

## DOCUMENTACIÓN DE API EXTERNA ESCOGIDA PARA LA OBTENCIÓN DEL CLIMA

### API Externa sobre clima: OpenWeather

Link: <https://openweathermap.org/>

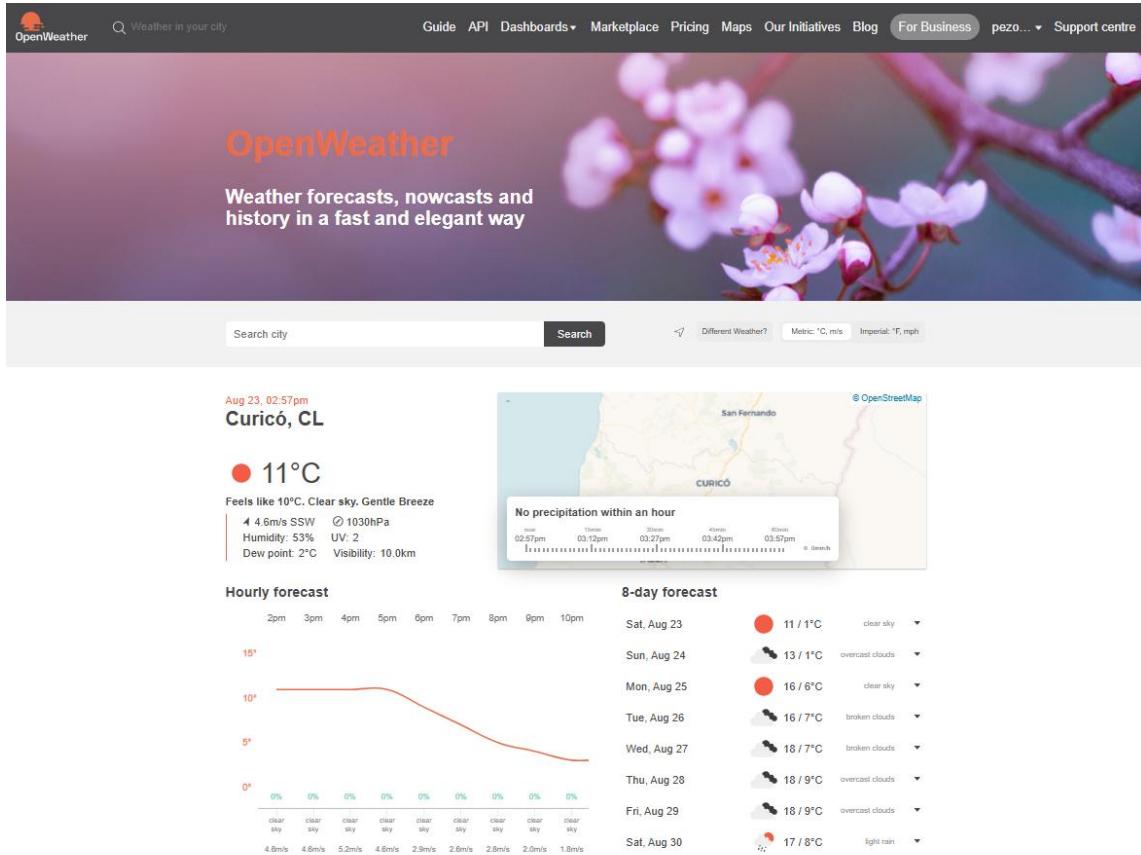


Imagen de la página de OpenWeather

OpenWeather funciona con peticiones HTTP, por lo que, si pegamos una URL completa en el navegador, nos devuelve un JSON en crudo.

Por ejemplo, si buscamos el clima de Curicó, Chile necesitamos usar una **API key** junto a la siguiente URL:

#### 1. API key de prueba

251d4c7019173fafefc8e94ec71ebf9b

#### 2. URL

<https://api.openweathermap.org/data/2.5/weather?q=Curico,CL&appid=251d4c7019173fafefc8e94ec71ebf9b&lang=es&units=metric>

## Parámetros importantes de la URL

- q=Curico,CL → ciudad y país en formato ISO.
- appid=... → la API key.
- lang=es → idioma de la respuesta, en este caso la es en español.
- units=metric → temperaturas en °C.

Programa de ejemplo en java, que imprime el JSON en crudo.



```
1 import java.io.BufferedReader;
2 import java.io.InputStreamReader;
3 import java.net.HttpURLConnection;
4 import java.net.URL;
5
6 public class SolicitarClima {
7     public static void main(String[] args) {
8         // API Key de OpenWeather
9         String apiKey = "251d4c7019173fafefc8e94ec71ebf9b";
10        String nombreCiudad = "Curico,CL"; // Ciudad que se quiere revisar
11
12        String urlString = "https://api.openweathermap.org/data/2.5/weather?q="
13            + nombreCiudad + "&appid=" + apiKey + "&lang=es&units=metric";
14
15        try {
16            URL url = new URL(urlString);
17            HttpURLConnection conn = (HttpURLConnection) url.openConnection();
18            conn.setRequestMethod("GET");
19            conn.connect();
20
21            // Verifica si la conexión fue exitosa
22            int codigoRespuesta = conn.getResponseCode();
23            if (codigoRespuesta != 200) {
24                throw new RuntimeException("HttpResponseCode: " + codigoRespuesta);
25            } else {
26                StringBuilder inline = new StringBuilder();
27                try (BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()))) {
28                    String linea;
29                    while ((linea = br.readLine()) != null) {
30                        inline.append(linea);
31                    }
32                }
33                // Muestra la respuesta de la API (JSON)
34                System.out.println(inline.toString());
35            }
36
37        } catch (Exception e) {
38            e.printStackTrace();
39        }
40    }
41 }
42 }
```

Código del programa.

```
[Running] cd "c:\Users\Diego\Desktop\" && javac SolicitarClima.java && java SolicitarClima
Note: SolicitarClima.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
{"coord":{"lon":-71.2333,"lat":-34.9833},"weather":[{"id":800,"main":"Clear","description":"cielo claro","icon":"01d"}],
"base":"stations","main":{"temp":13.06,"feels_like":11.57,"temp_min":13.06,"temp_max":13.06,"pressure":1030,"humidity":44,
"sea_level":1030,"grnd_level":998},"visibility":10000,"wind":{"speed":3.6,"deg":170},"clouds":{"all":0},"dt":1755976099,"sys": {"type":1,"id":8497,"country":"CL","sunrise":1755947744,"sunset":1755987581},"timezone":-14400,"id":3892870,"name":"Curicó",
"cod":200}

[Done] exited with code=0 in 1.739 seconds
```

*Salida por consola, es lo mismo que se mostraría en el navegador al usar el URL anteriormente explicado.*