  2.39958785,  7.31629868])
```

In [7]:

```python
plt.figure(figsize=(6,2))
plt.scatter(x,y)
plt.xlabel('x-axis')
plt.ylabel('y-axis')
plt.title('Scatter Plot')
```

```
Out[7]: Text(0.5, 1.0, 'Scatter Plot')
```



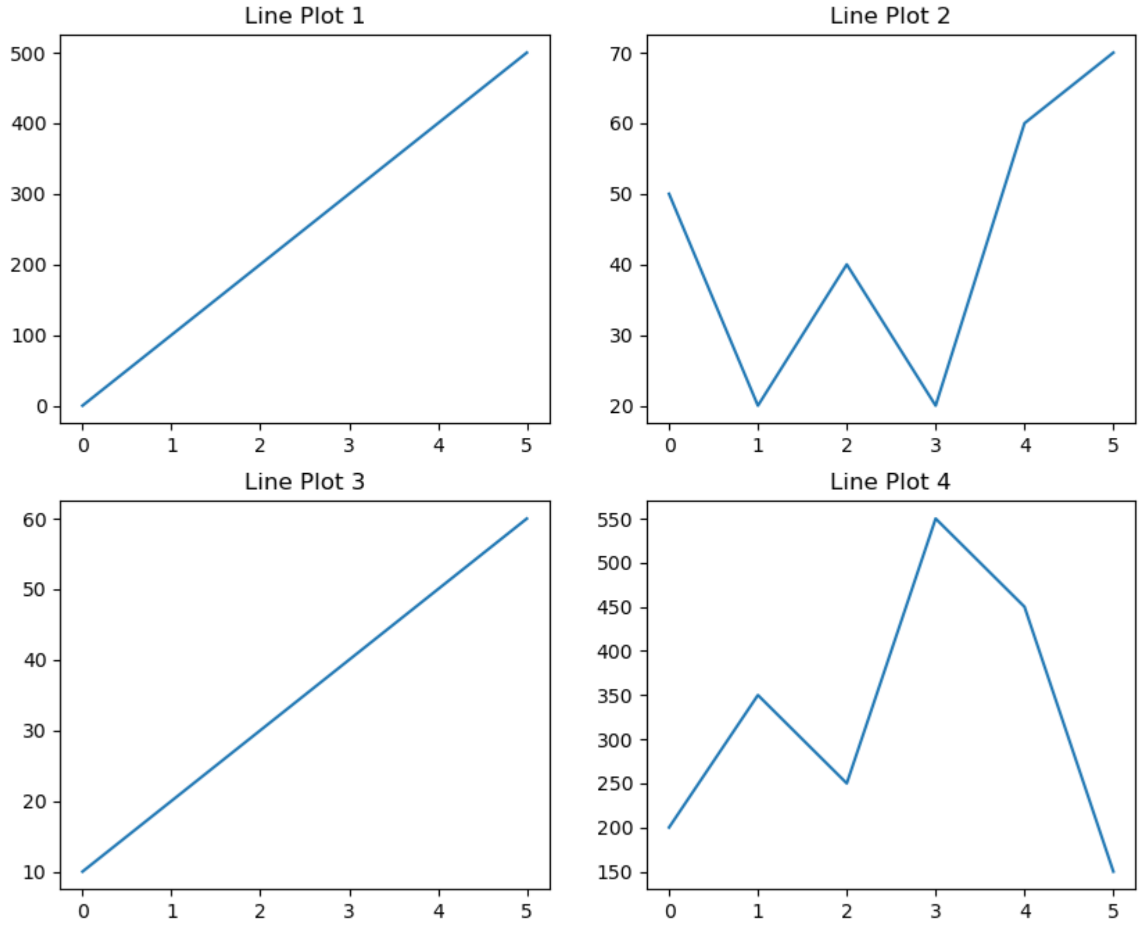
Q3.

""" The subplot() function in Matplotlib is used to create multiple plots or subplots within a single figure """

In [7]:

```python
x1 = np.array([0,1,2,3,4,5])
y1 = np.array([0,100,200,300,400,500])

x2 = np.array([0,1,2,3,4,5])
y2 = np.array([50,20,40,20,60,70])

x3 = np.array([0,1,2,3,4,5])
y3 = np.array([10,20,30,40,50,60])

x4 = np.array([0,1,2,3,4,5])
y4 = np.array([200,350,250,550,450,150])

plt.figure(figsize = (10,8))

plt.subplot(2,2,1)
plt.plot(x1,y1)
plt.title('Line Plot 1')

plt.subplot(2,2,2)
plt.plot(x2,y2)
plt.title('Line Plot 2')

plt.subplot(2,2,3)
plt.plot(x3,y3)
plt.title('Line Plot 3')

