Sure! Here's a detailed explanation of each CSS style:

1. \*\*`display: block;`\*\*:

- This property sets an element to display as a block-level element. Block-level elements occupy the full width available and start on a new line. Examples include `<div>`, `<h1>`, `<p>`, etc.

```css

div {

display: block;

}

```

2. \*\*`position: relative;`\*\*:

- This property sets an element's position relative to its normal position. The element remains in the document flow, but you can use the `top`, `right`, `bottom`, and `left` properties to move it relative to where it would normally be.

```css

div {

position: relative;

top: 10px;

left: 20px;

}

```

3. \*\*`position: absolute;`\*\*:

- This property sets an element's position relative to its nearest positioned ancestor (not the viewport), or the initial containing block if no positioned ancestors exist. The element is removed from the document flow.

```css

div {

position: absolute;

top: 50px;

left: 100px;

}

```

4. \*\*`transform: translate(-50%, -50%);`\*\*:

- This property moves an element along the X and Y axes. The values `-50%` and `-50%` move the element left and up by 50% of its own width and height, respectively. It's often used for centering an element.

```css

div {

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

}

```

5. \*\*`z-index: 2;`\*\*:

- This property specifies the stack order of an element. An element with a higher `z-index` is in front of an element with a lower `z-index`. This only works on positioned elements (`position` other than `static`).

```css

div {

position: relative;

z-index: 2;

}

```

6. \*\*`flex-basis: 31%;`\*\*:

- This property sets the initial main size of a flex item. It determines the size of the flex item before the remaining space is distributed based on the flex-grow and flex-shrink properties.

```css

.flex-item {

flex-basis: 31%;

}

```

7. \*\*`box-shadow`\*\*:

- This property adds shadow effects around an element's frame. You can set multiple shadows separated by commas. The values specify the horizontal offset, vertical offset, blur radius, spread radius, and color.

```css

div {

box-shadow: 10px 10px 5px #888888;

}

```

8. \*\*`overflow: hidden;`\*\*:

- This property specifies what happens if content overflows an element's box. `hidden` means that the overflow is clipped, and the rest of the content will be invisible.

```css

div {

overflow: hidden;

}

```

9. \*\*`justify-content: space-between;`\*\*:

- This property aligns flex items along the main axis of the container, with space between them. The first item is at the start, the last item is at the end, and the remaining items are distributed evenly between.

```css

.flex-container {

display: flex;

justify-content: space-between;

}

```

10. \*\*`transform: translateX(-50%);`\*\*:

- This property moves an element along the X-axis by -50% of its own width. It's often used for centering an element horizontally.

```css

div {

position: absolute;

left: 50%;

transform: translateX(-50%);

}

```