**Exercise 1: Static vs Relative Position**

Create a box (<div>) and use different positioning modes like static and relative to see how they behave.

**Task:**

1. Create a div element with a fixed width and height.
2. Apply position: relative; to the box and move it 50px down and 30px to the right using top and left.
3. Add another div with position: static;. Observe the difference in how these two divs are placed.

**Exercise 2: Absolute Positioning**

Use position: absolute to position an element relative to its nearest positioned ancestor.

**Task:**

1. Create a div container with two nested boxes inside it.
2. Set the container's position to relative.
3. Position one of the nested boxes using position: absolute; top: 10px; right: 10px;.
4. Observe how the absolute positioning behaves relative to the container.

**Exercise 3: Fixed Positioning**

Create a header bar that stays fixed at the top of the screen when scrolling.

**Task:**

1. Create a header element.
2. Set its position to fixed and make it stick to the top of the viewport.
3. Add a long content section that allows you to scroll and see the header stay fixed.

**Exercise 4: Sticky Positioning**

Use position: sticky to create a navigation bar that sticks to the top of the page when scrolling.

**Task:**

1. Create a div navigation bar at the top of the page.
2. Set its position to sticky and top: 0;.
3. Add content below the navigation bar and observe how the navigation sticks when you scroll past it.

**Exercise 5: Z-Index with Absolute Positioning**

Use position and z-index to overlap elements.

**Task:**

1. Create three div elements.
2. Position each one with absolute positioning.
3. Use z-index to control the stacking order of the elements.

**Exercise 6: Login Form with Absolute Positioning**

Create a login form that is centered using position: absolute.

**Task:**

1. Create a login form with fields for username and password.
2. Use position: absolute to center the form horizontally and vertically in the viewport.

**Exercise 7: Contact Form with Fixed Positioning**

Create a contact form that is always visible on the right side of the screen using position: fixed.

**Task:**

1. Create a contact form with fields for name, email, and message.
2. Use position: fixed to stick the form to the right side of the screen.

**Exercise 8: Registration Form with Sticky Positioning**

Create a registration form with a navigation bar that sticks to the top as you scroll down.

**Task:**

1. Create a form with fields for username, email, and password.
2. Add a navigation bar that sticks to the top of the viewport when the user scrolls down.

**Exercise 9: Feedback Form with Z-Index**

Create a feedback form that overlaps a background image using position: absolute and z-index.

**Task:**

1. Create a feedback form that appears over an image.
2. Use z-index to make the form appear on top of the image.

**Exercise 10: Sidebar Form with Absolute and Relative Positioning**

Create a sidebar with a form that slides out using position: absolute inside a relative container.

**Task:**

1. Create a sidebar that slides out from the left.
2. Use position: relative for the container and position: absolute for the sidebar.

**Exercise 11: Multi-Step Form with Sticky Progress Bar**

**Task:**

1. Create a multi-step form with fields spread across three steps (e.g., personal information, contact details, and account setup).
2. Use CSS checkboxes to move between the form steps.
3. Add a sticky progress bar at the top of the form to show the user's progress.

**Exercise 12: Complex Multi-Step Form with Sidebar Navigation**

**Task:**

1. Create a multi-step form with fields spread across four steps (personal info, address, contact details, and account setup).
2. Implement a fixed sidebar navigation that highlights the current step and provides step-based navigation.
3. Use CSS position properties for sticky elements (progress bar) and floating labels for form fields.

**Exercise 13: Product Catalog Multi-Step Form**

**Task:**

1. Build a large multi-step form to manage product information (details, pricing, images, inventory, shipping).
2. Include a sidebar navigation to switch between the steps.
3. Ensure the form is responsive and includes floating labels, a sticky sidebar, and advanced positioning techniques.

**Exercise 14: Department and Employee Management Form**

**Task:**

1. Create a multi-step form for adding department details and employee information.
2. Include a sidebar to navigate between the sections: Department Details, Add Employee, Edit Employee, and Assign Employee.
3. Use advanced CSS techniques like floating labels, sticky positioning for the sidebar, and grid layout for the employee fields.
4. Ensure the form is responsive.

**Exercise 15: Blog Post Management Form**

**Task:**

1. Create a multi-step form for managing blog posts.
2. Include sections for adding post details, assigning categories, adding tags, and previewing the post.
3. Implement a sidebar to navigate between sections.
4. Use grid layout for form fields and ensure responsive design.