Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

(1) There are 2 chefs, namely chef 1 and chef 2 in the MasterChef competition. The judge is going to judge on the basis of 3 categories: presentation, taste and hygiene to prepare the dishes. The marking is scaling from 1 to 100. The rating for chef 1 challenge is the triplet a = (a[0], a[1], a[2]), and the rating for Chef 2 challenge is the triplet b = (b[0], b[1], b[2]), where 0 index is presentation, 1 index is taste and 2 index is hygiene.

The task is to find their comparison points by comparing a[0] with b[0], a[1] with b[1], and a[2] with b[2].

 If a[i] > b[i], then Chef 1 is awarded 1 point.

 If a[i] < b[i], then Chef 2 is awarded 1 point.

 If a[i] = b[i], then neither person receives a point.

Comparison points are the total points a person earned.

Given a and b, determine their respective comparison points.

Design the algorithm for the same and implement using the programming language of your choice. Make comparative analysis for various use cases & input size.

Sample Input 1

27 48 70

89 26 7

Sample Output 1

2 1

Explanation 1

Comparing the 0th elements, 27<89 so Chef 2 receives a point.

Comparing the 1st and 2nd elements, 48>26 and 70>7 so Chef 1 receives two points.

The return array is [2,1].

CODE :-

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

- Python code:-

```
from flask import Flask, request, render_template_string

app = Flask(__name__)

def compare_chefs(chef1, chef2):
    chef1_points, chef2_points = 0, 0

    for i in range(3):
        if chef1[i] > chef2[i]:
            chef1_points += 1
      elif chef1[i] < chef2[i]:
            chef2_points += 1

    return chef1_points, chef2_points

@app.route('/', methods=['GET', 'POST'])
def index():
    chef1_points, chef2_points = 0, 0
    if request.method == 'POST':
        try:
            chef1 = [int(request.form[f'chef1_{i}']) for i in
range(1, 4)]
            chef2 = [int(request.form[f'chef2_{i}']) for i in
range(1, 4)]
            chef1_points, chef2_points = compare_chefs(chef1,
chef2)
        except ValueError:
            return "Invalid input. Please enter valid numbers."

    # Read the HTML file and render it with points
    with open('p1.html', 'r') as file:
        html_content = file.read()

    return render_template_string(html_content,
chef1_points=chef1_points, chef2_points=chef2_points)

if __name__ == "__main__":
```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

```
    app.run(debug=True)
```

- HTML code:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Chef Comparison</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
            margin-top: 50px;
        }
        .result {
            margin-top: 20px;
        }
    </style>
</head>
<body>
    <h1>Chef Comparison</h1>
    <form action="/" method="post">
        <h2>Enter scores for Chef 1 and Chef 2 </h2>
        <div>
            <h3>Chef 1 Scores:</h3>
            <input type="number" name="chef1_1" required>
            <input type="number" name="chef1_2" required>
            <input type="number" name="chef1_3" required>
        </div>
        <div>
            <h3>Chef 2 Scores:</h3>
            <input type="number" name="chef2_1" required>
            <input type="number" name="chef2_2" required>
```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

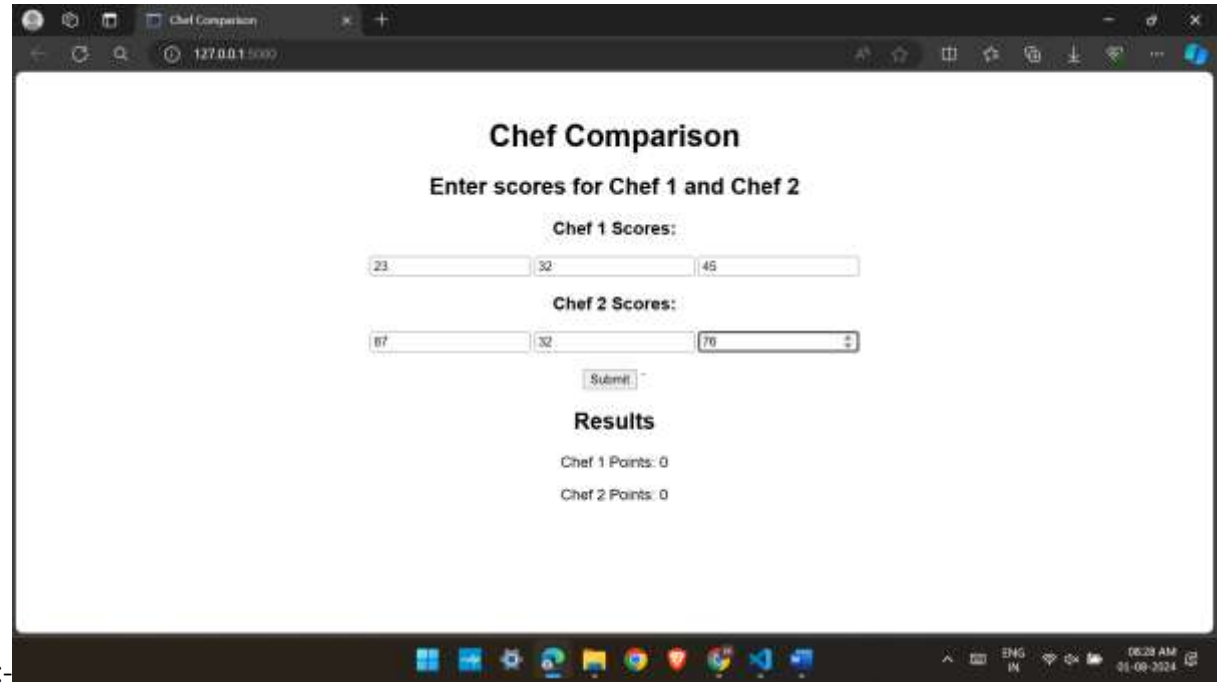
Practical-1

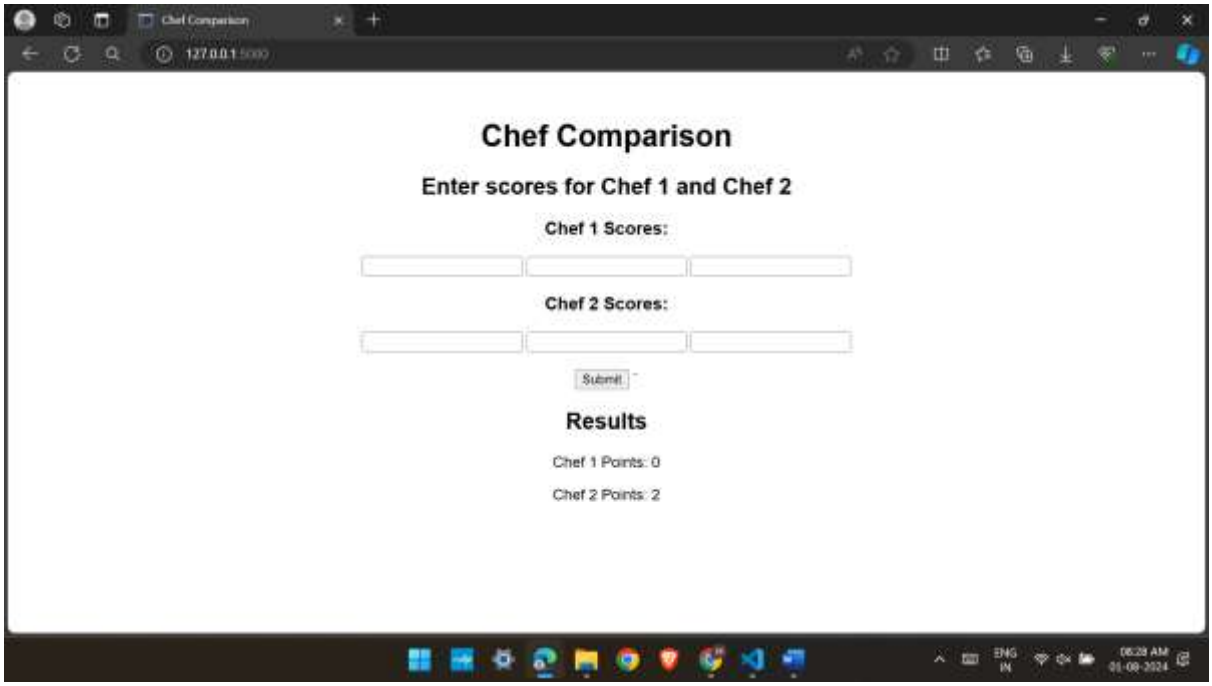
```
            <input type="number" name="chef2_3" required>
        </div>
        <br>
        <input type="submit" value="Submit">
  `   </form>

