Project 201

DP EDUCATION

Coding School

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel= "stylesheet"
    href= "imageOpacity.css">
    <title>Document</title>
</head>
<body>
    <div class="demo-wrap">
        <img
          class="img-bg"
          background-image src="scenery.jpg"
          alt=""
        >
        <div class="text">
          <h1>Hello World!</h1>
        </div>
```

html

Glitch

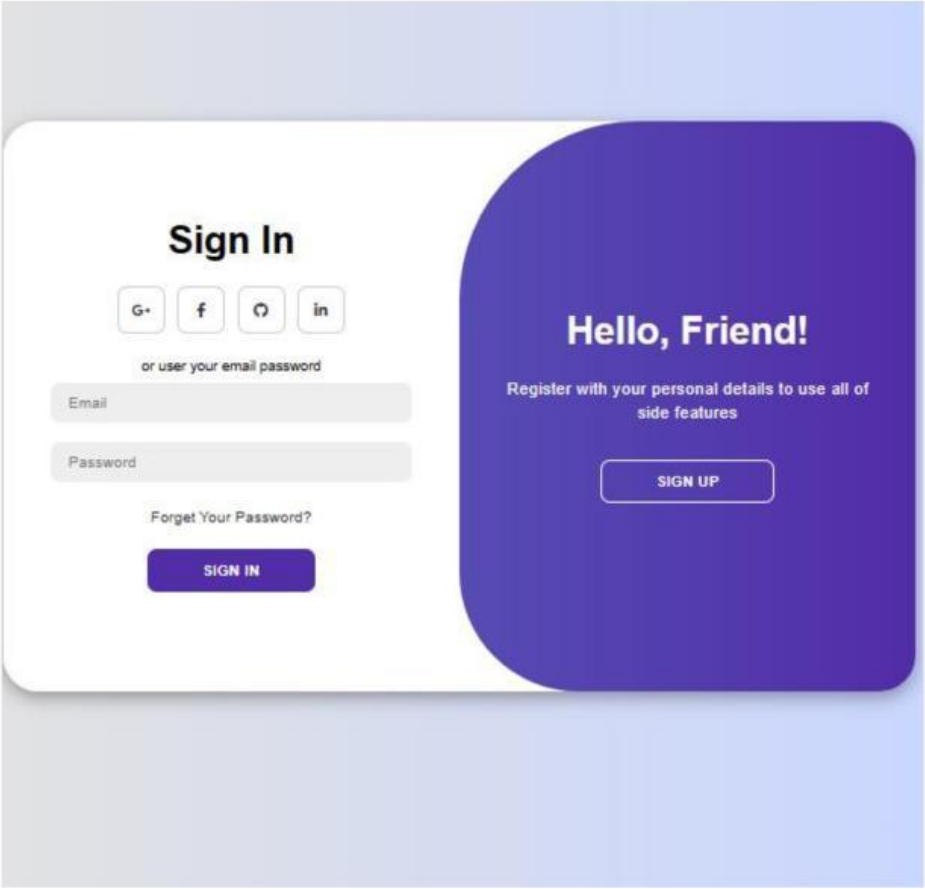
See the webpage

Glitch Start here

LIVEWORKSHEETS

Let's create a sign in page using HTML and CSS.

❖ By preparing this project, you can create a web page section as follows.



To create a section as above, HTML tags should be created as follows.

❖ First of all, prepare the basic tags in the index.html file as follows.

```html
<!DOCTYPE html>
<html lang="en">
  <head>

    <title>Modern Login Page | AsmrProg</title>
  </head>
  <body>

  </body>
</html>
```

LIVEWORKSHEETS

- Set a meta tag as follows in the head tag

```
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
```

- Then prepare the link tag as below below the title tag.

  The <link rel="stylesheet" href="style.css"> is used to add the style file to the html file using this link tag.

  In the same way, let's add the file related to the following link to the html file.

  https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css

  The file related to the above link should also be added to the html file.

  The relevant code to add the style file to the html file and the related file to the link is as follows

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css">
<link rel="stylesheet" href="style.css">
```

- Open a div tag in the body tag. Add a class as container for that div tag and add an id as container to that div tag
- Open another div tag in that div tag and add a class as form-container sign-in.
- Open a form tag in the div tag that added a class as form-container sign-in.
- Open an h1 tag in the form tag. Add a Sign In for that h1 tag

```
<div class="container" id="container">
    <div class="form-container sign-in">
        <form>
            <h1>Sign In</h1>
        </form>
    </div>
</div>
```

- Open a div tag below the h1 tag. Add a class as social-icons for that div tag.

