MDY3131 - 004V PROGRAMACIÓN DE BASE DE DATOS

Evidencias actividad tipos de datos LOB

|  |  |
| --- | --- |
| INTEGRANTES | Yehudi Lara  Yael Velásquez |
| FECHA ENTREGA | 05/05/22 |

**IMPORTANTE**:

* El script con el desarrollo de esta actividad se encuentra en el archivo ***act\_imgs.sql***;
* Las imágenes utilizadas en la actividad se encuentran en la carpeta ***MDY3131***;
* Tanto el archivo .sql como la carpeta se envían junto a este documento Word como parte de la entrega.

Contenido

[Actividad tipos de datos LOB 2](#_Toc102603759)

[Introducción 2](#_Toc102603760)

[1. Insertar un registro 2](#_Toc102603761)

[2. Actualizar un registro 5](#_Toc102603762)

[3. Eliminar un registro 6](#_Toc102603763)

[4. Actualización masiva de imágenes 8](#_Toc102603764)

# Actividad tipos de datos LOB

## Introducción

Se crea la tabla DISCO con una **columna de tipo BFILE** para almacenar imágenes de portadas de discos.

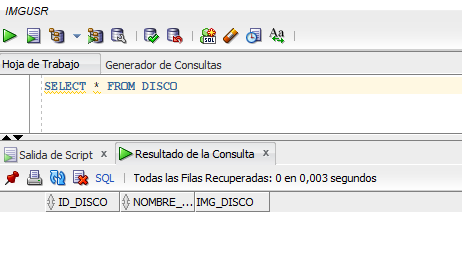
En el bloque PL/SQL, se declara la variable bind **B\_ACCION** para controlar las siguientes opciones:

1. Insertar un registro
2. Actualizar un registro
3. Eliminar un registro
4. Actualización masiva de imágenes

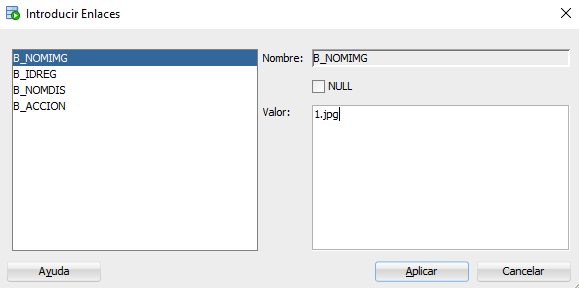
A continuación, se presentan las **evidencias de las pruebas realizadas para las cuatro acciones indicadas** anteriormente.

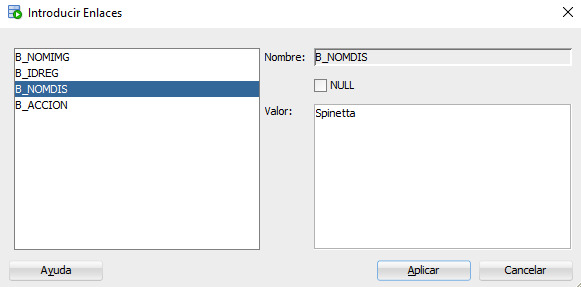
## 1. Insertar un registro

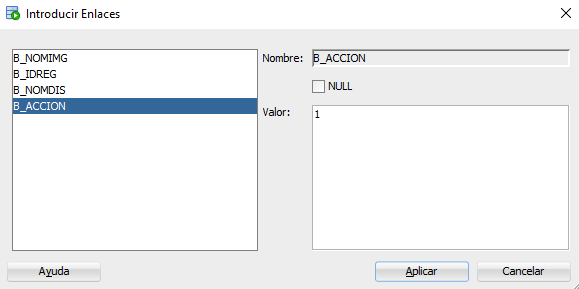
Tabla vacía:



Seteo de valores de variables bind:

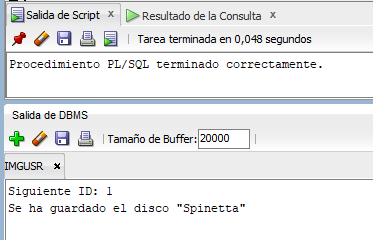




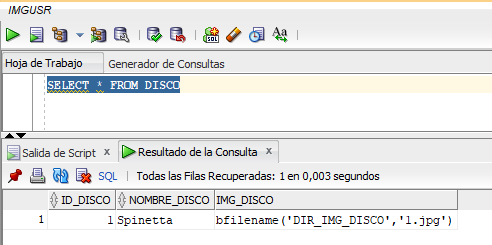


(B\_ACCION = 1 indica insertar)

