

Programación multimedia y dispositivos móviles

Actividad 4.2. Predicción del
tiempo con Dart

Francisco José García Cutillas | 2FPGS_DAM



Índice

Predicción del tiempo con una app creada en Dart.....	3
---	---

Predicción del tiempo con una app creada en Dart

Se pide crear una aplicación Dart para consultar información meteorológica a través de la API pública que ofrece la Agencia Estatal de Meteorología (AEMET), mediante su portal [OpenData](#). Para ello, deberemos seguir los pasos siguientes:

1. Crear una clase *PrevisionMeteorologica* con estas propiedades:

- municipio: el nombre del municipio.
- codigo: el código del municipio usado por AEMET.
- precipitacion: la probabilidad de precipitación de hoy.
- tMax: la temperatura máxima del día.
- tMin: la temperatura mínima del día.
- estado: el estado del cielo (nuboso, soleado, etc.).

2. Acceder a la API OpenData de la AEMET y obtener los datos para un código de población determinado.

En el adjunto puedes descargar (y luego descomprimir) el **fichero municipios_por_provincia.json**, con información sobre los diferentes municipios de España: su nombre, código, longitud y latitud. De momento, únicamente nos interesará el código.

SOLUCION (Sólo el primer paso para ayudaos)

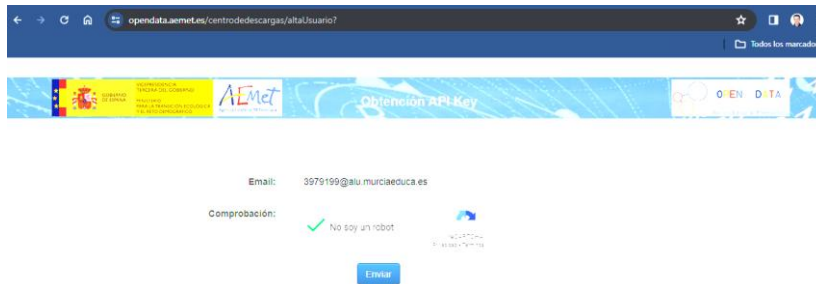
En primer lugar, debemos obtener una API Key personal para acceder a los servicios OpenData de la AEMET. Para ello, accedemos a la web <https://opendata.aemet.es/centrodedescargas/inicio> y buscamos la opción *Obtención de API Key*. Seguiremos los pasos indicados para obtener nuestra API Key, y, una vez la tengamos, podremos continuar.

Para consultar datos, la AEMET ofrece una extensa API cuya especificación podemos consultar en: <https://opendata.aemet.es/dist/index.html>.

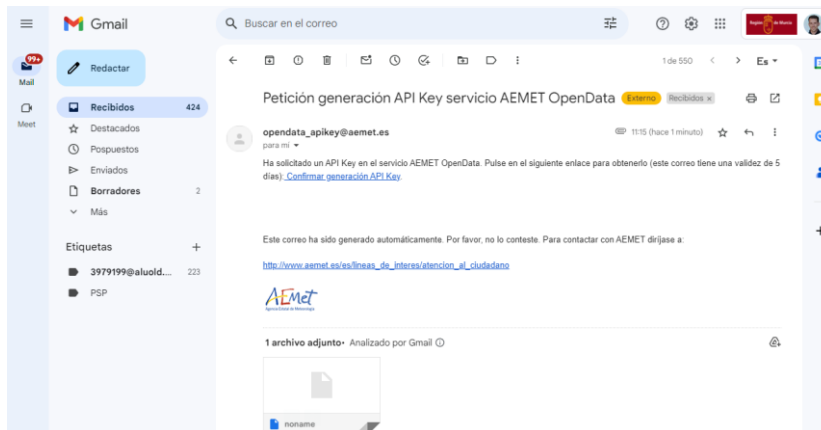
Concretamente, vamos a hacer una llamada GET para obtener la previsión meteorológica del día. La petición que hay que hacer es la siguiente:

```
/api/prediccion/especifica/municipio/diaria/{id_municipio}
```

