

## ASSIGNMENT

1. Create a personalized weather widget using HTML, CSS, and JavaScript. The widget should fetch real-time weather data from an API and display it dynamically. Users should be able to enter their location to get localized weather updates. Add styling to make the widget visually appealing. (Create)

### SOURCE CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Weather Widget</title>
  <link rel="stylesheet" href="style.css">
  <link rel="icon" href="./logo1.jpg">
</head>
<body>
  <div class="card">
    <div class="search">
      <input type="text" placeholder="Enter City Name"
spellcheck="false">
      <button></button>
    </div>
    <div class="error">
      <p>Invalid city Name</p>
    </div>
    <div class="weather">
      
      <h1 class="temp">22°C</h1>
      <h2 class="city">New York</h2>
      <div class="details">
        <div class="col">
          
          <div>
            <p class="humidity">50%</p>
            <p>Humidity</p>
          </div>
        </div>
        <div class="col">
          
          <div>
```

```
<p class="wind">15 km/h</p>  
<p>Wind Speed</p>  
</div>  
</div>  
</div>  
</div>  
  
<script>  
    const apiKey = "dcd8a1c6d5e14f4b89bdcc2c8a17bfb40";  
    const apiUrl =  
"https://api.openweathermap.org/data/2.5/weather?units=metric&q=";  
  
    const searchBox = document.querySelector(".search input");  
    const searchBtn = document.querySelector(".search  
button");  
    const weatherIcon = document.querySelector(".weather-  
icon");  
  
    async function checkWeather(city){  
        const response = await fetch(apiUrl + city +  
`&appid=${apiKey}`);  
  
        if(response.status == 404){  
            document.querySelector(".error").style.display =  
"block";  
            document.querySelector(".weather").style.display  
= "none";  
        }  
        else{  
            var data = await response.json();  
            document.querySelector(".city").innerHTML =  
data.name;  
            document.querySelector(".temp").innerHTML =  
Math.round(data.main.temp) + "°C";  
            document.querySelector(".humidity").innerHTM  
L = data.main.humidity + "%";  
            document.querySelector(".wind").innerHTML =  
data.wind.speed + "km/h";  
  
            if(data.weather[0].main == "clouds"){  
                weatherIcon.src = "images/clouds.png";  
            }  
            else if(data.weather[0].main == "clear"){  
                weatherIcon.src = "images/clear.png";  
            }  
            else if(data.weather[0].main == "Rain"){  
                weatherIcon.src = "images/rain.png";  
            }  
        }  
    }  
}
```

```

        else if(data.weather[0].main == "Drizzle"){
            weatherIcon.src = "images/drizzle.png";
        }
        else if(data.weather[0].main == "Mist"){
            weatherIcon.src = "images/mist.png";
        }

        document.querySelector(".weather").style.display = "block";
        document.querySelector(".error").style.display = "none";
    }

    }

    searchBtn.addEventListener("click", ()=>{
        checkWeather(searchBox.value);
    })
</script>

</body>
</html>

```

## CSS CODE:

```

body {
    background-image: url('bg.jpg');
}

.card{
    width: 90%;
    max-width: 470px;
    background: linear-gradient(135deg, #00feba, #5b548a);
    color: #fff;
    margin: 100px auto 0;
    border-radius: 20px;
    padding: 40px 35px;
    text-align: center;
}

.search {
    width: 100%;
    display: flex;
    align-items: center;
    justify-content: space-between;
}

.search input {
    border: 0;

```

```
    outline: 0;
    background: #ebfffc;
    color: #555;
    padding: 10px 25px;
    height: 60px;
    border-radius: 30px;
    flex: 1;
    margin-right: 16px;
    font-size: 18px;
}

.search button{
    border: 0;
    outline: 0;
    background: #ebfffc;
    border-radius: 50%;
    width: 60px;
    height: 60px;
    cursor: pointer;
}

.search button img{
    width: 16px;
}

.weather-icon{
    width: 170px;
    margin-top: 30px;
}

.weather h1{
    font-size: 80px;
    font-weight: 500;
}

.weather h2{
    font-size: 45px;
    font-weight: 400;
    margin-top: -10px;
}

.details{
    display: flex;
    align-items: center;
    justify-content: space-between;
    padding: 0 20px;
    margin-top: 50px;
}

.col{
    display: flex;
    align-items: center;
    text-align: left;
```

```

}
.col img{
  width: 40px;
  margin-right: 10px;
}
.humidity, .wind{
  font-size: 28px;
  margin-top: -6px;
}
}
.weather{
  display: none;
}
.error{
  text-align: left;
  margin-left: 10px;
  font-size: 14px;
  margin-top: 10px;
  display: none;
}
}

```

## OUTPUT:

