**\*\*\*\*\*Assignment No. 2\*\*\*\*\***

**\*\*\*\*\*TYCOB129\*\*\*\*\***

**1.form.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<link rel="stylesheet" href="style.css" />

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.3.0/font/bootstrap-icons.css" />

<title>Form Validation</title>

</head>

<body>

<h1><span class="cf-gradient-span">Sign Up</span></h1>

<div class="container">

<form id="signup" class="form">

<div class="form-field">

<label for="username">Username:</label>

<input type="text" name="username" id="username" autocomplete="off">

<small></small>

</div>

<div class="form-field">

<label for="email">Email:</label>

<input type="text" name="email" id="email" autocomplete="off">

<small></small>

</div>

<div class="form-field">

<label for="password">Password:</label>

<input type="password" name="password" id="password" autocomplete="off">

<small></small>

</div>

<div class="form-field">

<label for="confirm-password">Confirm Password:</label>

<div class="form-field input-group">

<input type="password" name="confirm\_password" id="confirm-password" autocomplete="off">

<i class="bi bi-eye-slash" id="togglePassword"></i>

<small></small>

</div>

</div>

<div class="form-field">

<input class="btn" type="submit" value="Sign Up">

</div>

</form>

</div>

<script src="scripts.js"></script>

</body>

</html>

**2.style.css**

@import url('https://fonts.googleapis.com/css?family=Open+Sans&display=swap');

:root {

--error-color: #dc3545;

--success-color: #28a745;

--warning-color: #ffc107;

}

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Open Sans', sans-serif;

}

body {

font-family: 'Open Sans', sans-serif;

font-size: 16px;

background-color: #26282A;

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

min-height: 100vh;

background-image: url("/imagees/4.jpg");

margin: 0;

}

.cf-gradient-span{

-webkit-text-fill-color: transparent;

background-image: linear-gradient(to right,#b16cea 20%, #ff5e69 60%, #ff8a56 80%, #ffa84b);

-webkit-background-clip: text;

background-clip: text;

font-family: Verdana, Tahoma, sans-serif;

font-size: 35px;

}

.container {

background-color: #fff;

margin-top: 2em;

padding: 1em;

border-radius: 5px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.3);

width: 400px;

}

.form {

padding: 10px 20px;

}

.form h1 {

font-size: 1.5em;

text-align: center;

margin-bottom: 20px;

}

.form-field {

margin-bottom: 5px;

}

.form-field label {

display: block;

color: black;

margin-bottom: 5px;

}

.form-field input {

border: solid 2px #0a0a0a;

padding: 10px;

margin-bottom: 5px;

font-size: 14px;

display: block;

width: 100%;

}

.form-field input:focus {

outline: none;

}

.form-field.error input {

border-color: var(--error-color);

}

.form-field.success input {

border-color: var(--success-color);

}

.form-field small {

color: var(--error-color);

}

.input-group{

position: relative;

}

.bi-eye-slash{

position: absolute;

top: 15%;

right: 5%;

cursor: pointer;

}

/\* button \*/

.btn {

width: 75%;

padding: 3%;

border-radius: 20px;

z-index: auto;

background: #ffa84b;

background-image: linear-gradient(to right,#b16cea 20%, #ff5e69 60%, #ff8a56 80%, #ffa84b);

color: #fff;

text-transform: uppercase;

}

.btn:hover {

cursor: pointer;

}

.btn:focus {

outline: none;

}

**3.srcipts.js**

const username = document.querySelector('#username');

const email = document.querySelector('#email');

const password = document.querySelector('#password');

const confirmPassword = document.querySelector('#confirm-password');

const form = document.querySelector('#signup');

const togglePassword = document.querySelector("#togglePassword");

const showError = (input, message) => {

const formField = input.parentElement;

formField.classList.remove('success');

formField.classList.add('error');

const error = formField.querySelector('small');

error.textContent = message;

