**Luis Fernando Sena-BDA2-Evidencia 1-Hito2**

**Pregunta 1**

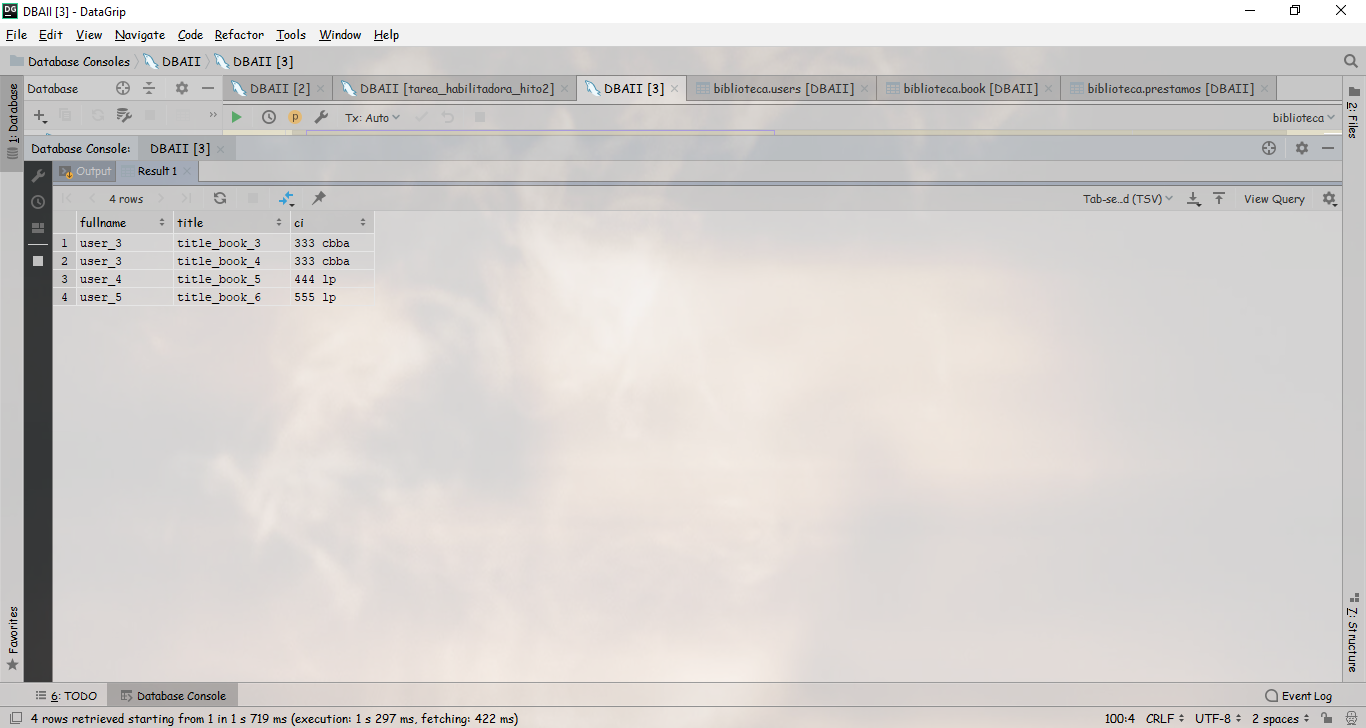
Mostrar el título del libro, los nombres, CI de los usuarios que se prestaron libros donde la categoría sea COMEDIA o MANGA.

**Consulta:**

**select** per.**fullname**,b.**title**,per.**ci  
 from** users **as** per **inner join** prestamos p **on** per.**id\_user** = p.**id\_user  
 inner join** book b **on** p.**id\_book** = b.**id\_book  
 inner join** category c **on** b.**id\_book** = c.**id\_book**

**where** c. **type** =**'manga' or** c. **type**=**'comedia'**;

**Resultado esperado:**



**Pregunta 2**

Se desea saber cuántos usuarios se prestaron libros que correspondan a la editorial IBRANI y que la cantidad de sus páginas sea mayor a 90.  
Deberá de crear una función que retorne esa cantidad.  
La función debe de recibir 2 parámetros.  
Editorial y la cantidad de páginas.