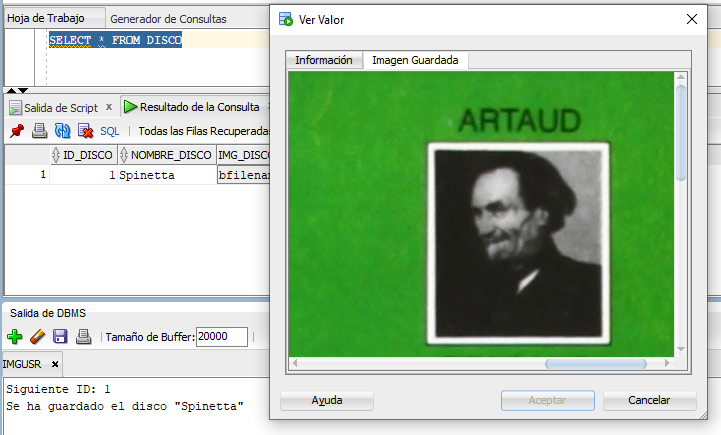
Se ejecuta el bloque con los valores configurados:



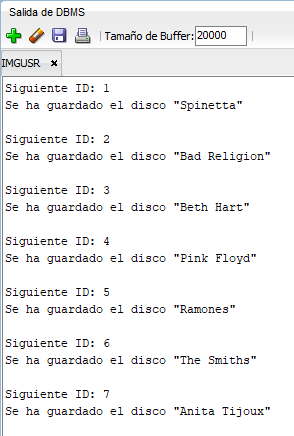
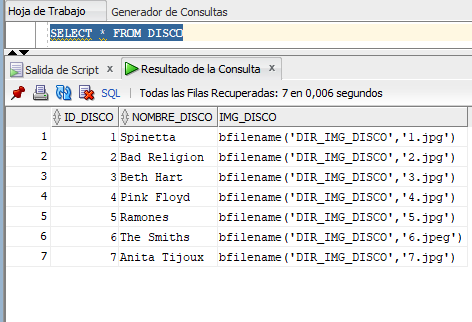
Se consulta la tabla nuevamente para visualizar el registro insertado:



Al hacer doble clic en el campo IMG\_DISCO, se puede visualizar la imagen referenciada por la variable BFILE:

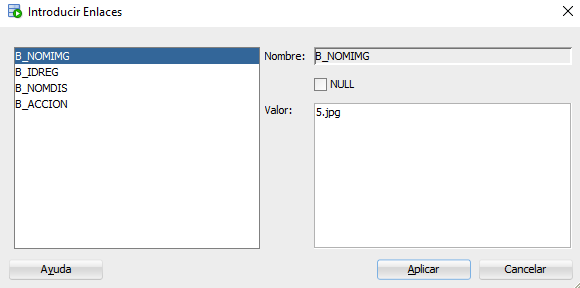


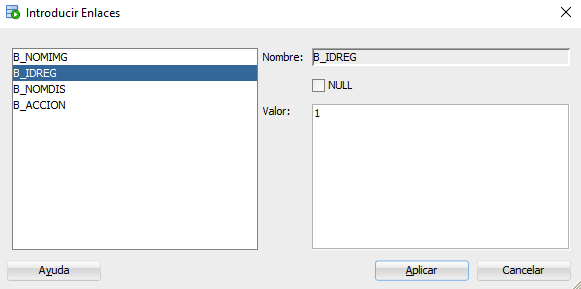
Se realiza el mismo procedimiento para insertar nuevos registros:

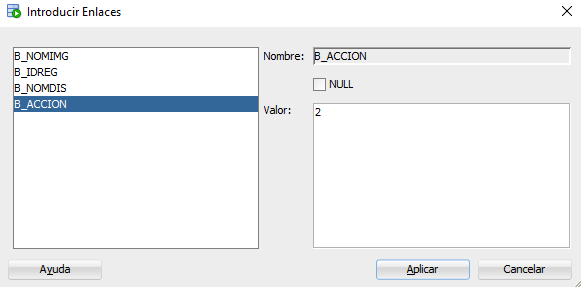
 

## 2. Actualizar un registro

Seteo de valores de variables bind para actualizar el registro con id = 1 (se le asignará la imagen 5.jpg):