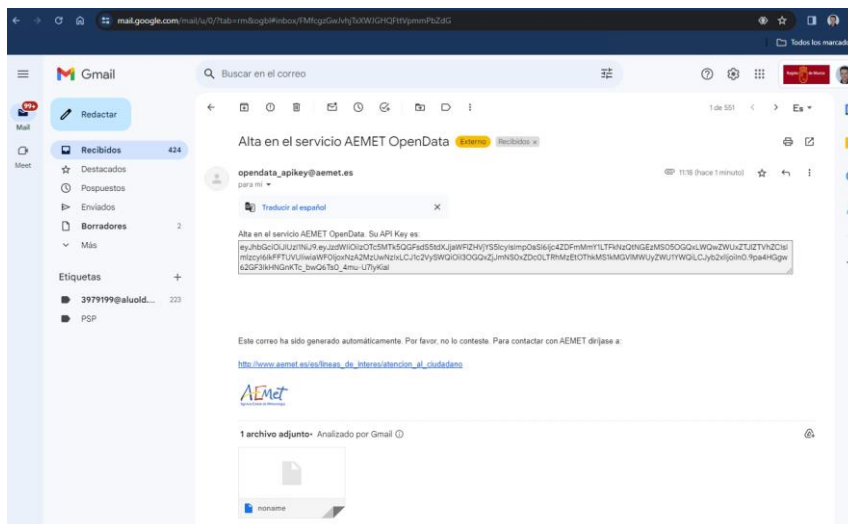
Comenzamos con la obtención de la API key de la AEMET



Una vez que recibimos el correo, pulsamos sobre confirmar generación de API key.



Ésta será la API key.



Para crear un proyecto nuevo, nos situaremos en el directorio en el que lo queremos crear y escribimos el siguiente comando.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema4\CasoPractico4_2> dart create -t console-simple prediccion_tiempo
Creating prediccion_tiempo using template console-simple...

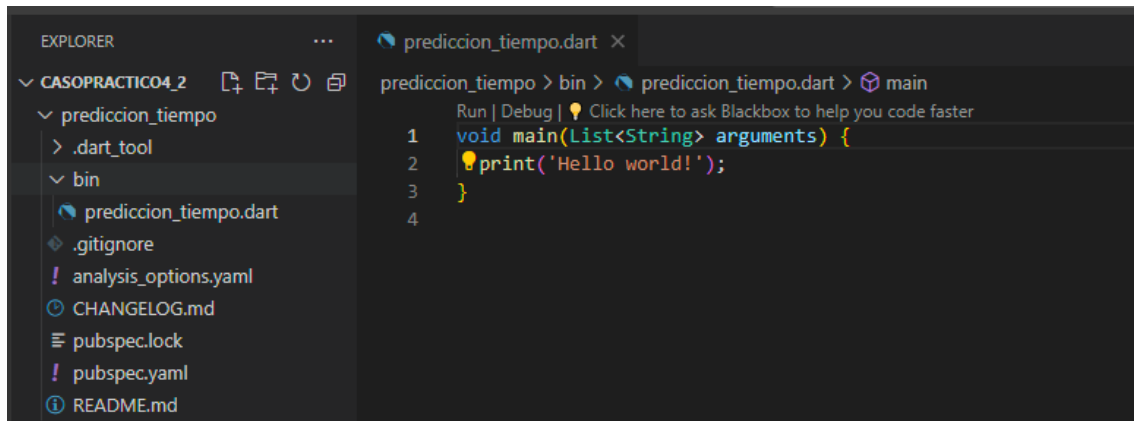
.gitignore
analysis_options.yaml
CHANGELOG.md
pubspec.yaml
README.md
bin\prediccion_tiempo.dart

Running pub get...
Resolving dependencies...
Changed 1 dependency!
1 package has newer versions incompatible with dependency constraints.
Try 'dart pub outdated' for more information.

