

**11<sup>a</sup>**  
Emisión

# DIPLOMADO Desarrollo de Sistemas con Tecnología Java

## Módulo 6

### Introducción de Aplicaciones Empresariales con Spring Framework.

*Mtro. ISC Miguel Ángel Sánchez Hernández*



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Dirección General de Cómputo y de Tecnologías de información y Comunicación

Dirección de Docencia en TIC



Educación  
Continua  
1971 - 2021

# Convenios

- Tolerancia de inicio de clases 15 minutos
- 20 minutos de receso
  - Viernes 18:30-18:50
  - Sábados 11:30-11:50

# Objetivo

Conocer como Spring Data maneja la persistencia, por medio JDBC Template, creando un CRUD y un Microservicio



# Lo que veremos

- Como maneja Spring Data la persistencia
- Cuando configurar y ocupar JDBC Template
- Ejemplo del uso de JDBC Template
- Ejemplo de CRUD con JDBC Template
- Ejemplo de Microservicio con JDBC en Spring Boot



# Persistencia en Spring

Ahora Spring ya cuenta con una familia de marcos de trabajo de acceso a datos que integran diversas tecnologías de acceso.

- Persistencia de datos por medio de JDBC directo.
- Mapeo del modelo de objetos al modelo relacional (ORM) (se vera en el siguiente módulo)



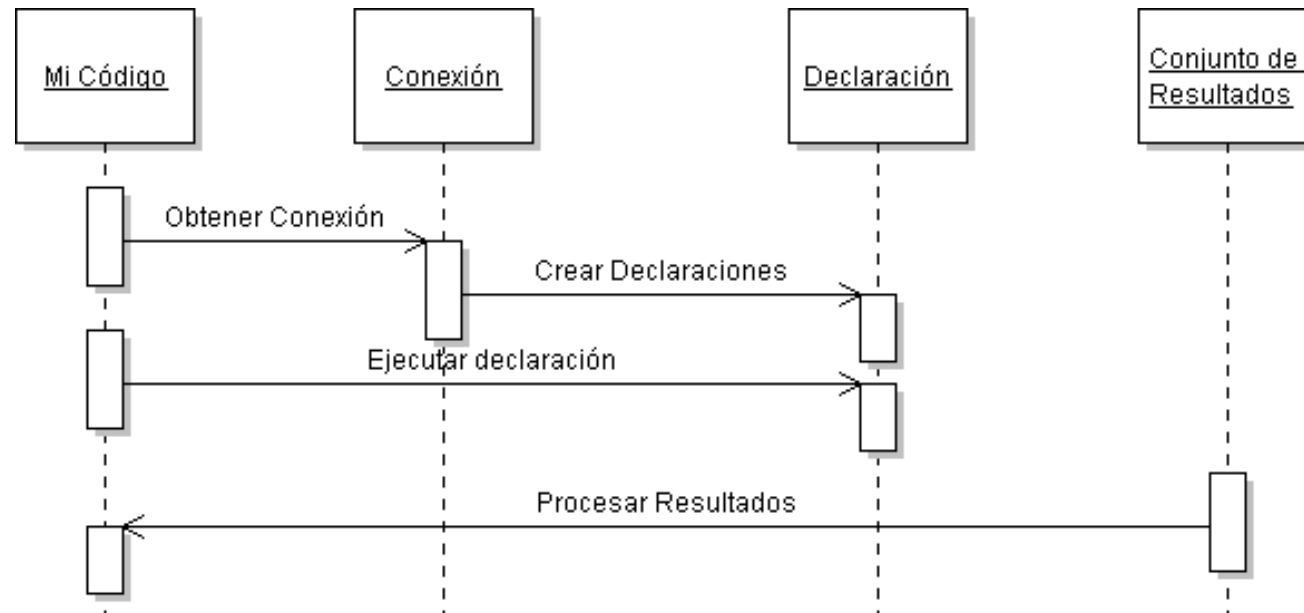
# ¿Cuándo se ocupa JDBC Template?

En estas situaciones se recomienda ocupar JDBC Template

- La estructura de la base de datos no se puede modificar y que no este normalizada, dando como resultado modelar las entidades de una manera compleja.
- Tablas sin llave primaria.
- Ocupar directamente SQL para aprovechar las funcionalidades que nos de la base de datos misma, logrando así un rendimiento optimo, por ejemplo uso constante de procedimientos almacenados.



# ¿Secuencia para la programación en JDBC?



- 1.- Cargar el Controlador JDBC
- 2.- Conectar con el DBMS
- 3.- Crear y Ejecutar la instrucción SQL
- 4.- Procesar los datos que devuelve DBMS
- 5.- Terminar la conexión con el DBMS



# Filosofía de acceso a datos en Spring Data

Para obtener una persistencia de datos real, ocuparemos una de las estrategias que ocupa como patrón de diseño Java EE, estamos hablando de el patrón DAO (Data Access Object, Objeto de Acceso a Datos).

La idea básica es encapsular la creación, lectura, actualización y eliminación conocidas como funciones CRUD (Create,Read,Update,Delete) para trabajar con objetos y base de datos.

Estas funcionalidades se hacen por medio de una interfaz, haciendo que el resto de la aplicación acceda por medio de ellas.



## Práctica Cuatro: JdbcTemplate en Spring Boot





# • Microservicio y JDBC con Spring Boot

Se dividen en varias practicas y cada una de ellas realiza lo siguiente:

- Práctica Cinco: Se arma los conceptos aprendidos de Microservicios y JDBC, para crear un Microservicio.
- Práctica Seis: Creación de un cliente en JavaScript que consuma el Microservicio.
- Práctica Siete: Una aplicación Web en Spring Boot accede también al Microservicio.



**Práctica Cinco: Microservicio y JDBC en Spring Boot**



**Práctica Seis: Cliente JavaScript**



**Práctica Siete: Aplicación Web consume Microservicio**