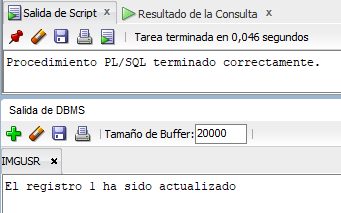




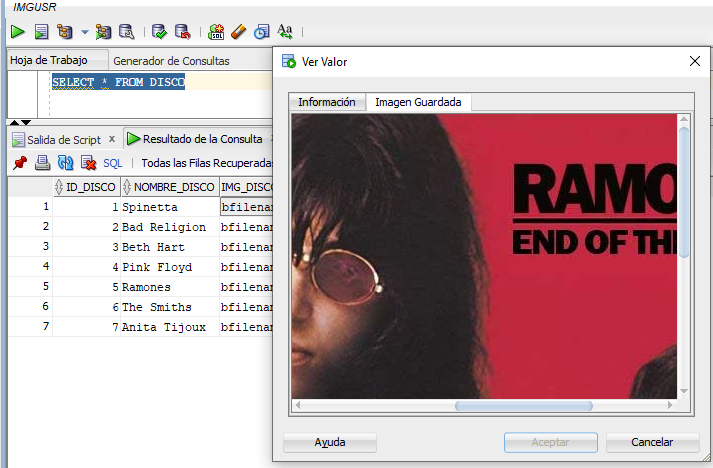
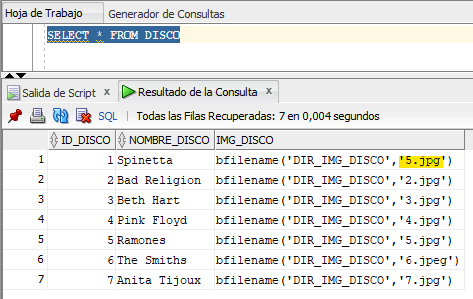


(B\_ACCION = 2 indica actualizar un registro)

Se ejecuta el bloque:

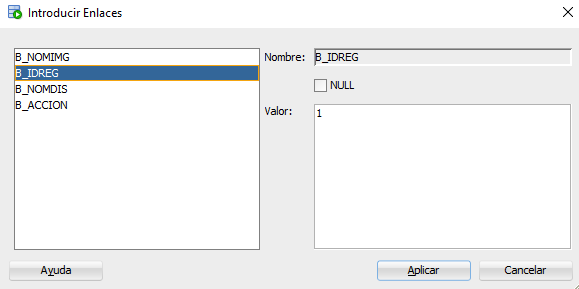


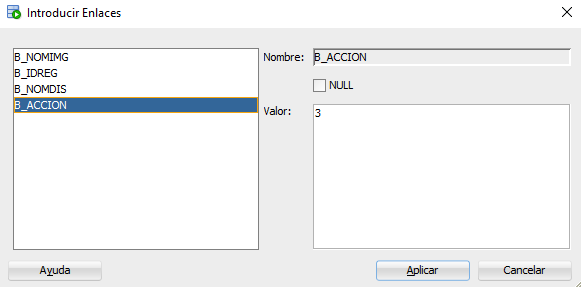
Se valida que el registro con id = 1 ha sido actualizado:



## 3. Eliminar un registro

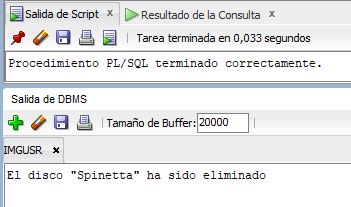
Seteo de valores de variables bind para eliminar el registro con ID\_DISCO = 1:



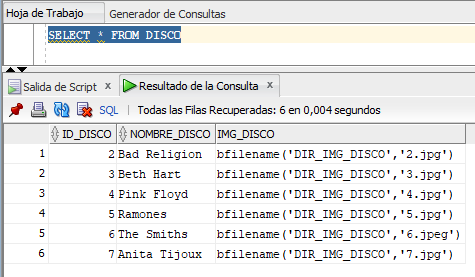


(B\_ACCION = 3 indica eliminar un registro)

