

Q.2 B) Create two lists, one representing subject names and the other representing marks obtained in those subjects. Display the data in a pie chart.

```
import matplotlib.pyplot as plt
subjects = ['Math', 'Science', 'English', 'History', 'Art']
marks = [85, 90, 78, 88, 92]
plt.figure(figsize=(8, 8))
plt.pie(marks, labels=subjects, autopct='%1.1f%%', startangle=140, colors=['gold', 'lightcoral', 'lightskyblue', 'lightgreen', 'lightpink'])
plt.title('Marks Distribution by Subject', fontsize=16)
plt.axis('equal')
plt.show()
```

Q.2 C) Write a program in python to perform following task (Use winequality-red.csv) [5] Import Dataset and do the followings: a) Describing the dataset b) Shape of the dataset c) Display first 3 rows from dataset

```
import pandas as pd
df = pd.read_csv('winequality-red.csv')
print("Describing the Dataset:")
print(df.describe())
print("\nShape of the Dataset:")
print(df.shape)
print("\nFirst 3 Rows of the Dataset:")
print(df.head(3))
```

Slip no 10

Q.1) Write a script to accept two integers(Use html form having 2 textboxes). Write a PHP script to,

- Find mod of the two numbers.
- Find the power of first number raised to the second.

- c. Find the sum of first n numbers (considering first number as n)
- d. Find the factorial of second number. (Write separate function for each of the above operations.)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Integer Operations</title>
</head>
<body>
  <h1>Integer Operations Form</h1>
  <form action="process.php" method="post">
    <label for="num1">Enter First Integer:</label><br>
    <input type="text" id="num1" name="num1" required><br><br>
    <label for="num2">Enter Second Integer:</label><br>
    <input type="text" id="num2" name="num2" required><br><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

PHP FILE

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $num1 = (int)$_POST['num1'];
  $num2 = (int)$_POST['num2'];
  function findMod($a, $b) {
    return $a % $b;
  }
  function findPower($a, $b) {
```

```

        return pow($a, $b);
    }

    function sumOfFirstNNumbers($n) {
        return ($n * ($n + 1)) / 2;
    }

    function factorial($n) {
        if ($n < 0) return "Undefined";
        return ($n == 0) ? 1 : $n * factorial($n - 1);
    }

    $mod = findMod($num1, $num2);
    $power = findPower($num1, $num2);
    $sum = sumOfFirstNNumbers($num1);
    $factorial = factorial($num2);
    echo "<h1>Results</h1>";
    echo "<p>Mod of $num1 and $num2: <strong>$mod</strong></p>";
    echo "<p>$num1 raised to the power of $num2: <strong>$power</strong></p>";
    echo "<p>Sum of first $num1 numbers: <strong>$sum</strong></p>";
    echo "<p>Factorial of $num2: <strong>$factorial</strong></p>";
} else {
    echo "<p>No data received.</p>";
}
?>

```

Q.2 A) Write a python program to Display column-wise mean, and median for the SOCRHeightWeight dataset.

```

import pandas as pd
# Load the SOCRHeightWeight dataset
df = pd.read_csv('SOCRHeightWeight.csv') # Replace with the actual file path
# Calculate column-wise mean
mean_values = df.mean()
# Calculate column-wise median
median_values = df.median()
# Display the results

```

```
print("Column-wise Mean:")
print(mean_values)
print("\nColumn-wise Median:")
print(median_values)
```

Q.2 B) Write a python program to compute sum of Manhattan distance between all pairs of points.

```
import numpy as np
# Sample data points (you can modify these points or read from a file)
points = np.array([[1, 2], [3, 5], [4, 1], [8, 0]])
def manhattan_distance(point1, point2):
    return np.abs(point1[0] - point2[0]) + np.abs(point1[1] - point2[1])
def total_manhattan_distance(points):
    total_distance = 0
    num_points = len(points)

    for i in range(num_points):
        for j in range(i + 1, num_points): # Avoid duplicate pairs
            total_distance += manhattan_distance(points[i], points[j])

    return total_distance
# Compute the sum of Manhattan distances
result = total_manhattan_distance(points)
# Display the result
print(f"Sum of Manhattan distances between all pairs of points: {result}")
```

Slip no 11

Q.1) Create a button with different style (Secondary, Primary, Success, Error, Info, Warning, Danger) using BootStrap.

```
<!DOCTYPE html>
<html lang="en">
```