SPARK JAVA - PARTE III SQL2o





El objetivo particular de la sección III de los trabajos prácticos referidos a Spark, es manejar la persistencia de objetos. SQL2o es una pequeña biblioteca, que hace que sea fácil convertir el resultado de las sentencias sql en objetos. Permite hacer un mapeo Objeto-Relacional (ORM del inglés *Object Relational Mapping*) de una manera simple. Los ORM permiten convertir los datos de los objetos en un formato correcto para poder guardar la información en una base de datos (mapeo) donde los datos que se encuentran en la aplicación, quedan vinculados a la base de datos (realizándose la persistencia de los datos que están en los objetos de la aplicación).



Herramientas / Material de Consulta

Para llevar a cabo los ejercicios se debe tener completo el Trabajo Práctico
Spark-Parte I, Parte II y tener el entorno de desarrollo operativo que se especificó en dicho trabajo práctico.

https://www.sql2o.org



Tenga en cuenta la siguiente Nomenclatura:

PK = Primary Key.

FK = Foreign Key.

Nota: Eligan los tipos mas adecuados.

MVC sin Velocity y SQL2o

Ejercicio 1: Un sistema de areas de una empresa y sus empleados. Implemente la BD e inicialícela con datos útiles.

AREA (codigo, nombre, director). PK codigo

EMPLEADO (nombre, categoria, dedicacion, codigo). PK nombre.

FK codigo -> AREA

- retornar todos las areas.
- retornar todos las áreas con director x
- buscar todos los empleados con dedicación YY
- buscar todas áreas que tengan empleados
- buscar todos los empleados que trabjajen en un area con codigo X
- buscar un director que también sea empleado

MVC con Velocity y SQL2o

Ejercicio 2: Usando como base el ejercicio 1, desarrollar los siguientes microservicios web con Velocity combinando arquitectura MVC y REST:

- Retornar todos las áreas
- Dada un área, conocer los empleados de la misma.
- Agregar un área.
- Listar todos los empleados.
- Buscar un empleado y ver a qué área pertenece.
- Cantidad de empleados por área.
- Agregar un empleado con el área al que pertenece. Solo un área.

Construya un vtl de forma cohesiva, no haga un vtl por cada resultado.

Ejercicio 3: Un Netflix básico, de consulta de películas. De cada película, necesitas su título, año, nacionalidad y algunos datos de su director: el nombre, la fecha de nacimiento y su país de origen. Además, saber su idioma, si es en blanco y negro o en color, si tiene alguna restricción de edad, un resumen. La ficha técnica de cada película también debe incluir el reparto de actores, donde aparecerá su nombre, su nacionalidad y el nombre del personaje que tiene en la película.

Implemente la BD e inicialícela con datos útiles.

DIRECTOR (nombre, nacion, fnac).

PK nombre

PELICULA (titulo, año, nacion, idioma, color, resumen, nombre).

PK titulo.

FK nombre -> DIRECTOR

ACTOR (nombre, nacion).

PK nombre

REPARTO (titulo, nombre, personaje).

PK titulo, nombre.

FK titulo -> PELICULA.

FK nombre -> ACTOR

DESARROLLAR LOS SIGUIENTES MICROSERVICIOS WEB combinando arquitectura MVC y REST:

- Mostrar todas las películas que hay de un director dado.
- Listar las películas en las que trabaja un actor dado.
- La nacionalidad de los actores que un director ha dirigido.

- Ingresar una lista de usuarios en la BD predeterminada:
 - 1. Cada vez que se ve una película, registrar dicha película para ese usuario.
 - 2. Listar películas recomendadas, a partir de las ya vistas, por nación de la pelicula, actor, director, ... arme el algoritmo que le parezca mas adecuado.

Implemente los microservicios.

Ejercicio 4: Estilo Plataforma 10. Empresas de transporte desean crear una base de datos para almacenar información sobre sus rutas para que los usuarios puedan comprar.

Desarrolle el microservicio asumiendo la inicialización de la BD con datos existentes. Los microservicios en REST deben permitir mostrar las empresas que viajan desde un origen a un destino seleccionado por el usuario y el precio de dicho recorrido.

Ejercicio 5: Se desea construir un e-commmerce como PedidosYa u otro propuesto por usted.

Desarrolle el sistema donde muestre funcionalidades básicas