

Crea un proyecto en Maven con Eclipse (donde configuraremos Java 11 y las dependencias que vayas necesitando en el proyecto).

1. La primera opción del menú es la coincidencia en películas. Lo primero que queremos saber es en qué películas han coincidido dos identificadores que pediremos al usuario. Por ejemplo si pasamos 1 y 4 (Luke Skywalker y Darth Vader), nos dirá **el title (no el número ni la url, por lo que tendrás que acceder a la ruta para coger el título)** de las películas donde han actuado ambos al mismo tiempo. Un ejemplo de ejecución sería así (4 puntos):

NOTA: utiliza la función llamada `readUrl` que lee una url y devuelve el json de dicha url.

```
1.-Películas comunes
0.-Salir
Option:
1
Introduzca el código del primer actor:
1
Introduzca el código del segundo actor:
4
The Empire Strikes Back
Revenge of the Sith
Return of the Jedi
A New Hope
1.-Películas comunes
0.-Salir
Option:
```

2. Añade la opción de serializar y deserializar los personajes que has ido comparando las películas, para ello, cada vez que vayas comparando personajes (Luke Skywalker y Darth Vader por ejemplo), si no los tienes sus objetos ya en una lista, guárdalos (no guardes repetidos). Luego da la opción tanto de serializar esa lista, como otra de deserializar y mostrar los resultados (2 puntos):

```
The Empire Strikes Back
Revenge of the Sith
Return of the Jedi
A New Hope
1.-Películas comunes
2.-Serializar actores
3.-Deserializar actores
0.-Salir
Option:
2
1.-Películas comunes
2.-Serializar actores
3.-Deserializar actores
0.-Salir
Option:
3
La lista tiene 2 elementos
People [name=Luke Skywalker, height=172, mass=77, hair_color=blond]
People [name=Darth Vader, height=202, mass=136, hair_color=none, si]
1.-Películas comunes
2.-Serializar actores
3.-Deserializar actores
0.-Salir
```

3. Tenemos un problema con el almacenamiento de los pasajeros que son capaces de transportar las naves (starships). Queremos saber que naves (starships) de una película, cuyo código pediremos por teclado pueden llevar más pasajeros que la del fichero Json que nos han pasado. Para ello, preguntaremos por consola el código de la película (no debemos leer todas las películas solamente la del código pasado) y sacaremos un listado indicando el nombre (name) y la capacidad (passengers) de las naves que aparecen en esa película (film) con mayor capacidad que el Json que tenemos (cr90.json). Crea una nueva opción del menú para obtener los resultados.
4. A partir del fichero Json (cr90.json) busca una utilidad por Internet que transforme el Json pasado a un Xml.
5. Crea otra opción para leer el XML creado en el apartado anterior guardándolo en una clase de Java, mostrando por pantalla toda la información usando el método `toString` de dicha clase.