

EXPERIMENTS 1 : HTML & CSS

a) Create a web page to embed a map along with hot spot, frames & links.

AIM :

Embed an interactive map with clickable hotspots, external frames, and navigation links for a dynamic user experience.

ALGORITHM :

□ Start :

- Open an HTML document and structure it with the basic tags: `<html>`, `<head>`, and `<body>`.

□ Map Embedding :

- Use an `iframe` to embed a Google Map.
- Get the embed URL of the desired map location (e.g., Google Maps URL) and insert it in the `src` attribute of the `iframe`.

□ Hotspot Creation:

- Use the `<map>` tag to define areas within the map that can be clicked (hotspots).
- Each hotspot should be defined using the `<area>` tag and should link to a relevant page, e.g., a new map location.

□ Frames for External Content:

- Use `<iframe>` to embed external content (e.g., Wikipedia or any other site).

□ Adding Links :

- Create hyperlinks (`<a>`) that allow users to navigate to external websites or internal sections of the page.

□ Styling :

- Use internal CSS (within the `<style>` tag) to style the map container, hotspots, and frames to ensure they look clean and professional.

□ End :

- Close the HTML tags properly, ensuring a well-structured, responsive, and interactive webpage.

CODE:

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

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Map with Hotspots and Frames</title>

  <link
href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap"
rel="stylesheet">

  <style>
    body {
      font-family: 'Poppins', sans-
      serif; background-color:
      #f4f7fc; margin: 0; padding: 0;
      color: #333;

    }

    header {
      background-color: #0077b6;
      color: white;
      padding: 20px 0;
      text-align: center;
    }

    #map-container {
      max-width: 800px;
      margin: 20px auto;
      border-radius: 8px;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
```

```
        overflow: hidden;
    }
```

```
iframe {
    width: 100%;
    height: 400px;
    border: none;
}
```

```
.hotspot-info {
    background-color: #fff;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    margin: 20px auto;
    max-width: 800px;
}
```

```
.hotspot-info a {
    color: #0077b6;
    text-decoration: none;
    font-weight: bold;
}
```

```
footer {
    background-color: #0077b6;
    color: white;
    padding: 15px 0;
    text-align: center;
    margin-top: 40px;
```

```
}

footer a {
    color: #fff;
    text-decoration: none;
    font-weight: bold;
}
</style>
</head>
<body>

<header>
    <h1>Explore the World with Our Map</h1>
</header>

<!-- Embedded Map -->
<div id="map-container">
    <iframe
src="https://www.google.com/maps/embed/v1/place?q=New+York&key=YOUR_API_KEY
" allowfullscreen></iframe>
</div>

<!-- Hotspot Information -->
<div class="hotspot-info">
    <h2>Discover New York</h2>
    <p>Click the hotspot on the map to view New York on Google Maps:</p>
    <a href="https://www.google.com/maps/place/New+York" target="_blank">View New
York on Google Maps</a>
</div>

