



Enunciado

Deseamos desarrollar una aplicación móvil utilizando Alloy y titanium SDK 9.3.0 que recupere de un servicio API REST la lista de las temperaturas de ciudades de España y las muestre en pantalla con una apariencia similar a la siguiente:



Ejercicio 1. Obtención y guardado de los datos en Base de datos (10 puntos)

Los datos se obtienen mediante una llamada HTTP a un servicio REST

https://www.upv.es/pls/dorest/sic_app_test.get_grados

La estructura de los datos devueltos es un vector JSON de la forma

```
[ { "codCiudad": TEXT, "ciudad": TEXT, "temperatura": INTEGER}, ...]
```

Ejemplo de un registro

```
{ "codCiudad": "V", "ciudad": "Valencia", "temperatura": 17}
```

Utilizando el fichero

- `app/controllers/index.js`

En este ejercicio la APP deberá tras arrancarse la aplicación invocar el servicio API REST y guardar los datos en una colección de Alloy.

Para la llamada al API REST se puede utilizar el método `Ti.Network.createHttpClient`



Ejemplo de uso

```
var url;  
  
var client = Ti.Network.createHTTPClient({  
    onload : function(e) {  
        },  
    onerror : function(e) {  
        },  
    timeout : 5000 // in milliseconds  
});  
  
client.open("GET", url);  
client.send();
```

El modelo Alloy donde se guardarán los datos deberá seguir el siguiente esquema:

```
exports.definition = {  
    config: {  
        columns: { .... },  
        adapter: {  
            db_name:  
            type:  
            collection_name:  
            idAttribute:  
        }  
    }  
};
```



Ejercicio 2. Visualización de los datos (10 puntos)

Utilizando al menos los ficheros

- app/views/index.xml
- app/styles/index.tss

Mostraremos los datos en pantalla utilizando la colección creada en el ejercicio 1 y dataBinding.

La información mostrada deberá coincidir con el prototipo mostrado en el enunciado del ejercicio.



Ejercicio 3. Recuperar los datos (10 puntos)

Se deberá modificar el controlador `app/controllers/Index.js` para que realice las siguientes acciones:

Recuperar los datos de la colección y hacer las siguientes transformaciones

- El valor de las temperatura es numérico pero deberá mostrarse en pantalla añadiendo el símbolo de grados Celsius, °C
- Los datos aparecerán ordenados por temperatura de mayor a menor y en caso de ser la misma por el nombre de la ciudad