1. Explain all the CSS positions(static, fixed, sticky, relative, absolute) with one code example each.

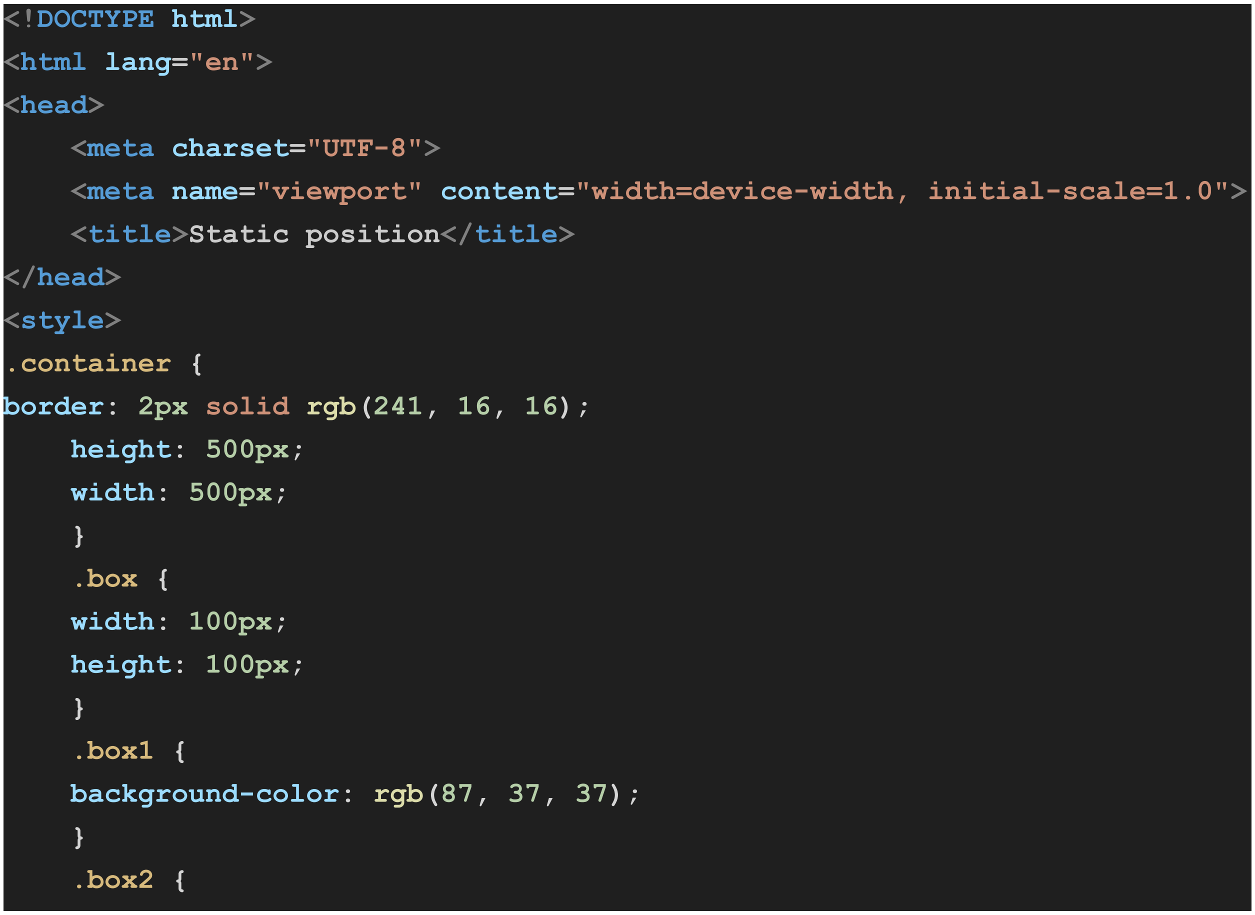
Ans:-

CSS Position:- In CSS, positioning refers to the process of controlling the layout and placement of elements on a webpage.

There are five main positioning types available in CSS

1. Static
2. Relativ
3. Absolut
4. Fixed
5. Sticky

Static Positions:- This is the default position for all HTML elements.Elements with static positioning are positioned based on the normal document flow, meaning they are placed one after the other in the order they appear in the HTML code.