- Open a tag in that div tag. Add a class as icon for that a tag. Prepare an i tag for that a tag. For that i tag you should add a class related to an icon..

LIVEWORKSHEETS

Here, first of all, let's add an icon related to google. The class related to that icon is shown below.

fa-brands fa-google-plus-g

- As above, give the relevant icon by the following class.
  fa-brands fa-facebook-f
  fa-brands fa-github
  fa-brands fa-linkedin-in

```
<div class="social-icons">
    <a href="#" class="icon"><i class="fa-brands fa-google-plus-g"></i></a>
    <a href="#" class="icon"><i class="fa-brands fa-facebook-f"></i></a>
    <a href="#" class="icon"><i class="fa-brands fa-github"></i></a>
    <a href="#" class="icon"><i class="fa-brands fa-linkedin-in"></i></a>
</div>
```

- Add a span tag below the div tag related to the social-icons class. Give r user as your email password in that span tag.
- Open two input tags below the div tag. Set the tag to provide the Email through one input and the password through the other input.

- Add a tag. Forgot Your Password for a tag? Add a text as.

- Let's add a button tag to add a button. Give the text of that button as Sign In.

```
<span>or user your email password</span>
<input type="email" placeholder="Email">
<input type="password" placeholder="Password">
<a href="#">Forget Your Password?</a>
<button>Sign In</button>
```

- Add a div tag below the close tag in the div tag that has added a class as form-container sign-in. Add a class as toggle-container for that div tag.
- Open another div tag inside the div tag that has added a class as toggle-container. Add a toggle class for that div tag.

- Open another div tag in the div tag that has added a class as a toggle. Add a class as toggle-panel toggle-right for that div tag.

LIVEWORKSHEETS

- Add a class as toggle-panel toggle-right and open an h1 tag in the div tag.
- For that h1 tag Hello, Friend! Add one as
- Open a p tag below the h1 tag. Add a text as Register with your personal details to use all of side features for that p tag.
- Add a button below the P tag. Add a class as hidden and an id as register for that button. Give the button text as Sign Up

- Add a button below the P tag. Add a class as hidden and an id as register for that button. Give it as Sign Up for the button text.
  Let's now create the CSS section related to this project.

```html
<div class="toggle-container">
    <div class="toggle">
        <div class="toggle-panel toggle-right">
            <h1>Hello, Friend!</h1>
            <p>Register with your personal details to use all of side features</p>
            <button class="hidden" id="register">Sign Up</button>
        </div>
    </div>
</div>
```

- Adjust the CSS as follows for the entire web page.

```css
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: 'Montserrat', sans-serif;
}
```

- Prepare the CSS added for the body as follows.

```css
body{
    background-color: #c9d6ff;
    background: linear-gradient(to right, #e2e2e2, #c9d6ff);
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column;
    height: 100vh;
}
```

- Prepare the CSS for the container class as follows.

LIVEWORKSHEETS

```
.container{
    background-color: #fff;
    border-radius: 30px;
    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.35);
    position: relative;
    overflow: hidden;
    width: 768px;
    max-width: 100%;
    min-height: 480px;
}
```

❖ Prepare the CSS as follows for the p tag in the container class.

```
.container p{
    font-size: 14px;
    line-height: 20px;
    letter-spacing: 0.3px;
    margin: 20px 0;
}
```

❖ Prepare the CSS as follows for the span tag in the container class.

```
.container span{
    font-size: 12px;
}
```

❖ Prepare the CSS for the a tag in the container class as follows.

```
.container a{
    color: #333;
    font-size: 13px;
    text-decoration: none;
    margin: 15px 0 10px;
}
```

❖ Prepare the CSS for the button tag in the container class as follows.

```
.container button{
    background-color: #512da8;
    color: #fff;
    font-size: 12px;
    padding: 10px 45px;
    border: 1px solid transparent;
    border-radius: 8px;
    font-weight: 600;
    letter-spacing: 0.5px;
    text-transform: uppercase;
    margin-top: 10px;
    cursor: pointer;
}
```

❖ Prepare the CSS as follows for the hidden class added to the button tag of the container class.

```
.container button.hidden{
    background-color: transparent;
    border-color: #fff;
}
```

LIVEWORKSHEETS

❖ Prepare the CSS as follows for the form tag in the container class.

```
.container form{
    background-color: #fff;
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column;
    padding: 0 40px;
    height: 100%;
}
```

❖ Prepare the CSS as follows for the input tag in the container class.

```
.container input{
    background-color: #eee;
    border: none;
    margin: 8px 0;
    padding: 10px 15px;
    font-size: 13px;
    border-radius: 8px;
    width: 100%;
    outline: none;
}
```

❖ Prepare the CSS for the form-container class as follows.

```
.form-container{
    position: absolute;
    top: 0;
    height: 100%;
    transition: all 0.6s ease-in-out;
}
```

❖ Prepare the CSS for the sign-in class as follows.

```
.sign-in{
    left: 0;
    width: 50%;
    z-index: 2;
}
```

❖ Prepare a css set as follows.

```
.container.active .sign-in{
    transform: translateX(100%);
}
```

LIVEWORKSHEETS

- When the size of the devices changes, the size of the web page changes. Adjust the CSS as follows to change the opacity and z-index.

```
@keyframes move{
    0%, 49.99%{
        opacity: 0;
        z-index: 1;
    }
    50%, 100%{
        opacity: 1;
        z-index: 5;
    }
}
```

- Prepare the CSS for the social-icons class as follows.

```
.social-icons{
    margin: 20px 0;
}
```

- Prepare the CSS for the a tag in the social-icons class as follows.

```
.social-icons a{
    border: 1px solid #ccc;
    border-radius: 20%;
    display: inline-flex;
    justify-content: center;
    align-items: center;
    margin: 0 3px;
    width: 40px;
    height: 40px;
}
```

- Prepare the CSS for the toggle-container class as follows.

```
.toggle-container{
    position: absolute;
    top: 0;
    left: 50%;
    width: 50%;
    height: 100%;
    overflow: hidden;
    transition: all 0.6s ease-in-out;
    border-radius: 150px 0 0 100px;
    z-index: 1000;
}
```

- Prepare a css set as follows.

```
.container.active .toggle-container{
    transform: translateX(-100%);
    border-radius: 0 150px 100px 0;
}
```

- Prepare the CSS for the toggle class as follows.

LIVEWORKSHEETS

```
.toggle{
    background-color: #512da8;
    height: 100%;
    background: linear-gradient(to right, #5c6bc0, #512da8);
    color: #fff;
    position: relative;
    left: -100%;
    height: 100%;
    width: 200%;
    transform: translateX(0);
    transition: all 0.6s ease-in-out;
}
```

- Prepare a css set as follows.

```
.container.active .toggle{
    transform: translateX(50%);
}
```

- Prepare the CSS for the toggle-panel class as follows.

```
.toggle-panel{
    position: absolute;
    width: 50%;
    height: 100%;
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column;
    padding: 0 30px;
    text-align: center;
    top: 0;
    transform: translateX(0);
    transition: all 0.6s ease-in-out;
}
```

- Adjust the CSS for the toggle-right class as follows.

```
.toggle-right{
    right: 0;
    transform: translateX(0);
}
```

- Prepare a css set as follows.

```
.container.active .toggle-right{
    transform: translateX(200%);
}
```

LIVEWORKSHEETS