**Consulta:**

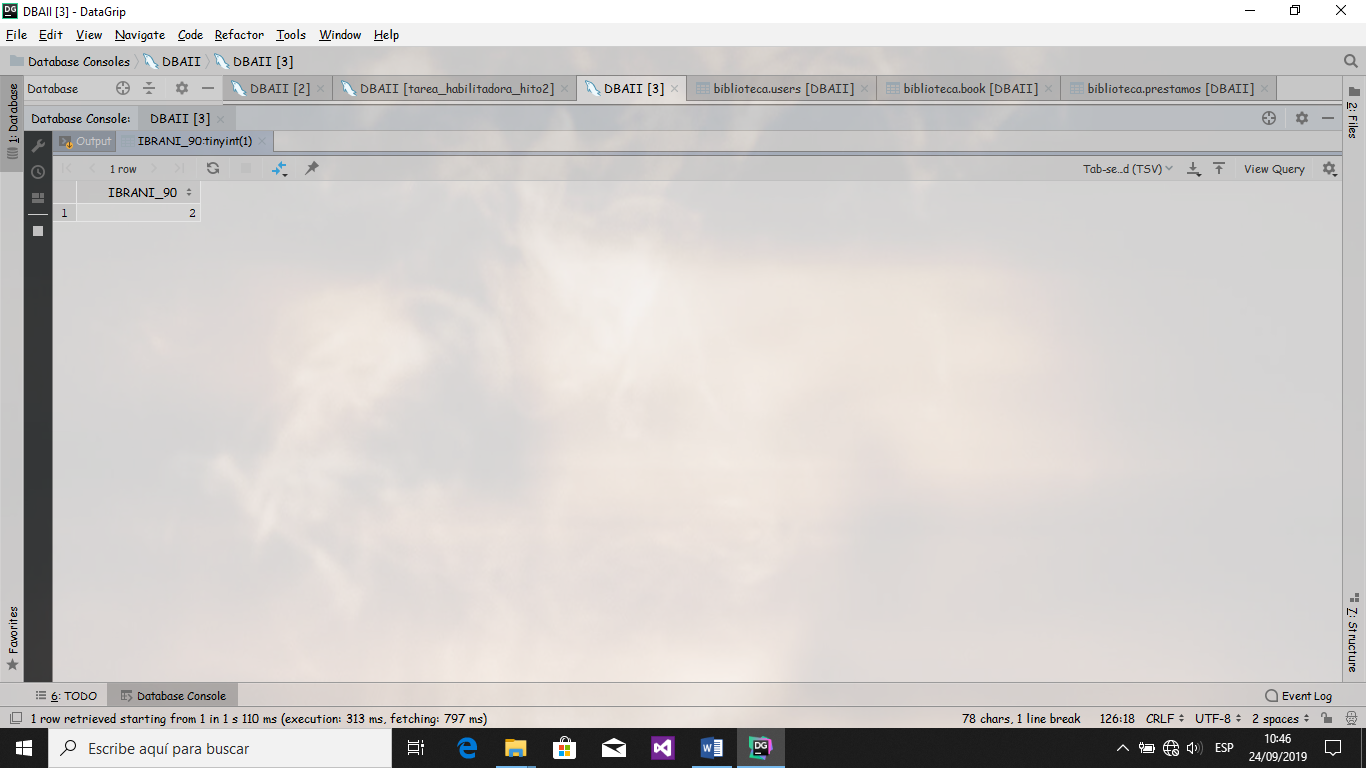
Creando función:

**create function** *encontrar*(editorial **text**, catnpag **int** )**returns boolean  
 begin  
 declare** busca **boolean**;  
 **set** busca = *if*(editorial =**'IBRANI'**&& catnpag >=90 ,1,0);  
 **return** busca;  
 **end**;

Query utilizando la función:

**select** *sum*( *encontrar*(lib.**editorial**,lib.**pages**)) **as** IBRANI\_90  
 **from** book **as** lib;

**Resultado esperado:**



**Pregunta 3**

Se desea saber qué libros tienen una cantidad de páginas par de la categoria MANGA.  
 Deberá de crear una función que verifique si es par o no y retornar un TEXT que indique PAR o IMPAR adicionalmente concatenado el número de páginas.  
 Esta función recibe como parámetro la cantidad de páginas.  
 La función deberá ser usada en la cláusula SELECT.  
 Deberá de crear una función que concatena cadenas.  
 Se tiene el siguiente comportamiento esperado. (Debe de usar los mismos alias que de la imagen)