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Static position</title>
</head>
<style>
.container {
border: 2px solid rgb(241, 16, 16);
    height: 500px;
    width: 500px;
    }
    .box {
    width: 100px;
    height: 100px;
    }
    .box1 {
    background-color: rgb(87, 37, 37);
    }
    .box2 {
```

```
background-color: rgb(171, 218, 171);
position: static;
top: 50px;
}
.box3 {
background-color: rgb(152, 152, 188);
}
</style>
<body>
<div class="container">
<div class="box box1"></div>
<div class="box box2"></div>
<div class="box box3"></div>
</div>
</body>
</html>
```

As we can see in below output that the default output and this one is the same even after applying the topproperty on box2 which will provide a space from top of 20px as per code, but has no effect because we are using the static property which is the default one and the properties like top, bottom, left and right has no effect in this.

Relative Positions:- With relative positioning, an element can be moved up, down, left or right from its normal position, but it still takes up its original space in the document flow.

```
<!DOCTYPE html>
<html lang="en-US">

<head>
    <meta charset="UTF-8">
    <meta nam.container1e="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Relative position</title>
</head>
<style>
    .container {
        border: 2px solid black;
        height: 500px;
        width: 500px;
    }

    .box {
        width: 100px;
        height: 100px;
    }

    .box1 {
        background-color: red;
    }

    .box2 {
        position: relative;
        top: 50px;
        left: 50px;
        background-color: green;
    }

    .box3 {
        background-color: blue;

    }
</style>

<body>
    <div class="container">
        <div class="box box1"></div>
        <div class="box box2"></div>
        <div class="box box3"></div>
    </div>
</body>

</html>
```

After applying the position relative, we can see that box2 has moved from its top by 50px and left position by 50px, leaving space at its original position and also not breaking the document flow.

Absolute Positions:- **With absolute positioning, an element is positioned relative to its nearest positioned ancestor. If there is no positioned ancestor, then the element is positioned relative to the document body.It will break the normal document flow to position the element on the page. The properties like top, left, right, bottom and z-index will have an effect on the element.The element will not leave any space at its original position.**

```
<!DOCTYPE html>
<html lang="en-US">

<head>
    <meta charset="UTF-8">
    <meta nam.container1e="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Absolute position</title>
</head>
<style>
    .container {

        border: 2px solid black;

        height: 500px;

        width: 500px;

        position: relative;

    }

    .box {
        width: 100px;
        height: 100px;
    }

    .box1 {
        background-color: red;
    }

    .box2 {
        position: absolute;

        top: 50px;

        left: 50px;

        background-color: green;

    }

    .box3 {
        background-color: blue;

    }
</style>

<body>
    <div class="container">
        <div class="box box1"></div>
        <div class="box box2"></div>
        <div class="box box3"></div>
    </div>
</body>

</html>
```

As we can see in the output below, the box2 has come out of the flow of the document and is also positioned 50px left and 50px top with respect to the container because it is an positioned ancestor.

Fixed Positions:- **Fixed positioning is similar to absolute positioning, but the element is positioned relative to the viewport instead of an ancestor element. This means that the element stays in the same position even if the user scrolls the web page.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Fixed position</title>
</head>
<style>
.container {
border: 2px solid black;
height: 150px;
width: 500px;
position: relative;
overflow: scroll;
}
    .box {
    width: 100px;
    height: 100px;
    }
    .box1 {
    background-color: rgb(87, 37, 37);
    }
    .box2 {
position: fixed;
top: 100px;
left: 100px;
background-color: green;

}
.box3 {
background-color: rgb(152, 152, 188);
}
</style>
<body>
<div class="container">
<div class="box box1"></div>
<div class="box box2"></div>
<div class="box box3"></div>
</div>
</body>
</html>
```

Sticky Positions:- An element with sticky positioning is positioned relative to its nearest ancestor with a scrolling mechanism, which means that the element behaves like a relatively positioned element until the user scrolls to a certain point, and then it becomes fixed to the viewport.

```
<!DOCTYPE html>
<html lang="en-US">

<head>
    <meta charset="UTF-8">
    <meta nam.container1e="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Sticky position</title>
</head>
<style>
    .container {
        border: 2px solid black;
        height: 150px;
        width: 500px;
        position: relative;
        overflow: scroll;
    }
.box {
        width: 100px;
        height: 100px;
    }
.box1 {
        background-color: red;
    }
    .box2 {
        background-color: green;
        position: sticky;
        top: 10px;
        left: 50px;
    }
    .box3 {
        background-color: blue;
    }
</style>
```

```html
<body>
    <div class="container">
        <div class="box box1"></div>
        <div class="box box2"></div>
        <div class="box box3"></div>
    </div>
</body>

</html>
```