

## 1. Navigation Bar and Header Section

Create a simple navigation bar and header section using only plain HTML and CSS.

### Requirements:

- Use <a> tags to create clickable links for redirection.
  - Include a person's image using the <img> tag.
  - Add a brand name.
  - All items should be displayed inline (horizontally).
  - On hover, the background color of the links should change.
- 

## 2. HTML Paragraph Formatting

Write the following paragraph using plain HTML and apply the given formatting rules:

### Paragraph:

FITA is an Online and Offline transformative upskilling platform for working tech and business professionals. \*

### Formatting Instructions:

- Make "Online and Offline transformative" appear in **bold**.
  - Make "upskilling platform" appear in *italic*.
  - Apply **strikethrough** to the phrase "and business".
  - Display the \* symbol as a **superscript**.
  - Use the & symbol instead of the word "and" where appropriate.
  - Add the © (**copyright**) symbol at the end using an HTML entity.
-

### 3. Book Table with Features

Create an HTML table to display information about 5 different books.

#### Requirements:

- The table must include a header row and 4 data rows (5 books total).
  - Each book entry should include:
    - Title
    - Author
    - Publication Date
    - Number of Pages
  - Use the following HTML features:
    - colspan to merge header cells.
    - rowspan for any repeated author entries.
    - border-collapse to remove spacing between table borders.
- 

### 4. Survey Form

Create an HTML form to collect information about users' favorite food.

#### Fields:

- Name
- Age
- Gender (radio buttons: Male/Female)

- Favorite Food (select dropdown: Pizza, Chinese, Indian, etc.)
  - "Do you like to cook?" (checkbox)
  - Additional Comments (textarea)
  - Submit button
- 

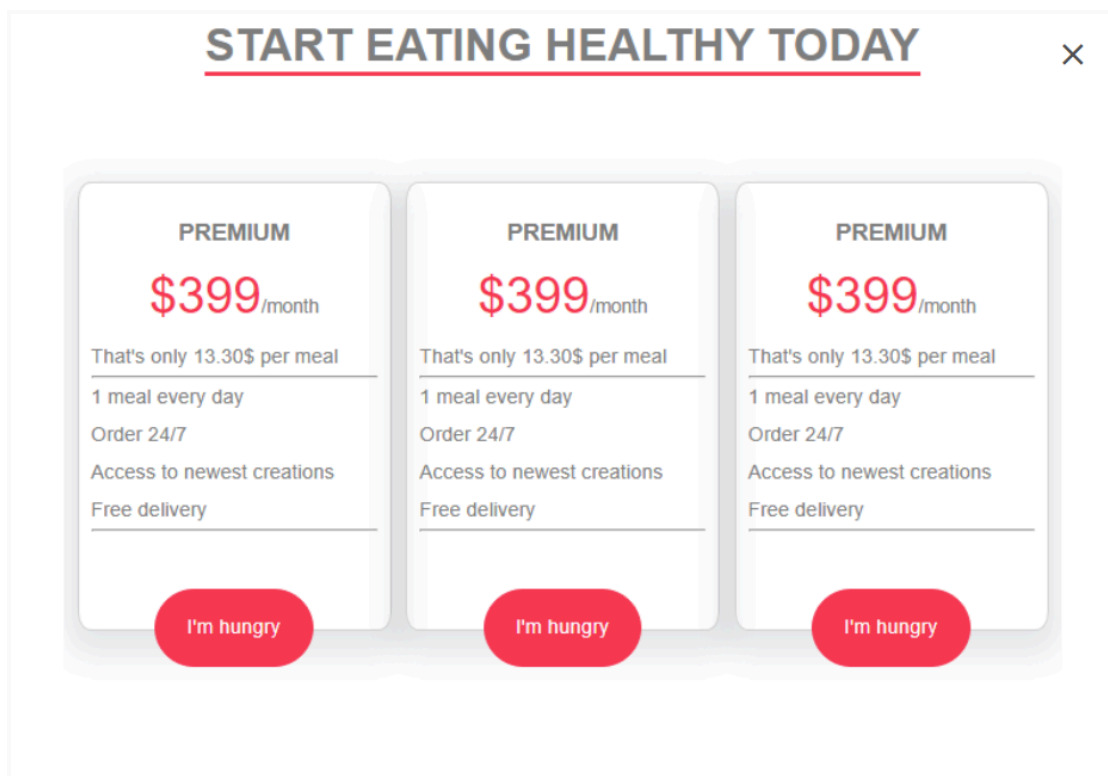
## 5. Ordered and Unordered Lists

Create a basic HTML webpage displaying two lists:

- Ordered List (5 cities around the world, use alphabetic markers)
  - Unordered List (3 beverages, use square markers)
- 

## 6. Flexbox Pricing Cards

Write HTML and CSS to create a responsive pricing card layout using **Flexbox**. Each card should be styled clearly with titles, prices, and buttons.



## 7. Simple Animation - Box Movement

Create a simple animation where a box moves from **left to right** in 5 seconds and repeats infinitely.

---

## 8. Transform on Hover

Create a box that **scales up to 1.2x** and **rotates 10 degrees** when hovered, using the transform property.

---

## 9. Animated Color-Changing Circle

Create a circular `<div>` that changes its background color infinitely in a smooth loop.

### Requirements:

- Size: 200px x 200px with fully rounded borders
  - Use `@keyframes` to animate background colors: red → yellow → blue → green
  - Use animation with an 8-second duration and infinite looping
- 

## 10. HTML Audio and Video Player

- Add an audio player that plays a .mp3 file with controls but **autoplay disabled**.
  - Create a `<video>` tag that plays a **local .mp4 file**, has controls, a poster image, and starts **muted and autoplayed**.
-

## 11. Responsive Card using Media Queries

Create a responsive `<div class="card">` that adjusts layout and style based on screen width.

### Breakpoints:

- **Mobile-first:**
  - Width: 100%
  - Text aligned center, font size: 14px
  - No shadow
- **≥ 600px:**
  - Width: 80%, font size: 16px
  - Box shadow + hover effect (translateY)
- **≥ 900px:**
  - Flex layout with an .image-section
  - Card: 70%, Image: 30%
  - Font size: 18px, border-radius: 10px

---

## 12. Login Form with Validation

Create a login form with:

- Email and Password fields using `type="email"`, `required`, and `minlength`
  - A "Remember Me" checkbox
  - A Submit button
-

### 13. Sticky Section Headings

Create multiple sections with headings that stick to the top of the viewport as you scroll using position: sticky.

---

### 14. Use nth-child() to style every 3rd item differently.

---

### 15. Local Storage and Session Storage

Create a page that:

- Stores user input (e.g., name or color preference) in **Local Storage** and **Session Storage**
  - Display the stored values in the **console** using console.log()
-