CSS Position Property

~~~markdown
### **Position Property in CSS**

 It determines how an element is positioned in the document flow and how it
interacts with other elements.

### **Position Values:**

1. **`static`** (default):
   - The default positioning for all HTML elements.
   - Elements are positioned according to the normal document flow
   - `top`, `right`, `bottom`, `left`, and `z-index` properties do not apply.

   **Example**:
   ```css
   div {
       position: static;
   }
   ```


2. **`relative`**:

   - The element is positioned relative to its normal position in the document
flow.
   - The `top`, `right`, `bottom`, and `left` properties can be used to adjust
the element's position relative to its original position.
   - Other elements' positions are not affected.

   **Example**:
   ```css
   div {
       position: relative;
       top: 20px;
       left: 10px;
   }
   ```

   In this example, the `div` is shifted 20px down and 10px to the right from
where it would normally be positioned.
~~~

3. **`absolute`**:

   - The element is removed from the normal document flow and positioned relative to its nearest positioned ancestor (i.e., an ancestor with a `relative`, `absolute`, or `fixed` position).
   - If no such ancestor exists, it is positioned relative to the initial containing block (usually the viewport).
   - `top`, `right`, `bottom`, and `left` properties are used to position the element.

   Example:
   ```css
   div {
       position: absolute;
       top: 50px;
       left: 30px;
   }
   ```

   In this example, the `div` is positioned 50px from the top and 30px from the left of its nearest positioned ancestor.

4. **`fixed`**:

   - The element is removed from the normal document flow and positioned relative to the viewport.
   - The element stays fixed in position even when the page is scrolled.
   - `top`, `right`, `bottom`, and `left` properties are used to position the element.

   Example:
   ```css
   div {
       position: fixed;
       top: 10px;
       right: 10px;
   }
   ```

   In this example, the `div` remains fixed 10px from the top and right of the viewport, even when the page is scrolled.

~~~markdown
5. **`sticky`**:

   - The element is positioned based on the user's scroll position.
   - It toggles between `relative` and `fixed`, depending on the scroll position.
   - The element behaves like `relative` until a given offset position is met, then it behaves like `fixed`.
   - The `top`, `right`, `bottom`, or `left` properties define the threshold where the element becomes sticky.

   **Example**:
   ```css
   div {
       position: sticky;
       top: 0;
   }
   ```

   In this example, the `div` will stick to the top of the viewport once the user scrolls past it.


                    <!-- !   z-index -->

- The `z-index` property controls the stacking order of positioned elements (elements with `relative`, `absolute`, `fixed`, or `sticky` position).
- Higher `z-index` values stack elements in front of those with lower values.

**Example**:
```css
div {
    position: absolute;
    z-index: 1;
}
```
~~~