plt.subplot(2,2,4)
plt.plot(x4,y4)
plt.title('Line Plot 4')
```

```
Out[7]: Text(0.5, 1.0, 'Line Plot 4')
```



Line Plot 1

Line Plot 2

Line Plot 3

Line Plot 4

Q4.

""" A bar graph is a graphical representation of data using rectangular bars or columns to compare different categories or data points

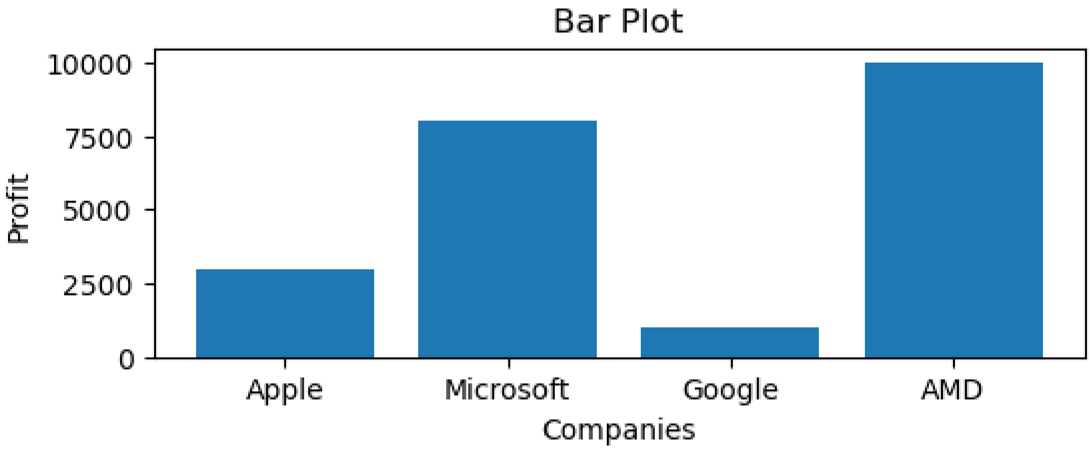
Uses :

- Comparisons
- Categorical Data
- Ranking
- Distribution

"""

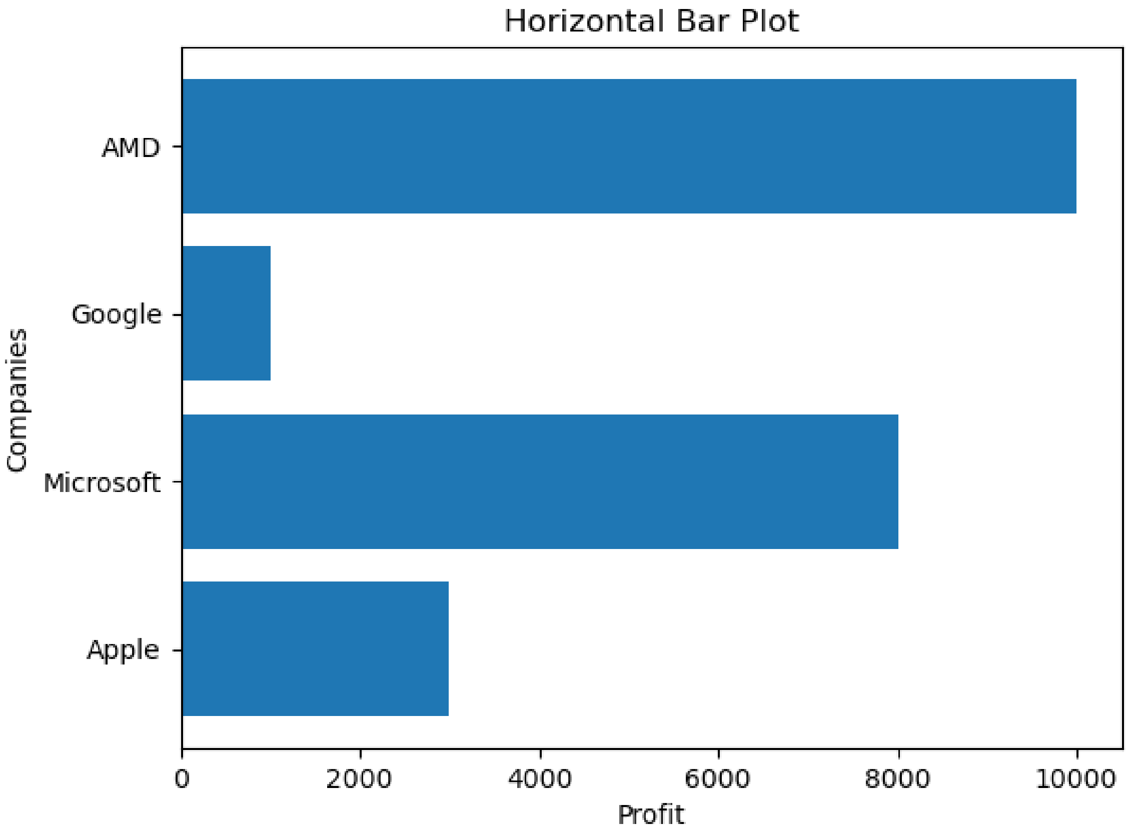
In [14]:

```python
company = np.array(["Apple","Microsoft","Google","AMD"])
profit = np.array([3000,8000,1000,10000])

plt.figure(figsize=(6,2))
plt.bar(company,profit)
plt.xlabel('Companies')
plt.ylabel('Profit')
plt.title('Bar Plot')
plt.show()
```



In [15]:

```python
plt.barh(company,profit)
plt.xlabel('Profit')
plt.ylabel('Companies')
plt.title('Horizontal Bar Plot')
plt.show()
```



Q5.

""" A box plot is a graphical representation of the distribution of a dataset. It provides a summary of key statistics and displays the spread and skewness of the data.

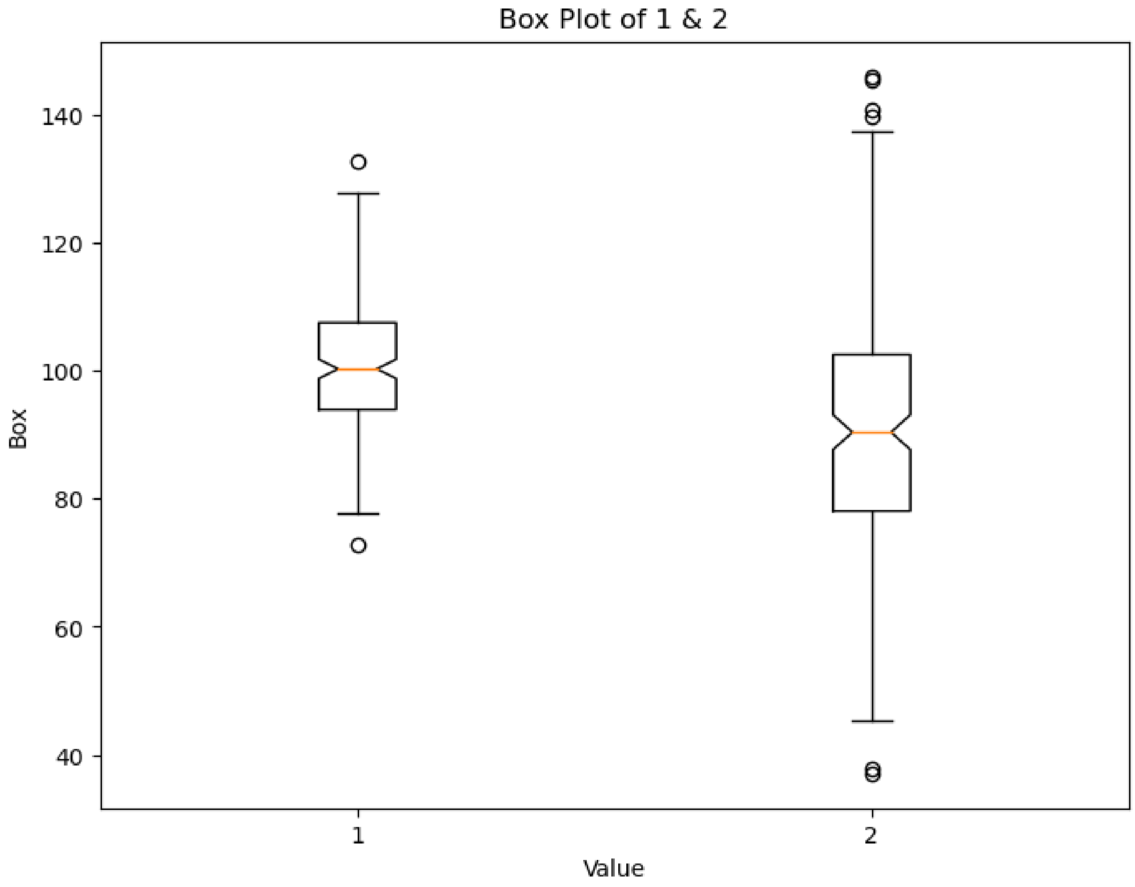
Uses :

- Visualizing Data Distribution
- Identifying Outliers
- Comparing Distributions
- Summarizing Data

"""

In [20]:

```
box1 = np.random.normal(100,10,200)
box2 = np.random.normal(90,20,200)

data = [box1,box2]
labels = ['Box1','Box2']

plt.figure(figsize=(8,6))
plt.boxplot(data,labels)
plt.xlabel('Value')
plt.ylabel('Box')
plt.title('Box Plot of 1 & 2')
plt.show()
```


Box Plot of 1 & 2
Box
Value