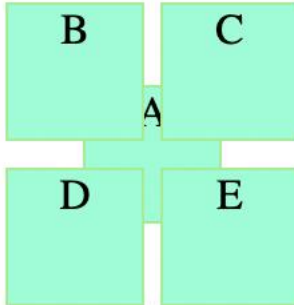


1. Write code to position 3 equal-sized (30 X 30px) boxes A/ B/ C/ D/ and E as follows.

```
Answer:- <!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<style>
.box {
width: 50px;
height: 50px;
background-color: aquamarine;
border: 1px solid lightgreen;
text-align: center;
}
.boxA {
position: relative;
top: 200px;
left: 200px;
}
.boxB {
position: absolute;
left: -30px;
bottom: 30px;
}
.boxC {
position: absolute;
right: -30px;
bottom: 30px;
}
.boxD {
position: absolute;
left: -30px;
top: 30px;
}
.boxE {
position: absolute;
right: -30px;
top: 30px;
}
</style>
</head>
<body>
<div class="box boxA">
A
<div class="box boxB">B</div>
<div class="box boxC">C</div>
<div class="box boxD">D</div>
<div class="box boxE">E</div>
</div>
</body>
</html>
```



Output:-

2. Explain the difference between Absolute and Relative positioning.

Answer:-

### **Relative Positioning**

When position:relative is given to an element without any other attributes (top, bottom, left, right) nothing will happen.

But when an attribute left:20px is added the element moves 20px to the right from its normal position. The element effects no other element on the layout. Other content will not be adjusted to fit into any gap left by the element.

The child of this element can be positioned within this block only. So position:relative has its limitations.

### **Absolute Positioning**

When position:absolute is given to an element the element can be placed precisely where one wants to.

The element is positioned nearest to the first relatively or absolutely positioned parent element.

When there is no parent element the absolutely positioned element is positioned directly to the HTML page itself.

3. Create a card as shown in the picture below. (You can use CSS float property only for layout).

Answer:- `<!DOCTYPE html>`

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```

<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<style>
.card {
padding: 10px;
background-color: beige;
border-radius: 5px;
}
.card-image {
float: right;
padding-left: 10px;
padding-right: 10px;
}</style>
</head>
<body>
<div class="card">

<div class="card-body">
<h3 class="card-title">The Earth!!!</h3>
<p>
Nature is Earth's masterpiece, a symphony of life composed through
millennia. Its landscapes whisper tales of ancient mountains, winding
rivers, and flourishing forests, reminding us of the planet's
boundless beauty. From delicate petals to towering redwoods, nature's
artistry stirs awe and reconnects us to the essence of existence.
Nature is Earth's masterpiece, a symphony of life composed through
millennia. Its landscapes whisper tales of ancient mountains, winding
rivers, and flourishing forests.
</p>
</div>
</div>
</body>
</html>

```

4. Create a simple header that sticks to the top of a webpage upon scrolling.

```

Answer: <!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link rel="stylesheet" href="Css.css" />
<title>Document</title>
<body>
<div class="main">
<header class="header">
<div class="logo">LOGO</div>
<div class="nav">
<ul>
<li class="nav-item active">Home</li>
<li class="nav-item">Pricing</li>

```

```

<li class="nav-item">Contact</li>
<li class="nav-item">Terms and Condition</li>
<li class="nav-item button"><button>Get Started</button></li>
</ul>
</div>
</header>
<main>
<section>
<h3>Heading 1</h3>
<p>
Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsum ossimus modi magni
deleniti rerum provident minima nisi cum delectus, inventore, ...
</p>
</section>
<section>
<h3>Heading 2</h3>
<p>
Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsum possimus modi
magni deleniti rerum provident minima nisi cum
delectus, inventore, facere corrupti ...
</p>
</section>
<section>
<h3>Heading 3</h3>
<p>
Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsum possimus modi
magni deleniti rerum provident minima nisi...
</p>
</section>
</main>
</div>
</body>
</html>

```

**CSS Style Sheet:** \* {

```
padding: 0px;
margin: 0px;
}
```

```
.header {
position: sticky;
top: 0px;
height: 80px;
color: #f0efee;
background-color: #2b2922;
}
```

```
.logo {
position: absolute;
top: 0px;
bottom: 0px;
left: 20px;
margin: auto 0px;
height: fit-content;
font-weight: bold;
}
```

```
.logo img {
```

```

width: 120px;
}
.nav {
position: absolute;
top: 0px;
bottom: 0px;
right: 0px;
margin: auto 0px;
height: fit-content;
font-family: "Red Rose", cursive;
}
.nav-item {
display: inline;
margin: 0px 15px;
color: #63615a;
}
.nav-item.active {
color: #ffffff;
}
.nav-item.button > button {
padding: 10px 20px;
background: transparent;
border: 2px solid #a08f36;
border-radius: 10px;
color: #a08f36;
font-family: "Red Rose", cursive;
}
main {
padding: 20px;
}
section {
margin-top: 20px;
}

```

5. Explain the z-index, with a code example.

Answer:- The CSS z-index property controls the stacking order of positioned elements on a webpage. It determines which elements appear in front or behind others when they overlap. Higher `z-index` values bring elements forward, while lower values push them back, enhancing layout control.

The z-index property is used to displace elements on the z-axis, i.e., in or out of the screen. It defines the order of elements if they overlap each other.

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<style>
.container {
position: relative;
}

```

```

.black-box {
position: relative;
z-index: 1;
border: 2px solid black;
height: 100px;
margin: 30px;
}

.gray-box {
position: absolute;
z-index: 3; /* gray box will be above both green and black box */
background: lightgray;
height: 60px;
width: 70%;
left: 50px;
top: 50px;
}

.green-box {
position: absolute;
z-index: 2; /* green box will be above black box */
background: lightgreen;
width: 35%;
left: 270px;
top: -15px;
height: 100px;
}
</style>
</head>
<body>
<h1>Z-index Example</h1>

<p>An element with greater stack order is always above an element with a lower stack order.</p>

<div class="container">
<div class="black-box">Black box (z-index: 1)</div>
<div class="gray-box">Gray box (z-index: 3)</div>
<div class="green-box">Green box (z-index: 2)</div>
</div>

</body>
</html>

```

Output:-

## Z-index Example

An element with greater stack order is always above an element with a lower stack order.



