

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Weather Dashboard</title>
    <style>
      @import url('https://fonts.googleapis.com/css2?
family=Open+Sans:wght@400;500;600;700&display=swap');
      * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
        font-family: 'Open Sans', sans-serif;
      }
      body {
        background: #E3F2FD;
      }
      h1 {
        background: #5372F0;
        font-size: 1.75rem;
        text-align: center;
        padding: 18px 0;
        color: #fff;
      }
      .container {
        display: flex;
        gap: 35px;
        padding: 30px;
      }
      .weather-input {
        width: 550px;
      }
      .weather-input input {
        height: 46px;
        width: 100%;
        outline: none;
        font-size: 1.07rem;
        padding: 0 17px;
        margin: 10px 0 20px 0;
        border-radius: 4px;
        border: 1px solid #ccc;
      }
      .weather-input input:focus {
        padding: 0 16px;
        border: 2px solid #5372F0;
      }
      .weather-input .separator {
        height: 1px;
        width: 100%;
        margin: 25px 0;
```

```

    background: #BBBBBB;
    display: flex;
    align-items: center;
    justify-content: center;
}
.weather-input .separator::before{
    content: "or";
    color: #6C757D;
    font-size: 1.18rem;
    padding: 0 15px;
    margin-top: -4px;
    background: #E3F2FD;
}
.weather-input button {
    width: 100%;
    padding: 10px 0;
    cursor: pointer;
    outline: none;
    border: none;
    border-radius: 4px;
    font-size: 1rem;
    color: #231d1d;
    background: #525e8d;
    transition: 0.2s ease;
}
.weather-input .search-btn:hover {
    background: #111215;
}
.weather-input .location-btn {
    background: #4a4b4b;
}
.weather-input .location-btn:hover {
    background: #5c636a;
}
.weather-data {
    width: 100%;
}
.weather-data .current-weather {
    color: #20b4e1;
    background: #34406e;
    border-radius: 5px;
    padding: 20px 70px 20px 20px;
    display: flex;
    justify-content: space-between;
}
.current-weather h2 {
    font-weight: 700;
    font-size: 1.7rem;
}
.weather-data h6 {
    margin-top: 12px;
    font-size: 1rem;
}

```

```

    font-weight: 500;
}
.current-weather .icon {
    text-align: center;
}
.current-weather .icon img {
    max-width: 120px;
    margin-top: -15px;
}
.current-weather .icon h6 {
    margin-top: -10px;
    text-transform: capitalize;
}
.days-forecast h2 {
    margin: 20px 0;
    font-size: 1.5rem;
}
.days-forecast .weather-cards {
    display: flex;
    gap: 20px;
}
.weather-cards .card {
    color: #fff;
    padding: 18px 16px;
    list-style: none;
    width: calc(100% / 5);
    background: #6C757D;
    border-radius: 5px;
}
.weather-cards .card h3 {
    font-size: 1.3rem;
    font-weight: 600;
}
.weather-cards .card img {
    max-width: 70px;
    margin: 5px 0 -12px 0;
}
@media (max-width: 1400px) {
    .weather-data .current-weather {
        padding: 20px;
    }
    .weather-cards {
        flex-wrap: wrap;
    }
    .weather-cards .card {
        width: calc(100% / 4 - 15px);
    }
}
@media (max-width: 1200px) {
    .weather-cards .card {
        width: calc(100% / 3 - 15px);
    }
}

```

```

    }
    @media (max-width: 950px) {
        .weather-input {
            width: 450px;
        }
        .weather-cards .card {
            width: calc(100% / 2 - 10px);
        }
    }
    @media (max-width: 750px) {
        h1 {
            font-size: 1.45rem;
            padding: 16px 0;
        }
        .container {
            flex-wrap: wrap;
            padding: 15px;
        }
        .weather-input {
            width: 100%;
        }
        .weather-data h2 {
            font-size: 1.35rem;
        }
    }
}
</style>
</head>
<body>
    <h1>Weather Dashboard</h1>
    <div class="container">
        <div class="weather-input">
            <h3>Enter a City Name</h3>
            <input class="city-input" type="text" placeholder="E.g., New
York, London, Tokyo">
            <button class="search-btn">Search</button>
            <div class="separator"></div>
            <button class="location-btn">Use Current Location</button>
        </div>
        <div class="weather-data">
            <div class="current-weather">
                <div class="details">
                    <h2>_____ ( _____ )</h2>
                    <h6>Temperature: ____°C</h6>
                    <h6>Wind: ____ M/S</h6>
                    <h6>Humidity: ____%</h6>
                </div>
            </div>
            <div class="days-forecast">
                <h2>5-Day Forecast</h2>
                <ul class="weather-cards">
                    <!-- Weather forecast cards go here -->
                </ul>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>
</div>

