



MERN Stack Journey – Class 09

Developing Layouts: Favourite Places Section

This class focuses on **building real UI layouts** using HTML & CSS by creating a **Favourite Places section**. You will learn how images, margins, background containers, and flex layouts work together to form a complete webpage section.



HTML Elements

1.1 Image Element (`img`)

The HTML `img` element is used to display images on a webpage.

Syntax:

```

```

1.2 `src` Attribute

The `src` attribute specifies the **path (URL)** of the image.

```

```

1.3 How to Get Image URL from Internet

1. Open <https://images.google.com/>
 2. Search for the required image
 3. Click on the image
 4. Open image in new tab
 5. Copy the URL from browser
 6. Paste it inside the `src` attribute
-

1.4 Applying Height & Width to an Image

We use **CSS** to control image size.

```

```

```
.image {  
  width: 80px;  
  height: 100px;  
  border-radius: 8px;  
}
```

MCQs (With Answers)

Q1. Which HTML element defines an image? - p - h1 - ☒ img - div

Q2. Which attribute specifies the image path? - class - href - path - ☒ src

Void Elements

Void elements contain **only a start tag** and **no closing tag**.

Example:

```

```

Statement: The `img` element is a void element. - ☒ True - False

CSS Box Model Property – Margin

3.1 Margin

Margin creates **space outside an element**.

```
.card-container {  
  margin: 10px;  
}
```

You can also apply margin individually: - `margin-top` - `margin-right` - `margin-bottom` - `margin-left`

MCQs

Q1. Which property adds space between elements? - border-width - space - ☒ margin

Q2. Correct value for margin:

```
.card { margin: 20px; }
```

Q3. Margin for top side: - margin-right -  margin-top - margin-bottom - margin-left

Favourite Places Section – Step by Step

Step 1: Background Container & Card

- Background image
- Heading
- Card container

```
.favourite-places-bg-container {  
  background-image: url("towerbg.png");  
  height: 100vh;  
  background-size: cover;  
}
```

Step 2: Add Image to Card

```
.favourite-place-card-image {  
  width: 80px;  
  height: 100px;  
}
```

Step 3: Align Text & Image Side by Side

Using **Bootstrap Flexbox**:

```
<div class="favourite-place-card-container d-flex flex-row">
```

Step 4: Multiple Cards & Spacing

```
.favourite-place-card-container {  
  margin: 15px;  
}
```

This creates clean spacing between cards.



Key Learnings

- How images work in HTML
 - Void elements concept
 - Margin vs Padding
 - Building layouts step-by-step
 - Real-world UI section creation
-



Interview Questions

1. What is a void element in HTML?
 2. Difference between margin and padding?
 3. How do you align elements side by side?
 4. Why is `background-size: cover` used?
 5. How do you control image size in CSS?
-



This class marks the transition from **basic CSS** to **real layout development**. Next up: **Flexbox Deep Dive & Responsive Design** 🚀