Se ejecuta el bloque:



Se consulta la tabla y el registro con ID\_DISCO = 1 ha sido borrado:



## 4. Actualización masiva de imágenes

Imagen del registro con ID\_DISCO = 2 **antes** de la ejecución de la actualización masiva:

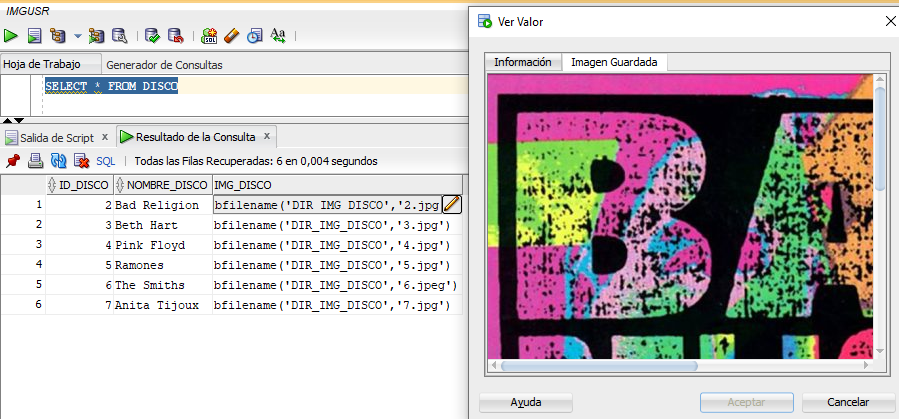
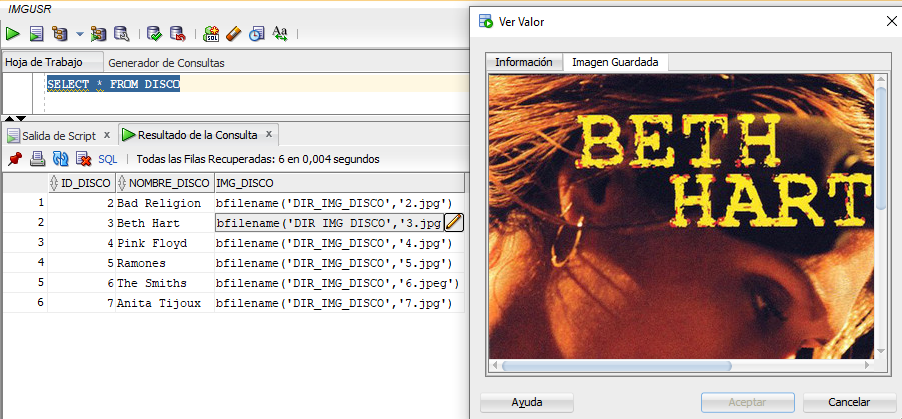
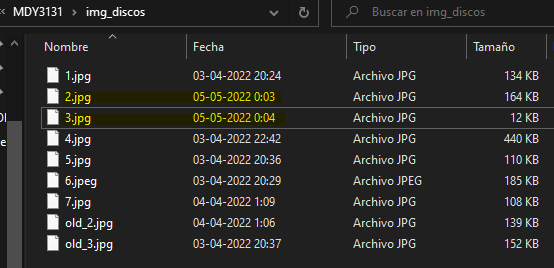


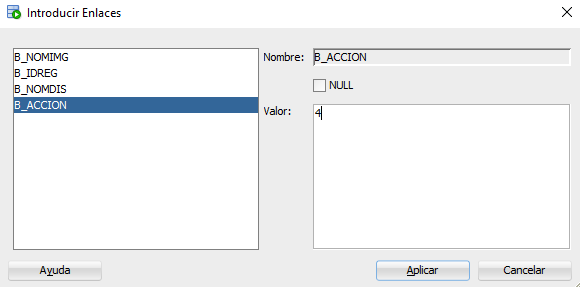
Imagen del registro con ID\_DISCO = 3 **antes** de la ejecución de la actualización masiva:



