

EXP: 1 HTML & CSS

A) Create a Web Page to Embed a Map with Hotspots, Frames & Links

AIM:

Create a Web Page to Embed a Map with Hotspots, Frames & Links.

ALGORITHM:

Step 1: Create the Main HTML File (index.html)

- Start with the basic HTML structure using `<html>`, `<head>`, and `<body>`.
- Add a `<title>` tag for the page title.
- Link an external CSS file for styling.

Step 2: Embed an OpenStreetMap

- Use the `<iframe>` tag to display a map.
- Set attributes such as width, height, and frameborder to customize the view.

Step 3: Create an Image Map for Hotspots

- Use the `` tag to add a map image.
- Define a `<map>` element with a name.
- Add `<area>` elements for interactive hotspots using different shapes (rect, circle, etc.).
- Assign href attributes to direct users to specific locations when they click.

Step 4: Implement Navigation Links

- Add `<a>` tags to navigate between sections.
- Create anchor links for quick page scrolling or external links to other pages.
- Use **hover effects** on links to enhance interactivity and provide visual feedback to users.

Step 5: Add an Inline Frame (iframe) for Extra Content

- Use <iframe> to display linked content without leaving the page.
- Set a name attribute to target it from navigation links.

Step 6: Create Additional Linked Pages (page1.html, page2.html)

- Build simple HTML pages with text, images, or additional information.
- Apply CSS for styling.
- Test the entire setup to ensure links and hotspots function correctly.

PROGRAM:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Embedded Map with Hotspots</title>
  <link rel="stylesheet" href="style.css"> <!-- External CSS -->
</head>
<body>
  <h1>Interactive Map with Hotspots</h1>

  <div class="frame">
    <iframe
src="https://www.openstreetmap.org/export/embed.html?bbox=77.5946%2C12.9716%2C78.5946%
2C13.9716&layer=mapnik"
    width="600" height="400" frameborder="0"></iframe>
  </div>

  
  <map name="hotspots">
    <area shape="rect" coords="50,50,100,100" href="https://example.com/place1" alt="Place 1">
    <area shape="circle" coords="150,150,40" href="https://example.com/place2" alt="Place 2">
  </map>
</body>
</html>
```

OUTPUT:

Interactive Map with Hotspots



RESULT: The image map with hotspots, frames, and links is successfully created.

B) Create a Web Page Using Embedded, External & Inline CSS

AIM:

Create a Web Page Using Embedded, External & Inline CSS

ALGORITHM:

Step 1: Set Up the HTML File (index.html)

- Create the base structure using `<html>`, `<head>`, and `<body>`.
- Add a `<title>` tag for the webpage.

Step 2: Link an External CSS File (style.css)

- Use `<link rel="stylesheet" href="style.css">` inside `<head>`.
- Define global styles like background color, font styles, and layout in `style.css`.

Step 3: Add Embedded CSS for Specific Elements

- Inside the `<style>` tag in `<head>`, define styles for certain elements (e.g., `div` and `p`).
- Set properties like color, font size, border radius, or box shadow for visual enhancement.

Step 4: Use Inline CSS for Individual Elements

- Apply the style attribute directly inside elements (`<h1>`, `<p>`).
- Example: `<p style="color: blue; font-size: 18px;">This text uses inline CSS.</p>`.

Step 5: Organize Content with Different CSS Types

- Demonstrate each type of CSS by styling different elements.
- Ensure clear visibility of inline, embedded, and external styles.

Step 6: Test the Page for Correct Styling

- Open the file in a browser and verify that all three CSS methods are working.
- Adjust the styles as needed for readability and responsiveness.

PROGRAM:

INDEX.HTML:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>CSS Styling Methods</title>

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

    <style>

        /* Embedded CSS */

        .embedded-css-example {

            color: green;

            font-style: italic;

        }

    </style>

</head>

<body>

    <h1>CSS Styling Examples</h1>

    <p style="color: blue; font-weight: bold;">This paragraph uses Inline CSS.</p>

    <div class="embedded-css-example">

        <p>This paragraph is styled using Embedded CSS.</p>

    </div>

    <div class="external-css-example">

        <p>This paragraph is styled using External CSS.</p>

    </div>

</body>

</html>
```

```
</div>

</body>

</html>
```

STYLE.CSS:

```
.external-css-example {
  color: red;
  font-size: 18px;
  text-decoration: underline;
}
```

OUTPUT:

CSS Styling Examples

This paragraph uses Inline CSS.

This paragraph is styled using Embedded CSS.

This paragraph is styled using External CSS.

RESULT: The webpage demonstrates external, embedded, and inline CSS styles effectively.