

TALLER #4

Sobre una base de datos se realizan 4 operaciones básicas a través del DML (Data Manipulation Language), insertar (Create), consultar (Read), actualizar (Update), eliminar (Delete). Conociendo la estructura de una tabla y los datos del registro a ingresar, es posible generar las sentencias más comunes para el CRUD.

Tomemos como ejemplo la tabla `tbl_usuarios` con la siguiente estructura:

`tbl_usuarios`

- `id_usuario`: Autonumérico y parte de la clave primaria
- `nombre`: `varchar(200)`, no permite nulos
- `apellido`: `varchar(200)`, no permite nulos
- `nacimiento`: `date`, permite nulos
- `telefono`: `varchar(50)` permite nulos

Y los siguientes valores

- `id_usuario`: 11
- `nombre`: Juan
- `apellido`: López
- `telefono`: NULL
- `nacimiento`: '1999-08-13'

Tenemos entonces las siguientes estrategias para generar las sentencias:

Insertar: Se genera la instrucción teniendo en cuenta todas las columnas no auto-incrementales.

```
INSERT INTO tbl_usuarios (nombre, apellido, telefono, nacimiento) VALUES ('Juan', 'López', null, '1999-08-13');
```

Consultar: Se crea la consulta solicitando todas las columnas de la tabla y condicionando la consulta por los datos no nulos del registro con AND.

```
SELECT
    id_usuario,
    nombre,
    apellido,
    telefono,
    nacimiento
FROM tbl_usuarios
WHERE
    id_usuario = 1
    AND nombre = "Juan"
    AND apellido = "López"
    AND nacimiento = '1999-08-13';
```

Actualizar: Se actualizan todos los campos que no hagan parte de la clave primaria y se condiciona la instrucción a los valores que se tienen en la clave primaria.

```
UPDATE tbl_usuarios SET nombre = 'Juan', apellido = 'López', telefono = null, nacimiento = '1999-08-13'  
WHERE id_usuario = 1;
```

Eliminar: Se elimina el registro condicionando la instrucción con valores que se tienen en la clave primaria.

```
DELTE FROM tbl_usuarios WHERE id_usuario = 1;
```

Teniendo en cuenta lo anterior se desea realizar un programa (en java) que leyendo un archivo de entrada (ingresado por parámetro desde consola) genere los scripts.

ENTRADA.

El archivo de entrada es un archivo en texto plano con la siguiente estructura:

Cada 5 líneas se define un caso de prueba así:

- **Primeria línea:** Nombre de la tabla.
- **Segunda línea:** Columnas de la tabla separadas por espacio.
- **Tercera línea:** Comienza con un entero K indicando el número de columnas que hacen parte de la clave primaria, seguido de las K columnas de la clave primaria. (todo separado por espacios).
- **Cuarta línea:** Comienza con un entero S indicando el número de columnas que son auto-incrementales, seguido de las S columnas auto-incrementales. (todo separado por espacios).
- **Quinta línea:** Contiene los valores cada una de las columnas indicadas en primera línea separados por coma. El primer valor a la primera columna, el segundo valor a la segunda columna y así respectivamente.

El archivo termina cuando no hay más líneas por leer.

SALIDA.

Por cada una de las anteriores 5 líneas se debe escribir en el archivo de salida las siguientes ocho líneas:

1. Sentencia INSERT, donde los valores se reemplazan con signo de interrogación (?).
2. Valores correspondientes a los signos de interrogación de la línea anterior, separados por coma. (conservando el orden de los mismos)
3. Sentencia SELECT, donde los valores se reemplazan con signo de interrogación (?)
4. Valores correspondientes a los signos de interrogación de la línea anterior, separados por coma.
5. Sentencia UPDATE, donde los valores se reemplazan con signo de interrogación (?).
6. Valores correspondientes a los signos de interrogación de la línea anterior, separados por coma.
7. Sentencia DELTE, donde los valores se reemplazan con signo de interrogación (?).
8. Valores correspondientes a los signos de interrogación de la línea anterior, separados por coma.

ENTREGABLES:

- Archivo *.jar para ejecutar la aplicación a través de consola. Debe recibir un único parámetro con la ruta del archivo que contiene las expresiones a evaluar.
- El archivo *.out se debe generar en la misma ruta y nombre del archivo *.in (sólo cambia la extensión).
- Archivo .zip con los códigos fuentes del proyecto.

A TENER EN CUENTA:

- Enviar al correo diegothomas@gmail.com antes del **21 de marzo a las 21:00**.
- Es individual, no se engañe y procure desarrollar la actividad a conciencia.
- El asunto del correo debe decir CES2 Taller#4 y el nombre del estudiante en el contenido.
- Cualquier duda al correo diegothomas@gmail.com.