<script>
    const cityInput = document.querySelector(".city-input");
    const searchButton = document.querySelector(".search-btn");
    const locationButton = document.querySelector(".location-btn");
    const currentWeatherDiv = document.querySelector(".current-
weather");
    const weatherCardsDiv = document.querySelector(".weather-
cards");
    const API_KEY = "1d9f78e30b2044005ce9938e00c57fb2"; // Use your
actual OpenWeatherMap API key

    const createWeatherCard = (cityName, weatherItem, index) => {
        if (index === 0) { // Main weather card
            return `
                <div class="details">
                    <h2>${cityName} (${weatherItem.dt_txt.split("
") [0]})</h2>
                    <h6>Temperature: ${weatherItem.main.temp -
273.15).toFixed(2)}°C</h6>
                    <h6>Wind: ${weatherItem.wind.speed} M/S</h6>
                    <h6>Humidity: ${weatherItem.main.humidity}%</h6>
                </div>
                <div class="icon">
                    
                    <h6>${weatherItem.weather[0].description}</h6>
                </div>
            `;
        } else { // Forecast cards
            return `
                <li class="card">
                    <h3>(${weatherItem.dt_txt.split(" ") [0]})</h3>
                    
                    <h6>Temp: ${weatherItem.main.temp -
273.15).toFixed(2)}°C</h6>
                    <h6>Wind: ${weatherItem.wind.speed} M/S</h6>
                    <h6>Humidity: ${weatherItem.main.humidity}%</h6>
                </li>
            `;
        }
    };

    const getWeatherDetails = (cityName, latitude, longitude) => {
        const WEATHER_API_URL =
`https://api.openweathermap.org/data/2.5/forecast?lat=${
latitude}&lon=${longitude}&appid=${API_KEY}`;

```

```

        fetch(WEATHER_API_URL)
        .then(response => response.json())
        .then(data => {
            const uniqueForecastDays = [];
            const fiveDaysForecast = data.list.filter(forecast => {
                const forecastDate = new
Date(forecast.dt_txt).getDate();
                if (!uniqueForecastDays.includes(forecastDate)) {
                    return uniqueForecastDays.push(forecastDate);
                }
            });

            cityInput.value = "";
            currentWeatherDiv.innerHTML = "";
            weatherCardsDiv.innerHTML = "";

            fiveDaysForecast.forEach((weatherItem, index) => {
                const html = createWeatherCard(cityName, weatherItem,
index);

                if (index === 0) {
                    currentWeatherDiv.insertAdjacentHTML("beforeend",
html);

                } else {
                    weatherCardsDiv.insertAdjacentHTML("beforeend", html);
                }
            });
        })
        .catch(() => {
            alert("An error occurred while fetching the weather
forecast!");
        });
    };

    const getCityCoordinates = () => {
        const cityName = cityInput.value.trim();
        if (cityName === "") return;

        const API_URL =
`https://api.openweathermap.org/geo/1.0/direct?q=${
cityName}&limit=1&appid=${API_KEY}`;

        fetch(API_URL)
        .then(response => response.json())
        .then(data => {
            if (!data.length) return alert(`No coordinates found for $
{cityName}`);
            const { lat, lon, name } = data[0];
            getWeatherDetails(name, lat, lon);
        })
        .catch(() => {
            alert("An error occurred while fetching the
coordinates!");
        });
    };

```

```

    });
};

const getUserCoordinates = () => {
  navigator.geolocation.getCurrentPosition(
    position => {
      const { latitude, longitude } = position.coords;
      const API_URL =
`https://api.openweathermap.org/geo/1.0/reverse?lat=${latitude}&lon=${
longitude}&limit=1&appid=${API_KEY}`;
      fetch(API_URL)
        .then(response => response.json())
        .then(data => {
          const { name } = data[0];
          getWeatherDetails(name, latitude, longitude);
        })
        .catch(() => {
          alert("An error occurred while fetching the city
name!");
        });
    },
    error => {
      if (error.code === error.PERMISSION_DENIED) {
        alert("Geolocation request denied. Please reset location
permission to grant access again.");
      } else {
        alert("Geolocation request error. Please reset location
permission.");
      }
    }
  );
};

locationButton.addEventListener("click", getUserCoordinates);
searchButton.addEventListener("click", getCityCoordinates);
cityInput.addEventListener("keyup", e => e.key === "Enter" &&
getCityCoordinates());
</script>
</body>
</html>

```

OUTPUT:-

Weather Dashboard

Enter a City Name

E.g., New York, London, Tokyo

Use Current Location

Pune (2025-02-02)
Temperature: 32.00°C
Wind: 0.96 M/S
Humidity: 22%

 Overcast Clouds

Temperature: 32.00°C



5-Day Forecast

(2025-02-03)



Temp: 21.18°C

Wind: 0.46 M/S

Humidity: 40%

(2025-02-04)



Temp: 20.31°C

Wind: 1.18 M/S

Humidity: 52%

(2025-02-05)



Temp: 19.66°C

Wind: 0.73 M/S

Humidity: 53%

(2025-02-06)



Temp: 20.62°C

Wind: 0.86 M/S

Humidity: 34%

(2025-02-07)



Temp: 20.91°C

Wind: 0.58 M/S

Humidity: 35%