<!-- External Frame -->
```

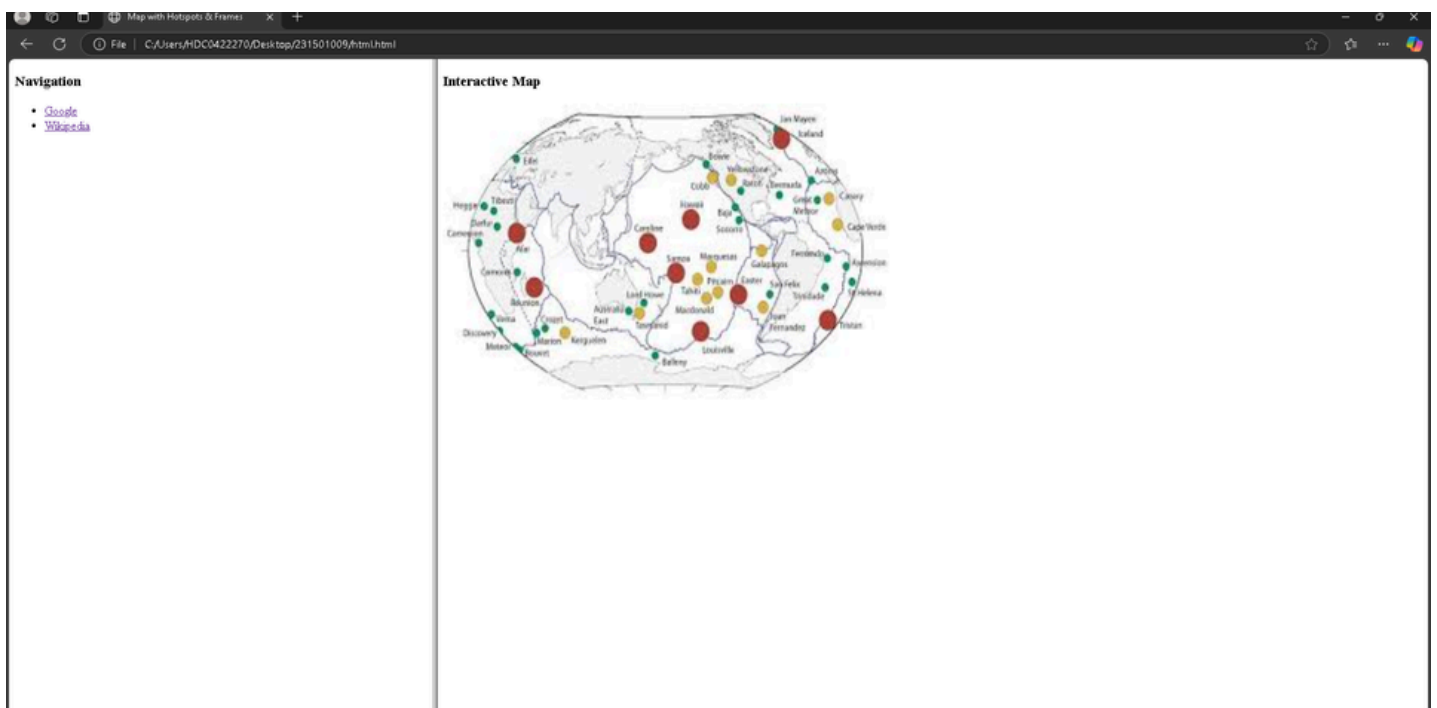
```

<div class="hotspot-info">
  <h2>Visit Wikipedia</h2>
  <p>Explore Wikipedia right here:</p>
  <iframe src="https://www.wikipedia.org/" width="100%" height="400"
allowfullscreen></iframe>
</div>

<!-- Footer -->
<footer>
  <p>&copy; 2025 Map Exploration | <a href="https://www.example.com">Visit our
website</a></p>
</footer>

</body>
</html>
OUTPUT :

```



RESULT :

Thus the give task is executed successfully.

b) Create a web page using an embedded, external and inline CSS file.

AIM:

Demonstrate the use of inline, embedded, and external CSS to style a webpage with a clean and professional design.

ALGORITHM:

□ Start :

- Open an HTML document and structure it with the basic tags: <html>, <head>, and <body> .

□ Inline CSS:

- In the body of the document, apply inline CSS using the style attribute within HTML tags (for example, styling a paragraph directly).

□ Embedded CSS:

- Inside the <head> section, use a <style> tag to write embedded CSS. This will apply styling rules to elements on the webpage, such as font, colors, and layouts.

□ External CSS :

- Create an external CSS file (e.g., styles.css) with all the styles for the webpage.
- Link the external CSS file to the HTML document using the <link> tag in the <head> section.

□ Add Content :

- Add content to the webpage, such as headings, paragraphs, buttons, and links, and apply the corresponding CSS classes to style them according to the method used (inline, embedded, or external).

□ Styling Elements:

- Use the external CSS file to define the layout and general styles (e.g., button styles, header styles).
- Use embedded CSS for specific styles within the webpage.
- Use inline CSS for unique, quick styling on individual elements.

□ End :

- Close the HTML tags properly, ensuring that the external, embedded, and inline styles are applied effectively, providing a well-styled webpage.

CODE :

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

  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Web Page with Inline, Embedded, and External CSS</title>

  <!-- External CSS Link -->
  <link rel="stylesheet" href="styles.css">

  <style>
    /* Embedded CSS */
    body {
      background-color: #f0f8ff;
      font-family: 'Arial', sans-serif;
    }
    h1 {
      color: #333;
      text-align: center;
      margin-top: 20px;
    }
    .embedded-style {
      background-color: #e2e8f0;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
      margin-top: 30px;
      text-align: center;
    }
  </style>
</head>
<body>
  <h1>Web Page with Inline, Embedded, and External CSS</h1>
  <div class="embedded-style">
    <p>This is a web page demonstrating the use of inline, embedded, and external CSS.
    The page features a light blue background, a white font family, and a
    centered heading. The embedded CSS is used to style the heading and
    the content area, while the external CSS is used to style the
    content area.
  </div>
</body>
</html>
```

```
.external-btn {
    background-color: #4CAF50;
    color: white;
    padding: 15px 32px;
    text-align: center;
    display: inline-block;
    font-size: 16px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
}

.external-btn:hover {
    background-color: #45a049;
}

</style>
</head>
<body>

    <h1>Welcome to My Stylish Web Page</h1>

    <!-- Inline CSS -->
    <p style="color: #ff6347; font-size: 18px; font-weight: bold; text-align: center;">
        This is an example of inline CSS styling.
    </p>

    <!-- Embedded CSS -->
    <div class="embedded-style">
        <p>This section is styled with embedded CSS. Notice the subtle background color and
        shadow effect.</p>
    </div>
```



```
<!-- External CSS -->
```

```
<button class="external-btn">Click Me</button>
```

```
</body>
```

```
</html>
```

