

## Documentación Weather.

El Proyecto Weather es una aplicación java desarrollada para obtener información del clima en una ciudad determinada perteneciente a un determinado grupo.

La aplicación usa para su funcionamiento la Api publica de AccuWeather.

Dicha api cuenta con un servicio que retorna 50 ciudades del mundo.

Weather tomara esas 50 ciudades y las almacenara en una base de datos, pero solo el nombre de la ciudad y la key que es un identificador único de la ciudad.

Esa información almacenada brinda la posibilidad de realizar nuevas consultas a la api con la key de la ciudad permitiendo obtener daos específicos.

Weather utiliza esa información para obtener la temperatura del día actual para esa ciudad.

## Estructura

El proyecto está centrado en 2 ramas, lo relacionado a las ciudades y lo relacionado al clima, por lo tanto, cuenta con 2 controllers CitiesControllerImpl Y WeatherControllerImpl.

Cada controlador tendrá las entradas a los servicios de cada rama. En la capa de servicios existen 3 servicios principales: CitiesServiceImpl, WeatherServiceImpl y AccuWeatherApi, Este último será quien se conecte con la api de AccuWeather y retorne a los servicios de cities y weather la información necesaria.

El proyecto también cuenta con una capa Dao, CitiesDaoImpl, la cual es la encargada de conectarse a la base de datos para almacenar las ciudades y posteriormente obtener información desde la DB sin necesidad de consultar a la api.

También cuenta con capas de testeo, que a través de Junit, evalúa ciertas partes del código a la hora de compilar.

## Funcionalidades y características

- Carga de datos en la base de datos.
- Limpieza de tabla de base de datos.
- Obtener información de clima de una ciudad determinada.

## Documentación de código

El código fuente cuenta con documentación para que la lectura del mismo sea más sencilla, los métodos y clases más importantes tienen una descripción e información útil para la comprensión.

## Instalación

Para llevar a cabo la instalación del proyecto y obtener la colección de postman para realizar pruebas manuales se deberá leer el archivo README que se encuentra en /weather/README.md.

## Modo de uso

El proyecto cuenta con 3 servicios:

- **loadCities:** Este servicio no requiere parámetros, se encarga de realizar la petición a la api solicitando 50 de ciudades del mundo. Almacenara en la base de datos el nombre y la key de cada ciudad. En caso de éxito retornara un arreglo con los nombres de las 50 ciudades insertadas en la base de datos.
- **cleanTable:** Este servicio existe solo a fines practico para poder eliminar los datos de la tabla cities y poder realizar pruebas.
- **getTodayWeather:** Este servicio se encarga de devolver las temperaturas máxima y mínima del día actual de una ciudad determinada. Esta ciudad deberá ser enviada en la solicitud por parámetro. El servicio como primera instancia evaluara que la ciudad elegida pertenezca al grupo de 50 ciudades haciendo una consulta a la base de datos, en caso exitoso obtendrá de la base de datos la key de esa ciudad y con ese dato llamara a un nuevo servicio de la api el cual devolverá la información deseada para este servicio.

## Api de AccuWeather

Para conectarse con la api es necesario generar una key que permitirá tener acceso a los endpoints brindados, para este proyecto la key es "TGsyg1B72GgjycfMtnviPQGhRn4Kggqr".

Esta key se encuentra almacenada en el properties de la aplicación para poder realizar las consultas a la api.

Para agregar una nueva consulta a la api se deberá copiar la url del servicio deseado desde la página de AccuWeather (<https://developer.accuweather.com/apis>) y al final de la misma se agregará dicha key como parámetro bajo el nombre de apiKey.

Ejemplo:

Para el caso del servicio que obtiene las 50 ciudades esta es la url que forma:

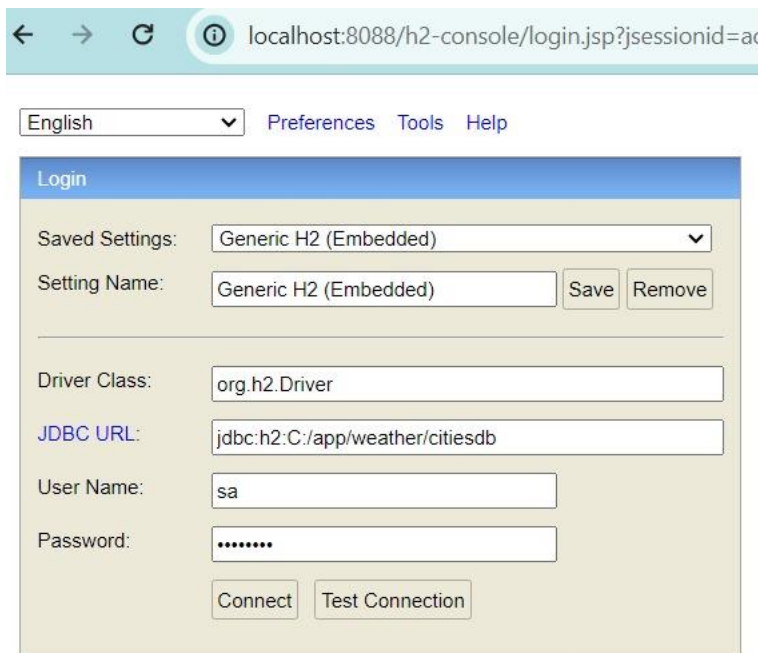
`https://dataservice.accuweather.com/locations/v1/topcities/50?apikey=TGsyg1B72GgjycfMtnviPQGhRn4Kggqr`

## Base de datos

Para ingresar al panel de control de la base de datos se deberá ingresar a <http://localhost:8088/h2-console/> una vez que el proyecto esté en ejecución.

Ahí los datos de ingreso se cargarán automáticamente a excepción de la pass la cual es “password” (por fines prácticos).

Agrego la siguiente imagen para visualizar el ingreso a la base de datos.



## FAQ

¿Qué ocurre si ejecuto loadCities más de una vez seguidas?

Cada ejecución de loadCities inserta 50 ciudades en la base de datos, y son las mismas 50 ciudades. Como el fin de este proyecto es demostrar ciertas habilidades de programación se decidió dejar que eso ocurra, para evitar que al querer obtener la key de una ciudad repetida se rompa la ejecución, se agregó “top 1” a la query que obtiene dicho dato, de esta manera me aseguro de que tome solo uno de los datos.

¿Es lo mismo buscar London que london?

Si, el getTodatWeather no es case sensitive, por lo tanto no importa si el nombre tiene o no mayúsculas, el método transformara todo a lowerCase para realizar la búsqueda.