    {% if chef1_points is not none and chef2_points is not none %}
    <div class="result">
        <h2>Results</h2>
        <p>Chef 1 Points: {{ chef1_points }}</p>
        <p>Chef 2 Points: {{ chef2_points }}</p>
    </div>
    {% endif %}
</body>
</html>

```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

⇨ Output:-





⇨ TASK – 2 :-

(2) Let us suppose that you are having an array containing both positive and negative numbers. Given the numbers you are supposed to find 2 such elements such that the sum of those numbers is closest to zero.
Sample Input 1
15, 5, -20, 30, -45

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

Sample Output 1
15, -20
Explanation 1
In all the comparison, the sum of 15 and -20 is smallest amount among all other comparison.

Sample Input 2
15, 5, -20, 30, 25
Sample Output 2
15, -20 & -20, 25
Explanation 2
In all the comparison, the sum of 15,-20 & -20, 25

- Code:-

  - Python code:-

```
- from flask import Flask, request, send_from_directory,
  render_template_string
-
- app = Flask(__name__)
-
- def find_closest_to_zero_pair(arr):
-     closest_sum = float('inf')
-     closest_pairs = []
-
-     # Generate all pairs manually
-     for i in range(len(arr)):
-         for j in range(i + 1, len(arr)):
-             pair_sum = arr[i] + arr[j]
-             if abs(pair_sum) < abs(closest_sum):
-                 closest_sum = pair_sum
-                 closest_pairs = [(arr[i], arr[j])]
-             elif abs(pair_sum) == abs(closest_sum):
-                 closest_pairs.append((arr[i], arr[j]))
-
-     return closest_pairs
-
- @app.route('/', methods=['GET', 'POST'])
```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

```
- def index():
-     closest_pairs = []
-     if request.method == 'POST':
-         input_str = request.form['numbers']
-         try:
-             numbers = list(map(int, input_str.split(',')))
-             closest_pairs = find_closest_to_zero_pair(numbers)
-         except ValueError:
-             closest_pairs = [('Error', 'Invalid input')]
-
-     # Render HTML directly from the file
-     return render_template_string(open('p2.html').read(),
  closest_pairs=closest_pairs)
-
- if __name__ == "__main__":
-     app.run(debug=True)
```

- Html code:-

```
- <!DOCTYPE html>
- <html lang="en">
- <head>
-     <meta charset="UTF-8">
-     <meta name="viewport" content="width=device-width, initial-
  scale=1.0">
-     <title>Closest Pair to Zero</title>
-     <style>
-         body {
-             font-family: Arial, sans-serif;
-             text-align: center;
-             margin-top: 50px;
-         }
-         .form-container {
-             margin-top: 20px;
-         }
-         .result {
```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

```
-               margin-top: 20px;
-           }
-       </style>
-   </head>
-   <body>
-       <h1>Find Closest Pair to Zero</h1>
-       <form action="/" method="post">
-           <label for="numbers">Enter numbers separated by
    commas:</label><br>
-           <input type="text" id="numbers" name="numbers"
    required><br><br>
-           <input type="submit" value="Submit">
-       </form>
-
-       {% if closest_pairs %}
-       <div class="result">
-           <h2>Results</h2>
-           <p>Closest Pairs to Zero Sum:</p>
-           <p>
-               {% if closest_pairs[0][0] == 'Error' %}
-                   {{ closest_pairs[0][1] }}
-               {% else %}
-                   {% for p1, p2 in closest_pairs %}
-                       {{ p1 }}, {{ p2 }}{% if not loop.last %} &
    {% endif %}
-                   {% endfor %}
-               {% endif %}
-           </p>
-       </div>
-       {% endif %}
-   </body>
-   </html>
-
-
-
```

Name:- Marvin Patel
ER no :- 22162101013
Batch:-51
SUB :- AAD

Practical-1

⇨ Output:-