External CSS File :

```
/* External CSS - styles.css */
```

```
body {
```

```
    font-family: 'Arial', sans-serif;
```

```
}
```

```
.external-btn {
```

```
    background-color: #0077b6;
```

```
    color: white;
```

```
    padding: 12px 24px;
```

```
    border-radius: 8px;
```

```
    border: none;
```

```
    cursor: pointer;
```

```
    font-size: 18px;
```

```
    text-align: center;
```

```
    transition: background-color 0.3s ease;
```

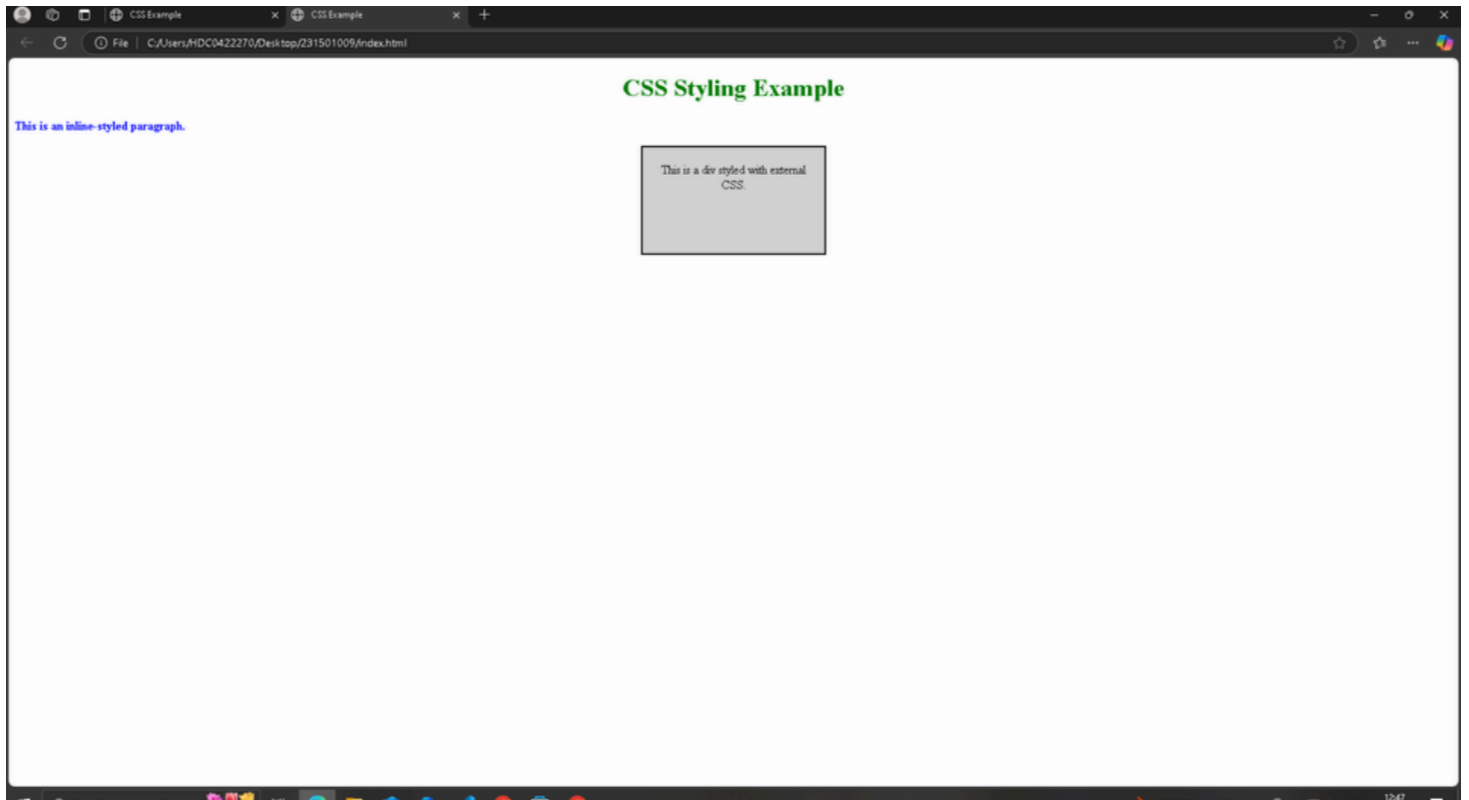
```
}
```

```
.external-btn:hover {
```

```
    background-color: #005f87;
```

```
}
```

OUTPUT:



RESULT:

Thus the given task is executed successfully.