**Consulta:**

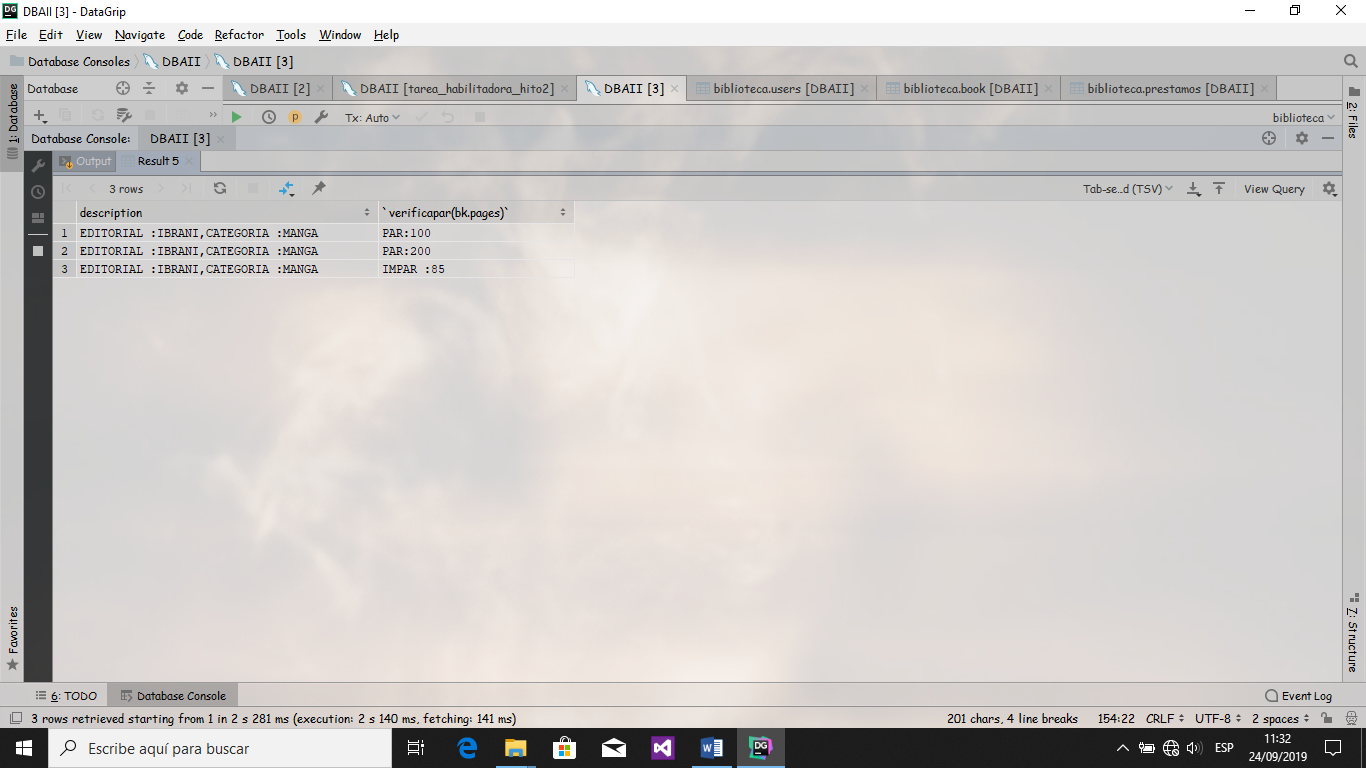
Creando función:

**create function** *verificapar*(paginaslibro **integer**)**returns text  
 begin  
 declare** respuesta **text default ''**;  
  
 **if** paginaslibro % 2=0  
 **then  
 set** respuesta = *concat*(**'PAR:'**, paginaslibro);  
 **else  
 set** respuesta = *concat*(**'IMPAR :'**,paginaslibro);  
 **end if**;  
 **return** respuesta;  
 **end**;

Query utilizando la función:

**select** *concat*(**'EDITORIAL :'**, bk.**editorial**,**','**,**'CATEGORIA :'**, c.**type**) **as description**,  
 *verificapar*(bk.**pages**)  
 **from** book **as** bk  
 **inner join** category c **on** bk.**id\_book** = c.**id\_book  
 where** c.**type**=**'MANGA'**;

**Resultado esperado:**



**Pregunta 4**

Crear una función que pueda eliminar, modificar o eliminar un registro de la tabla AUTOR.  
 La función recibe 4 parámetros. deleteOrModifyOrInsertTableAutor('delete', 6, 'autor\_name\_6', 'NICARAGUA')  
 El primer parámetro es la acción a realizar (puede ser delete - insert - update).  
 El segundo parámetro es el ID del autor  
 El tercer parámetro es el nombre del autor.  
 El cuarto parámetro es la nacionalidad del autor.  
 Si la acción es delete, debe de eliminar el registro con el ID que llega en el 2do parámetro.  
 Si la acción es update deberá de modificar el registro. (En este caso se utilizarán todos los parámetros)  
 Si la acción es insert deberá de crear un nuevo registro en la tabla autor.

