Position Property

Position Property in

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:

```
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 adjustthe element's position relative to its original position.
   - Other elements' positions are not affected.

   Example:

```
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 theelement.

   Example:

```
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 theelement.

   Example:

```
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.

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 thresholdwhere the element becomes sticky.

Example:

```
div {
    position: sticky;
    top: 100px;
}
```

<!-- ! 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:

```
div {
    position: absolute;
    z-index: 1;
}
```

Difference Between Relative And Absolute Position

Feature	position: relative	position: absolute
Definition	Positions the element **relative to its normal/static position**.	Positions the element **relative to the nearest positioned ancestor** or the viewport if no ancestor is positioned.
Document Flow	Element **remains in the document flow**; its space is reserved.	Element is **removed from the document flow**, and does not occupy space.
Positioning Reference	Uses its **original position** as the reference for top, right, bottom, and left.	Uses the **nearest positioned ancestor** (with relative, absolute, or fixed) or the viewport as the reference.
Sibling Elements	Does not overlap or affect the positioning of sibling elements.	May overlap or affect sibling elements, as it's removed from the flow.
Example CSS	position: relative; top: 20px; left: 30px;	position: absolute; top: 10px; left: 15px;