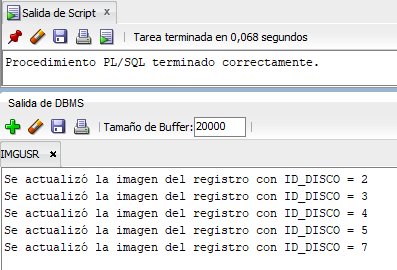
En el directorio de imágenes, se reemplazan los archivos 2.jpg y 3.jpg por nuevas imágenes:



Se setea la variable B\_ACCION = 4 para actualización masiva:

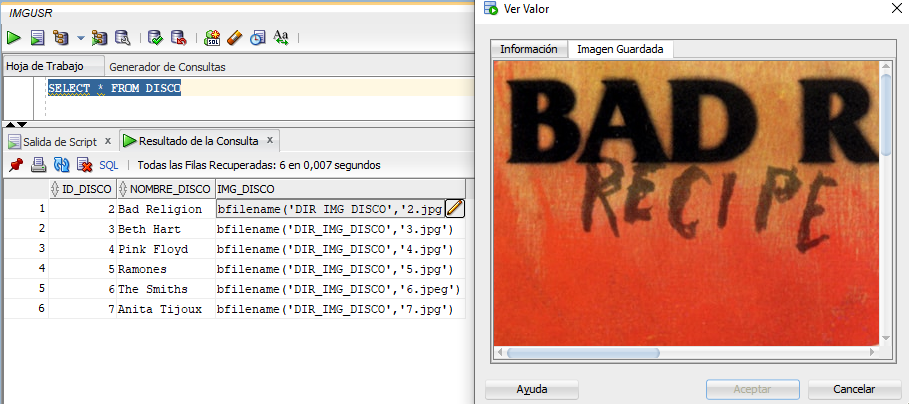


Se ejecuta el bloque:

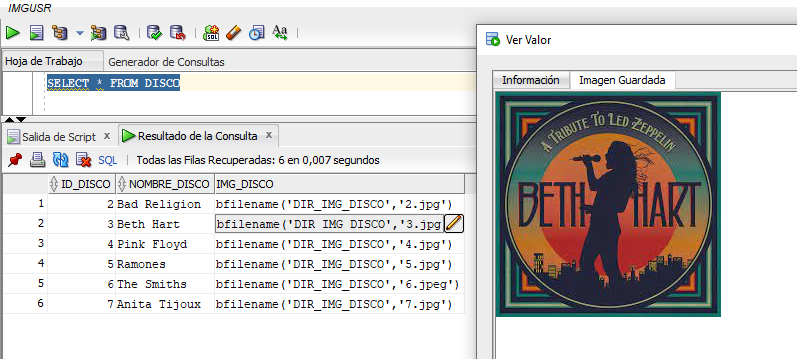


Se consulta la tabla para revisar las imágenes actualizadas:

* Imagen actualizada para registro 2 (Bad Religion):



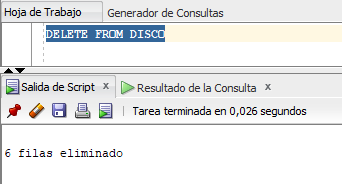
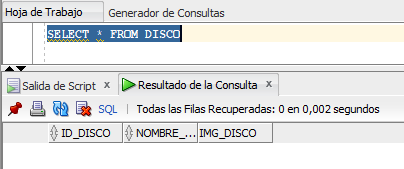
* Imagen actualizada para registro 3 (Beth Hart):



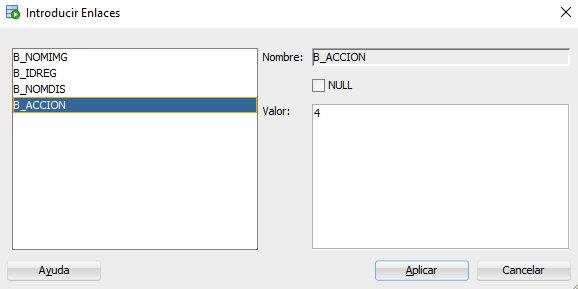
**Prueba adicional para opción 4**

Ejecutar opción de actualización masiva con la tabla vacía. Resultado esperado: “No es posible actualizar registros porque la tabla está vacía”.

* Se eliminan los registros de la tabla y se consulta la tabla vacía:

* Se setea B\_ACCION = 4 (actualización masiva):



* Se ejecuta el bloque, obteniendo el resultado esperado:

