# MID TERM SOLVED PAPAER

Name : Kaushik Pawar, Roll Number : 23BEECE07, ECE

Q-1 create a 1D array of 9 elements using numpy module and reshape it into the 2D array of size 3*3.

In [2]:
```
import numpy as np
x= np.arange(9)
print(x)
y= x.reshape(3,3)
print(y)
```

```
[0 1 2 3 4 5 6 7 8]
[[0 1 2]
 [3 4 5]
 [6 7 8]]
```

Q-2 Print the output of the following.

In [3]:
```
list3=[x for x in range (10) if x%2==0]
print(list3)
```

```
[0, 2, 4, 6, 8]
```

Q-3 Given the string text = "PythonProgramming". Perform string slicing to extract specific substrings according to the following instructions: a) Slice the string to obtain the first 6 characters. b) Extract a substring that includes the characters from index 6 to index 13. c) Slice the last 5 characters from the string. d) Create a new string by slicing and concatenating the first 4 characters and the last 3 characters.

In [4]:
```
x="PythonProgramming"
y=x[:6]
print(y)
z=x[6:14]
print(z)
m=x[-5:]
print(m)
t=x[:4]+x[-3:]
print(t)
```

```
Python
Programm
mming
Pything
```

Q-5 Write a python program to generate two DataFrames, namely, di and d2. Construct di utilizing a two-dimensional list, and create d2 using a dictionary?

In [7]:
```
import pandas as pd
list1 = [[1,2,3],[4,5,6]]
dic1 = {'Name': ['chery','tej','ram'],
         'Age':[20,28,32],
       }
d1 = pd.DataFrame(list1)
d2 = pd.DataFrame(dic1)
print(d1)
print(d2)
```

```
   0  1  2
0  1  2  3
1  4  5  6
    Name  Age
0  chery   20
1    tej   28
2    ram   32
```

Q-6 How to measure strength of association between two variables? Write a python code to discuss in detail about the variance, standard deviation. covariance, and correlation?

In [8]:
```
import numpy as np
import pandas as pd
data={
     'x':[10,15,20,25,30],
     'y':[5,10,15,20,25]
}
df=pd.DataFrame(data)
variance_x=np.var(df['x'])
variance_y=np.var(df['y'])
dev_x=np.std(df['x'])
dev_y=np.std(df['y'])
covariance=np.cov(df['x'],df['y'])[0,1]
correlation=np.corrcoef(df['x'],df['y'])[0,1]
print(variance_x)
print(variance_y)
print(dev_x)
print(dev_y)
print(covariance)
print(correlation)
```

```
50.0
50.0
7.0710678118654755
7.0710678118654755
62.5
1.0
```

Q-7 Write a python program to explain the concepts of standardization and normalization? Discuss the circumstances under which it is appropriate to utilize these techniques in data preprocessing?

In [1]:
```
import numpy as np
from sklearn.preprocessing import StandardScaler, MinMaxScaler

# Sample data
data = np.array([[1, 2, 3],
                 [4, 5, 6],
                 [7, 8, 9]])

# Standardization
scaler_std = StandardScaler()
data_std = scaler_std.fit_transform(data)

print("Data after standardization:")
print(data_std)
print()

# Normalization
scaler_norm = MinMaxScaler()
data_norm = scaler_norm.fit_transform(data)

print("Data after normalization:")
print(data_norm)
```

```
Data after standardization:
[[-1.22474487 -1.22474487 -1.22474487]
 [ 0.          0.          0.        ]
 [ 1.22474487  1.22474487  1.22474487]]

Data after normalization:
[[0.  0.  0. ]
 [0.5 0.5 0.5]
 [1.  1.  1. ]]
```

Q-10 Provide a detailed explanation of the PCA technique for dimensionality reduction, including its methodology and application, supported by an illustrative example.

Principal Component Analysis (PCA) is a powerful technique used in data analysis, particularly for reducing the dimensionality of datasets while preserving crucial information. It does this by transforming the original variables into a set of new, uncorrelated variables called principal components. Here's a breakdown of PCA's key aspects:

Dimensionality Reduction: PCA helps manage high-dimensional datasets by extracting essential information and discarding less relevant features, simplifying analysis. Data Exploration and Visualization: It plays a significant role in data exploration and visualization, aiding in uncovering hidden patterns and insights. Linear Transformation: PCA performs a linear transformation of data, seeking directions of maximum variance. Feature Selection: Principal components are ranked by the variance they explain, allowing for effective feature selection. Data Compression: PCA can compress data while preserving most of the original information. Clustering and Classification: It finds applications in clustering and classification tasks by reducing noise and highlighting underlying structure. Advantages: PCA offers linearity, computational efficiency, and scalability for large datasets. Limitations: It assumes data normality and linearity and may lead to information loss. Let's say we have a data set of dimension 300 (n) × 50 (p). n represents the number of observations, and p represents the number of predictors. Since we have a large p = 50, there can be p(p-1)/2 scatter plots, i.e., more than 1000 plots possible to analyze the variable relationship. Wouldn't it be a tedious job to perform exploratory analysis on this data?

In this case, it would be a lucid approach to select a subset of p (p << 50) predictor which captures so much information, followed by plotting the observation in the resultant low-dimensional space.

The image below shows the transformation of high-dimensional data (3 dimension) to low-dimensional data (2 dimension) using PCA. Not to forget, each resultant dimension is a linear combination of p features

In [ ]:
```
Q-12 Explain concept of hypothesis testing in statistical analysis? Define the p-value in the context of hypothesis testing and explain its significance?
```

Hypothesis testing is a form of statistical inference that uses data from a sample to draw conclusions about a population parameter or a population probability distribution. First, a tentative assumption is made about the parameter or distribution. The P value is defined as the probability under the assumption of no effect or no difference (null hypothesis), of obtaining a result equal to or more extreme than what was actually observed. The P stands for probability and measures how likely it is that any observed difference between groups is due to chance. P Value and Statistical Significance: An Uncommon Ground