Examen practico de servicio REST

**Instrucciones:**

1.- Crear una carpeta en donde clonaremos el repositorio de GitHub.

2.- Abrir la consola de comandos en la carpeta que creamos y utilizer el siguiente comando **git clone** [**https://github.com/patoacosta/ExamenSoftura.git**](https://github.com/patoacosta/ExamenSoftura.git)

3.- En PostgreSQL ejecutar cada Script en el siguiente orden:

* CreateDB\_Script.txt
* CreateSecuence\_Script.txt
* CreateTable\_m\_pelicula.txt
* CreateTable\_c\_horario.txt
* CreateTable\_r\_pelicula\_horario.txt

4.- Despues abriremos el Proyecto desde IntelliJ

5.- Despues de construir el Proyecto verificaremos que la parte de A**rchivo>Estructura del Proyecto>Proyecto SDK** este la **version 13** sino la descargaremos ahi mismo.

6.- Abriremos el archivo **…/pelicula/src/main/resources/application.properties** y le agregaremos lo siguiente para conectarnos a la base de datos que creamos:

server.port=9090  
spring.jpa.database=POSTGRESQL  
spring.sql.init.platform=postgres  
spring.datasource.url=jdbc:postgresql://localhost:5432/pelicula?currentSchema=public  
spring.datasource.username=postgres  
spring.datasource.password=root  
spring.jpa.hibernate.ddl-auto=validate

7.- Despues ejecutaremos el archivo **…/pelicula/src/main/java/com/example/pelicula/PeliculaApplication.java**

Para que se ejecute nuestro programa.

8. Despues en Google Chrome añadiremos la extencion **Talend API Tester**

Para poder hacer las peticiones

9.- Direcciones para las peticiones:

**GET – listar todos los elementos de la BD**

<http://localhost:9090/home/obtenerPelicula>

**POST – agregar un nuevo element**

<http://localhost:9090/home/guardarPelicula>

JSON con el formato para guardar el registro nuevo (JSON **Sin ID**)

{

"nombre\_pelicula":"Avengers : EndGame",

"imagenes\_pelicula":"URL",

"duracion\_pelicula":186,

"clasificacion\_pelicula":"A",

"status\_pelicula":true

}

**POST – actualizar un registro ya existente**

<http://localhost:9090/home/guardarPelicula>

JSON con el formato para actualizar un registro (incluimos el ID del registro a **modificar**)

{

"id\_pelicula":4,

"nombre\_pelicula":"Avengers : EndGame",

"imagenes\_pelicula":"http://URL",

"duracion\_pelicula":186,

"clasificacion\_pelicula":"A",

"status\_pelicula":true

}

**POST – eliminar un registro ya existente ()**

<http://localhost:9090/home/eliminarPelicula>

JSON con el formato para actualizar un registro (incluimos el ID del registro a **eliminar**)

{

"id\_pelicula":4

}