Created project prediccion_tiempo in prediccion_tiempo! In order to get started, run the following commands:

cd prediccion_tiempo
dart run
```

Con esto, nos genera el esqueleto de un proyecto en lenguaje Dart con el siguiente contenido. En la carpeta bin se encuentra el main del proyecto.



The screenshot shows the Visual Studio interface. On the left, the 'EXPLORER' pane displays the project structure for 'CASOPRACTICO4_2', including a subfolder 'prediccion_tiempo' which contains files like '.dart_tool', 'bin', 'prediccion_tiempo.dart', '.gitignore', 'analysis_options.yaml', 'CHANGELOG.md', 'pubspec.lock', 'pubspec.yaml', and 'README.md'. The 'bin' folder is expanded, showing 'prediccion_tiempo.dart'. On the right, the editor shows the content of 'prediccion_tiempo.dart', which is a simple Dart program with a 'main' function that prints 'Hello world!'.

Antes de nada, necesitamos la librería http para poder realizar conexiones a un servidor. Por lo tanto, nos vamos a ir al fichero pubspec.yaml y vamos a añadir la dependencia.

```
9  # Add regular dependencies here.
10 dependencies:
11   | Search # path in Dart Packages
12   | # path: ^1.8.0
13   | Search http in Dart Packages
14   | http: ^0.13.4
```

Una vez añadida, ejecutamos en el terminal de Visual Studio lo siguiente.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema4\CasoPractico4_2\prediccion_tiempo> dart pub get
Resolving dependencies...
http 0.13.6 (1.2.0 available)
lints 2.1.1 (3.0.0 available)
Got dependencies!
2 packages have newer versions incompatible with dependency constraints.
Try 'dart pub outdated' for more information.
```

Podemos observar que nos ofrece actualizaciones de dicha librería y de otra más que ya estaba por defecto. Las actualizamos.

```
# Add regular dependencies here.
```

```
dependencies:
```

```
  Search # path in Dart Packages
```

```
  # path: ^1.8.0
```

```
  Search http in Dart Packages
```

```
  http: ^1.2.0
```

```
dev_dependencies:
```

```
  Search lints in Dart Packages
```

```
  lints: ^3.0.0
```

Volvemos a ejecutar el comando “dart pub get” y ya no nos da el problema.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema4\CasoPractico4_2\prediccion_tiempo> dart pub get
Resolving dependencies...
Got dependencies!
```

Código de la aplicación.

```
prediccion_tiempo > bin > ! pubspec.yaml
prediccion_tiempo > bin > prediccion_tiempo.dart > PrevisionMeteorologica > mostrarPrevision
Click here to ask Blackbox to help you code faster
import 'package:http/http.dart' as http;
import 'dart:convert';

//API key proporcionada por la AEMET para consultar el tiempo
String apiKey =
    "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiIzOTc5MTk5QGFsdS5tdXJjaWFlZHVjYSSlcyIsImp0eSI6Ijc4ZDZmMmY1LTFFKwZQtNGEzMSo5OGQxLmVudC0iLCJpcyIsImNpdCI6ImFkbG8iLCJ1bmlrZXkiOiJkaXYuInRhdia...";

//Clase PrevisionMeteorologica {
String municipio = "";
String codigo = "";
int precipitacion = 0;
int tMax = 0;
int tMin = 0;
String estado = "";

/*Constructor de la clase. Recibe nombre del municipio de tipo String y su código de tipo int */
PrevisionMeteorologica(this.municipio, this.codigo);

/*Método obtenerPrevision encargado de realizar la petición a la API de la AEMET y mostrar por terminal la misma */
obtenerPrevision() async {
//URL que nos proporciona la AEMET para realizar la consulta
String url =
    "https://opendata.aemet.es/opendata/api/prediccion/especifica/municipio/diaria/$codigo?api_key=$apiKey";

//Realizamos la petición al servidor de la AEMET
var respuestaServidor = await http.get(Uri.parse(url));

//Cuerpo para el JSON a generar
Map cuerpoJSON = {};