};

const showSuccess = (input) => {

const formField = input.parentElement;

formField.classList.remove('error');

formField.classList.add('success');

const error = formField.querySelector('small');

error.textContent = '';

};

const isRequired = value => value === '' ? false : true;

const isBetween = (length, min, max) => length < min || length > max ? false : true;

const isEmailValid = (email) => {

const re = /^(([^<>()\[\]\\.,;:\s@"]+(\.[^<>()\[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;

return re.test(email);

};

const isPasswordSecure = (password) => {

const re = new RegExp("^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[0-9])(?=.\*[!@#\$%\^&\\*])(?=.{8,})");

return re.test(password);

};

const checkUsername = () => {

let valid = false;

const min = 3,

max = 25;

const usernameVal = username.value.trim();

if (!isRequired(usernameVal)) {

showError(username, 'Username cannot be blank.');

} else if (!isBetween(usernameVal.length, min, max)) {

showError(username, `Username must be between ${min} and ${max} characters.`)

} else {

showSuccess(username);

valid = true;

}

return valid;

};

const checkEmail = () => {

let valid = false;

const emailVal = email.value.trim();

if (!isRequired(emailVal)) {

showError(email, 'Email cannot be blank.');

} else if (!isEmailValid(emailVal)) {

showError(email, 'Email is not valid.')

} else {

showSuccess(email);

valid = true;

}

return valid;

};

const checkPassword = () => {

let valid = false;

const passwordVal = password.value.trim();

if (!isRequired(passwordVal)) {

showError(password, 'Password cannot be blank.');

} else if (!isPasswordSecure(passwordVal)) {

showError(password, 'Password must has at least 8 characters that include at least 1 lowercase character, 1 uppercase characters, 1 number, and 1 special character in (!@#$%^&\*)');

} else {

showSuccess(password);

valid = true;

}

return valid;

};

const checkConfirmPassword = () => {

let valid = false;

const confirmPasswordVal = confirmPassword.value.trim();

const passwordVal = password.value.trim();

if (!isRequired(confirmPasswordVal)) {

showError(confirmPassword, 'Confirm Password is required');

} else if (passwordVal !== confirmPasswordVal) {

showError(confirmPassword, 'Confirm Password does not match');

} else {

showSuccess(confirmPassword);

valid = true;

}

return valid;

};

togglePassword.addEventListener("click", function () {

const type = confirmPassword.getAttribute("type") === "password" ? "text" : "password";

confirmPassword.setAttribute("type", type);

this.classList.toggle("bi-eye");

});

form.addEventListener('submit', function (e) {

e.preventDefault();

let isUsernameValid = checkUsername(),

isEmailValid = checkEmail(),

isPasswordValid = checkPassword(),

isConfirmPasswordValid = checkConfirmPassword();

