index.html

```html
<!DOCTYPE html>
<html>
<frameset rows="50", cols="200">
    <frame src="top.html">
    <frameset cols="200">
        <frame src="left.html">
        <frame src="right.html">
    </frameset>
</frameset>
</html>
```

* top.html

```html
<h1>Bike Showroom</h1>
<ul>
    <li><a href="honda.html" target="right">Honda</a></li>
    <li><a href="yamaha.html" target="right">Yamaha.html"
    <li><a href="suzuki.html" target="right">Suzuki</a>
    <li></ul>
```

honda.html, yamaha.html, suzuki.html

(honda.html)
```html
<h2>Honda</h2>
<p>Model 1: CBR500R</p>
<p>Model 2: CBR1000R</p>
```

(yamaha.html)
```html
<h2>Yamaha</h2>
<p>Model 1: YZF-R6</p>
<p>Model 2: YZF-R1</p>
```

(suzuki.html)
```html
<h2>Suzuki</h2>
<p>Model 1: GSX-R606</p>
<p>Model 2: GSX-R1000</p>
```

2.
```html
<form>
    <h2>Job Application Form</h2>
    <label>First Name:</label>
    <input type="text" name="first-name"><br><br>
    <label>Last Name:</label>
    <input type="text" name="last-name"><br><br>
    <label>Degree:</label>
    <select name="degree">
        <option value="high school">High School</option>
        <option value="bachelors">Bachelor's</option>
        <option value="masters">Master's</option>
    </select><br><br>
    <label>Gender:</label>
    <input type="radio" name="gender" value="male">Male
    <input type="radio" name="gender" value="female">Female
    <input type="radio" name="gender" value="other">other<br><br>
    <button type="submit">Submit</button>
</form>
```

3. LAMP vs WAMP

LAMP (Linux, Apache, MySQL, PHP)
- Operating System: Linux
- Web Server: Apache
- Database: MySQL
- Programming Language: PHP

WAMP (Windows, Apache, MySQL, PHP)
- * Operating System: Windows
- * Web Server: Apache
- * Database: MySQL
- * Programming Language: PHP

Apache's Role:
- Handles HTTP requests
- Servers web pages
- Integrate with PHP and MySQL

4. HTTP Request
1. Method (GET, POST, PUT, DELETE)
2. URL
3. Headers (info about request)
4. Body (optional data)

HTTP Response.
1. Status Code (200 ok, 404 Not Found, etc.)
2. Headers (info about response)
3. Body (optional data)

Common Status Code
- 200 : Ok
- 404 : Not Found
- 500 : Server Error

HTML

index.html

```html
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="style.css">
</head>
<body>
    <h1>Home</h1>
    <a href="about.html">About</a>
    <a href="contact.html">Contact</a>
</html>
```

css (style.css):

```css
body {
    font-family: Arial;
    background-color: blue
}
h1 {
    color: red;
}
a {
    text-decoration: none;
    color: green;
}
```

6. **HTML 5 Features**

1. New input types
   (email, date, time)

2. Form validation
   (required, pattern)

3. Semantic Element
   (header, Footer, nav)

```html
<form>
    <label>Name:</label>
<input type="text" required>
    <label>Email:</label>
    <input type="email" required>
    <button>Submit</button>
</form>
```

HTML5 Benefits:

1. Improved accessibility
2. Better form validation
3. Enhanced user Experience.

7. HTML:

```html
<form>
<label> Name : </label
<input type="text" required>

<label>Email : </label>
<input type="email" required>

<label> Message : </label>
<textarea required></textarea>

<button> Submit </button>
</form>
```

8. Responsive Web Design

1. fixed Layout (fixed pixels)
2. Fluid layout (percentages)
3. Responsive Layout (adapts to Screen size)

CSS Techniques

1. Flexbox (flexible layouts)
2. Grid (grid-based layouts)
3. Media Queries (Styles for different screen size)

Example css

```css
• container {
   display : flex;
   flex -wrap : wrap;
 }
@media (max-width : 768px) {
   /* styles for smaller screens */
}
```

9. Key Principles:-

1. Flexible layouts
2. Scalable Images
3. Medical Queries

CSS Techniques:

1. Flex box
2. Grid
3. Bootstrap

Break Points:

1. Mobile (320px - 479px)
2. Tablet (480px - 767px)
3. Desktop (768px+)

Design Considerations:

1. Simpler Navigation
2. Clear Typography
3. Minimal Content

* Tools:

1. Google Chrome Dev Tools
2. Mozilla Responsive Design Mode
3. Mobile - Friendly Test Tool

---

10. Webpage Design

- Header
- Content (text, images)
- Footer

9. key Principles:-
1. Flexible layouts
2. Scalable Images
3. Medical Queries

CSS Techniques:
1. Flex box
2. Grid
3. Bootstrap

Break Points:
1. Mobile (320px - 479px)
2. Tablet (480px - 767px)
3. Desktop (768px+)

Design Considerations:
1. Simpler Navigation
2. Clear Typography
3. Minimal Content

* Tools:
1. Google Chrome Dev Tools
2. Mozilla Responsive Design Mode
3. Mobile - Friendly Test Tool

10. Webpage Design
- Header
- Content (text, images)
- Footer

HTML ELEMENTS

- h1-h6 (headings)
- p (paragraphs)
- img (images)
- a (links)
- ul/ol (lists)

HTML Response:

1. user requests webpage (GET)
2. Server sends HTML (200 OK)
3. Browser request resources (CSS, Java Script, images)
4. Server sends resources (200 OK)
5. Browser displays webpage