//Si el servidor nos responde con el código 200 significa que la respuesta ha tenido éxito
if (respuestaServidor.statusCode == 200) {
    cuerpoJSON = jsonDecode(respuestaServidor.body) as Map;
}

//Petición para que nos devuelva los datos
var respuestaDatos = await http.get(Uri.parse(cuerpoJSON["datos"]));

//Si vuelve a haber respuesta válida, extraemos el json y lo guardamos en una lista
var infoTiempo = [];
if (respuestaDatos.statusCode == 200) {
    infoTiempo = jsonDecode(respuestaDatos.body) as List;

    //Información de precipitaciones
    precipitacion = infoTiempo[0]["prediccion"]["dia"][0]["probPrecipitacion"]
        [0]["value"];

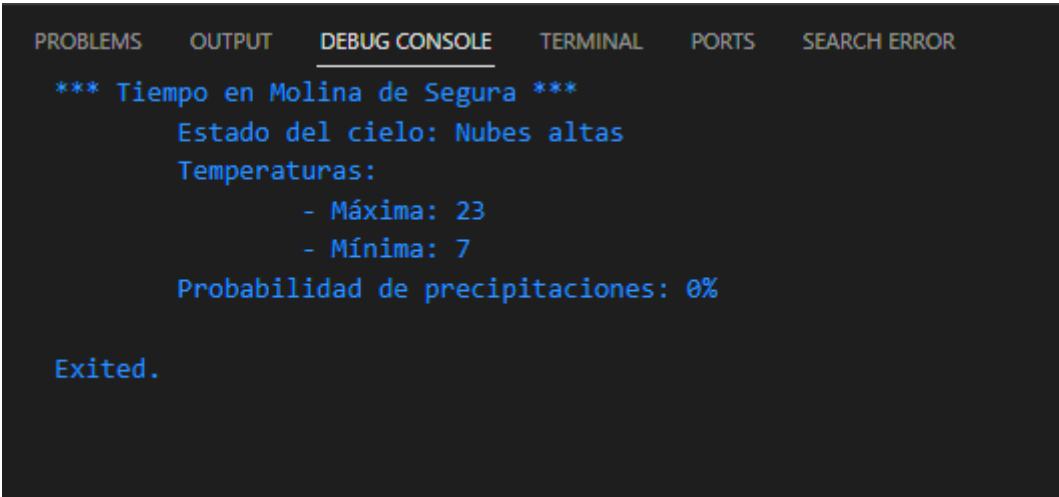
    //Temperaturas máximas y mínimas
    tMax = infoTiempo[0]["prediccion"]["dia"][0]["temperatura"]["maxima"];
    tMin = infoTiempo[0]["prediccion"]["dia"][0]["temperatura"]["minima"];

    //Estado del cielo.
    //Se hace con un while porque está dividido en 7 franjas horarias y no siempre hay datos
    //Por ello recorreremos el bucle hasta encontrar algún dato
    int i = 0;
    while (estado == "" && i < 7) {
        estado = infoTiempo[0]["prediccion"]["dia"][0]["estadoCielo"][i]
            ["descripcion"];
        i++;
    }
}

//Método mostrarPrevision encargado de mostrar por terminal la previsión del tiempo
void mostrarPrevision() {
print("*** Tiempo en $municipio ***");
print("\tEstado del cielo: $estado");
print("\tTemperaturas:");
print("\t\t- Máxima: $tMax");
print("\t\t- Mínima: $tMin");
print("\tProbabilidad de precipitaciones: $precipitacion%");
}

//Método main para la ejecución de programa
Run | Debug
void main(List<String> arguments) async {
    PrevisionMeteorologica prevision =
        PrevisionMeteorologica("Molina de Segura", "30027");
    await prevision.obtenerPrevision();
    prevision.mostrarPrevision();
}
```

Salida por terminal.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

*** Tiempo en Molina de Segura ***
  Estado del cielo: Nubes altas
  Temperaturas:
    - Máxima: 23
    - Mínima: 7
  Probabilidad de precipitaciones: 0%

Exited.
```