let isFormValid = isUsernameValid &&

isEmailValid &&

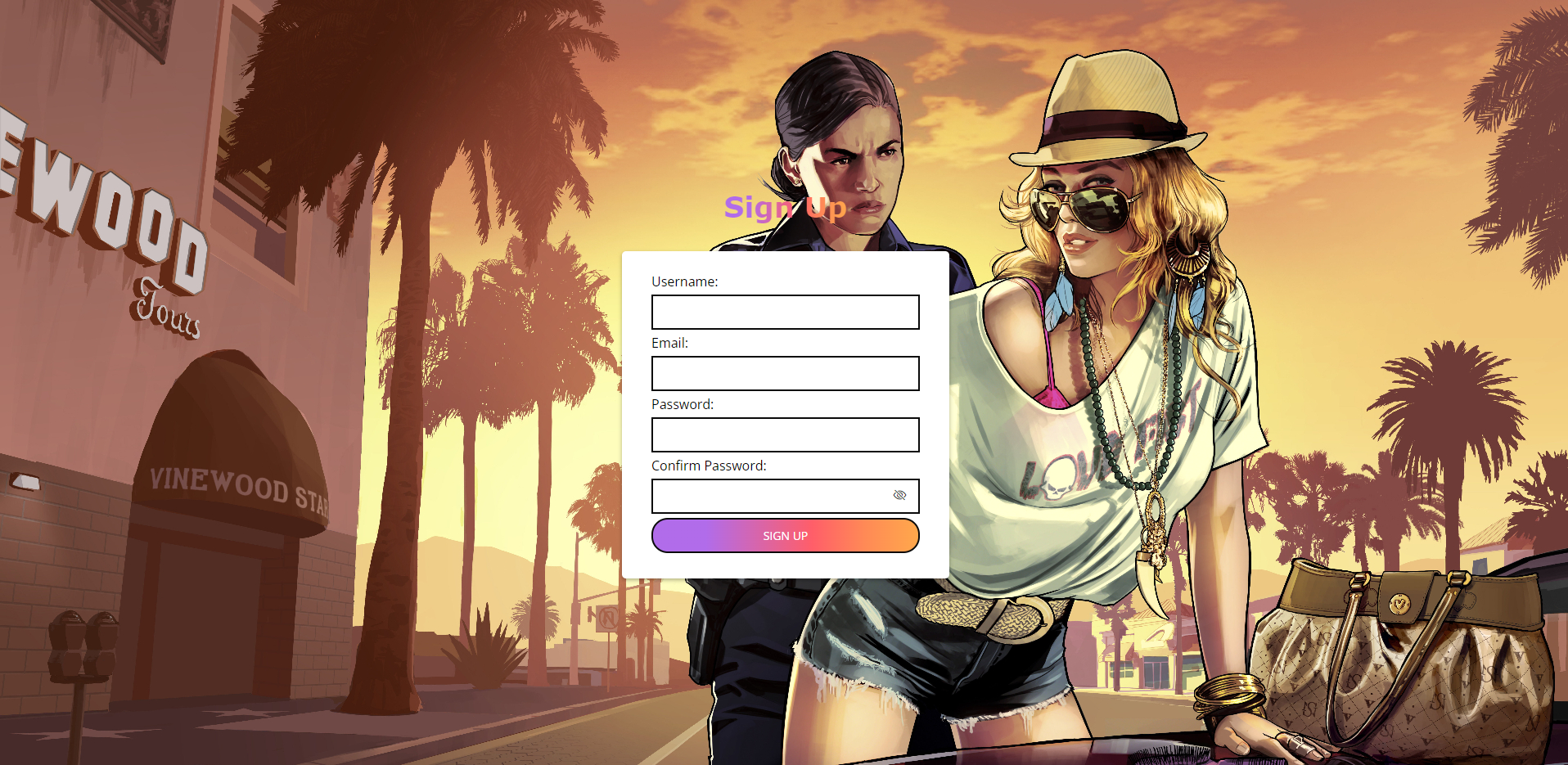
isPasswordValid &&

isConfirmPasswordValid;

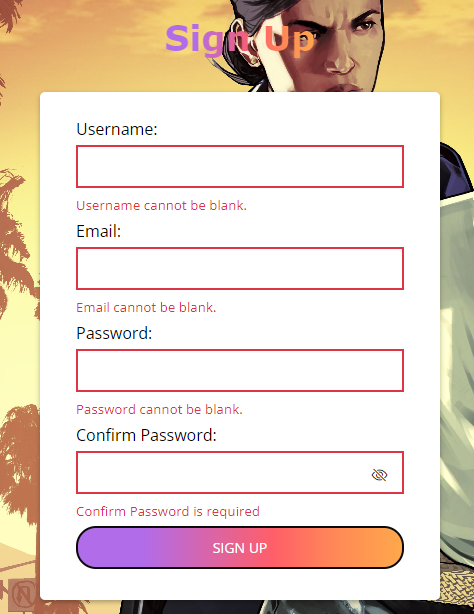
if (isFormValid) {

}

});

**Output :** 

A screenshot of a computer screen

Description automatically generated

A screenshot of a login form

Description automatically generated