

## EJERCICIO TÉCNICO DESARROLLADOR WEB JAVA L1

Metaphorce está en un proceso para el desarrollo de una solución de software en inglés de Recursos Humanos y está solicitando un primer entregable que tenga como alcance el desarrollo de **5 servicios web** para la gestión de sus empleados comenzando por sus datos básicos y tipos de contratos de tal forma que posteriormente una aplicación web o aplicación móvil pueda realizar las operaciones con estos catálogos.

Realizar el diseño de una base de datos llamada **MetaphorceBd**, su implementación y la API correspondiente que considere las operaciones indicadas en el punto 4 del requerimiento

### Catálogo de Empleados:

Nombre	Tipo de dato	Extra	Definición
Id Empleado	Entero	Auto incrementable	Folio del empleado único
Nombre Empleado	Texto	No permite nulo	Nombre o nombres del empleado
Apellidos Empleado	Texto	No permite nulo	Apellido paterno y/o materno del empleado
Fecha de nacimiento	Fecha sin hora	No permite nulo Formato yyyy-mm-dd	Fecha de nacimiento
Correo	Texto	No permite nulo	Correo personal del empleado
Celular	Texto	No permite nulo Formato (99) 99 99-99-99	Celular personal del empleado
Tipo de contrato	Entero	No permite nulo Valor foraneo	Tipo de contrato que tiene actualmente
Está activo (si / no)	Booleano	No permite nulo	Indica si está dado de baja o sigue activo en el sistema

### Catálogo de Tipo de Contratos:

Nombre	Tipo de dato	Extra	Definición
Id Tipo de contrato	Entero	Auto incrementable	Folio único del tipo de contrato
Nombre	Texto	No permite nulo	Nombre del tipo de contrato
Descripción	Texto	Permite nulo	Descripción del tipo de contrato

- Ambos catálogos están relacionados a través del tipo de contrato.
- El catálogo de tipos de contratos deberá tener los siguientes 3 registros precargados: Permanent, Fixed-Term, External.

## REQUERIMIENTO

1. Utilizar java 1.8 de preferencia.
2. Crear una aplicación web en **Eclipse / Netbeans / IntelliJ** o similar, utilizando **Java con Spring Framework / Spring Boot** para generar la **API** correspondiente al requerimiento expuesto.
3. Utilizar **Hibernate** para crear la conexión entre base de datos, mapeo del schema y las operaciones hacia cualquiera de los siguientes motores: MySQL u Oracle o PostgreSQL (a su elección, elegir solo uno).
4. Generar **únicamente** los siguientes 5 endpoints:
  - 1) Obtener listado de empleados activos en el sistema.
  - 2) Obtener la información de un empleado basado en su folio.
  - 3) Agregar un nuevo empleado.
    - i. Debe validar que el correo electrónico no exista, en caso de existir deberá mandar un error indicando que ya existe el registro.
  - 4) Actualizar un empleado existente a través de su folio.
    - i. Si el correo se modifica, debe validar previamente que el nuevo correo no exista en algún otro registro y en caso de que si mandar error.
  - 5) Eliminar un empleado.
    - i. La eliminación debe ser lógica modificando el campo activo.
5. No implementar ningún mecanismo de seguridad, la API será abierta.
6. **Opcional:** Documentar a través de OpenAPI (swagger o similar) o manualmente a través de un documento de Word como es que se deben acceder a los endpoints por parte de un cliente (aplicación web, móvil, etc) y las respuestas que arrojarán.
  - 1) Especificar el controller/method-name, verbo a utilizar, header y body.
  - 2) Especificar todas las respuestas posibles (Ok, Created, Bad Request, Server Error, etc) de cada endpoint con su estructura Json.

### Generales:

1. El proyecto deberá estar codificado en idioma inglés, así como el modelo de base de datos.
2. Deberá utilizar buenas convenciones del lenguaje Java.
  - a. Respetar notaciones camel case, pascal case, etc. para clases y métodos.
  - b. Cuidar el uso de excepciones.
  - c. Modular el proyecto de tal forma que el código sea legible.
3. Proveer todas las instrucciones para la ejecución correcta de la solución, incluyendo recursos de tablas y población de datos.

### Tiempo:

Duración: 28 horas naturales a partir de su entrega. Si se termina el tiempo entrega lo que se haya finalizado hasta ese momento, **no te sientas frustrado si no lograste terminar.**

### Repositorio:

Debe usar un repositorio personal público como GitHub, Bitbucket, Gitlab, Azure DevOps o similar, subir el proyecto y enviarnos la URL del repositorio junto con la documentación antes de la entrevista de revisión.

Domicilio: Plaza de la paz 102, Suite 705 en piso 7, Col. Guanajuato Puerto Interior C.P. 36275 Silao Gto.

Teléfono: +52 (477) 274 8866 Ext. 705

Email: [contacto@forteinnovation.mx](mailto:contacto@forteinnovation.mx)

[forteinnovation.mx](http://forteinnovation.mx)

## EJERCICIO BASE DE DATOS T-SQL

1. Dada la siguiente base de datos de un comercio de venta de productos por internet realizar las consultas solicitadas a través de Transact-SQL

Nombre Tabla:		<b>Cientes</b>				
Id	Nombre	Apellidos	Sexo	Edad	Estado	Ciudad
1	Oscar	Huerta Sánchez	M	25	Guanajuato	León
2	Octavio	Sánchez Ortega	M	30	Guanajuato	Irapuato
3	Julio	Morales Camarena	M	23	Jalisco	Guadalajara
4	María	Díaz Arellano	F	25	Michoacán	La Piedad
5	Lorena	López Ortiz	F	30	Nuevo León	Monterrey
6	Fernando	Huerta Mireles	M	28	Jalisco	Lagos de Moreno
7	Cassandra	Contreras Pérez	F	22	Guanajuato	León

Ejemplo:

Sentencia que arroje todos los registros con todas sus columnas

**R = SELECT \* FROM Cientes;**

- 1) Sentencia que arroje aquellos registros cuyo sexo sea masculino

**R=**

- 2) Escribir una sentencia que actualice el registro con Id 6 indicando que la ciudad es San Juan de los Lagos

**R=**

- 3) Escribir una sentencia que arroje aquellas personas que están entre 20 y 29 años

**R=**

- 4) Escribir una sentencia que arroje los registros cuyos Apellidos contienen el apellido "Huerta"

**R=**

- 5) Sentencia que arroje el conteo de los registros que están en el estado de Guanajuato.

**R=**

- 6) Escribir una sentencia que inserte un nuevo registro a dicha tabla considerando que todos los campos son obligatorios.

**R=**

¡Mucho éxito! Nos vemos en la siguiente. 😊