**Consulta:**

Creando función:

**create function** *modificador*( accion **text**, idAutor **integer**, nameAutor **text**, nacionalidadAutor **text**) **returns text  
begin**

**if** accion =**'delete'  
 then  
 delete from** autor **where id\_autor** = idAutor;  
 **end if**;

**if** accion=**'actualizar'  
 then  
 update** autor **set name** = nameAutor,**nacionality**=nacionalidadAutor **where id\_autor** = idAutor;  
 **end if**;

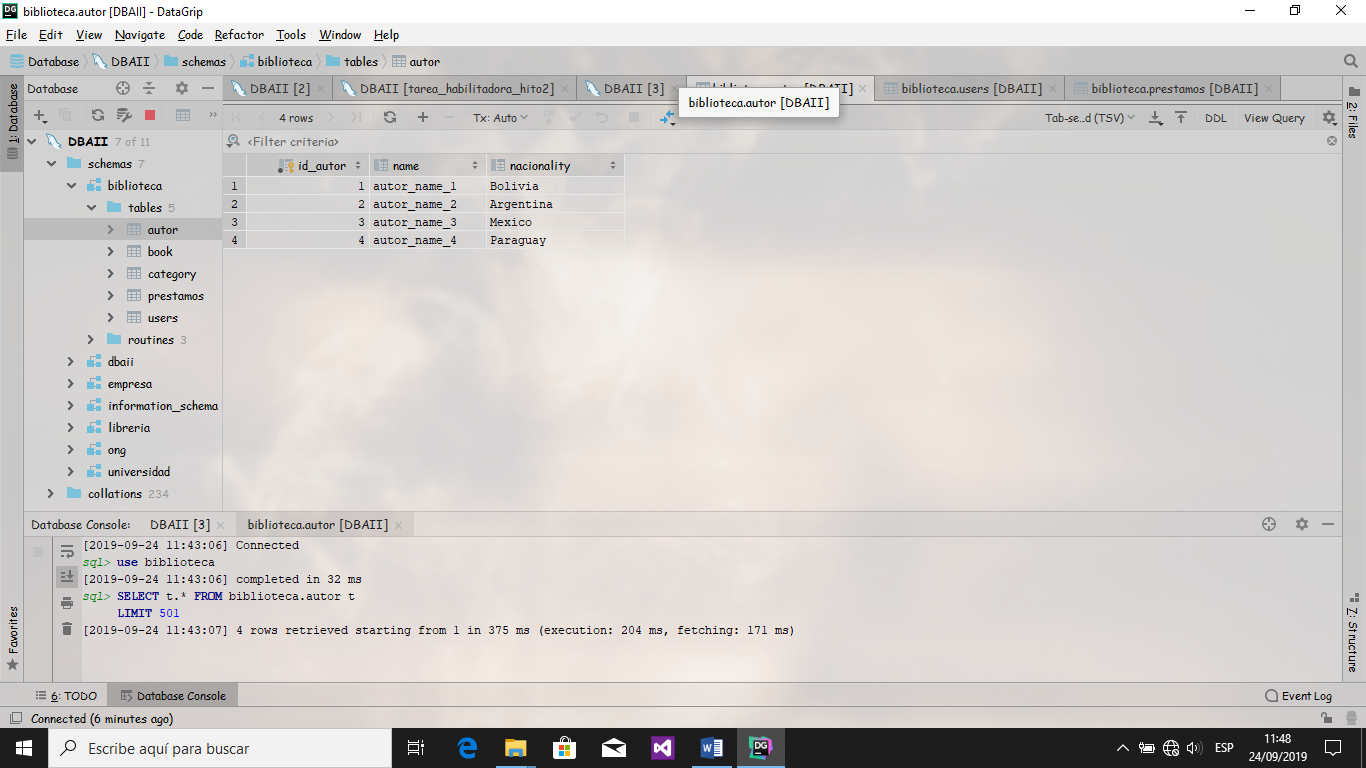
**if** accion =**'insertar'  
 then  
 insert into** autor(**name**,**nacionality**) **values** (nameAutor , **nacionality**);  
 **end if**;  
  
 **return** *concat*(accion,**'accion realizada exitosamente'**);  
**end**;

Query utilizando la función:

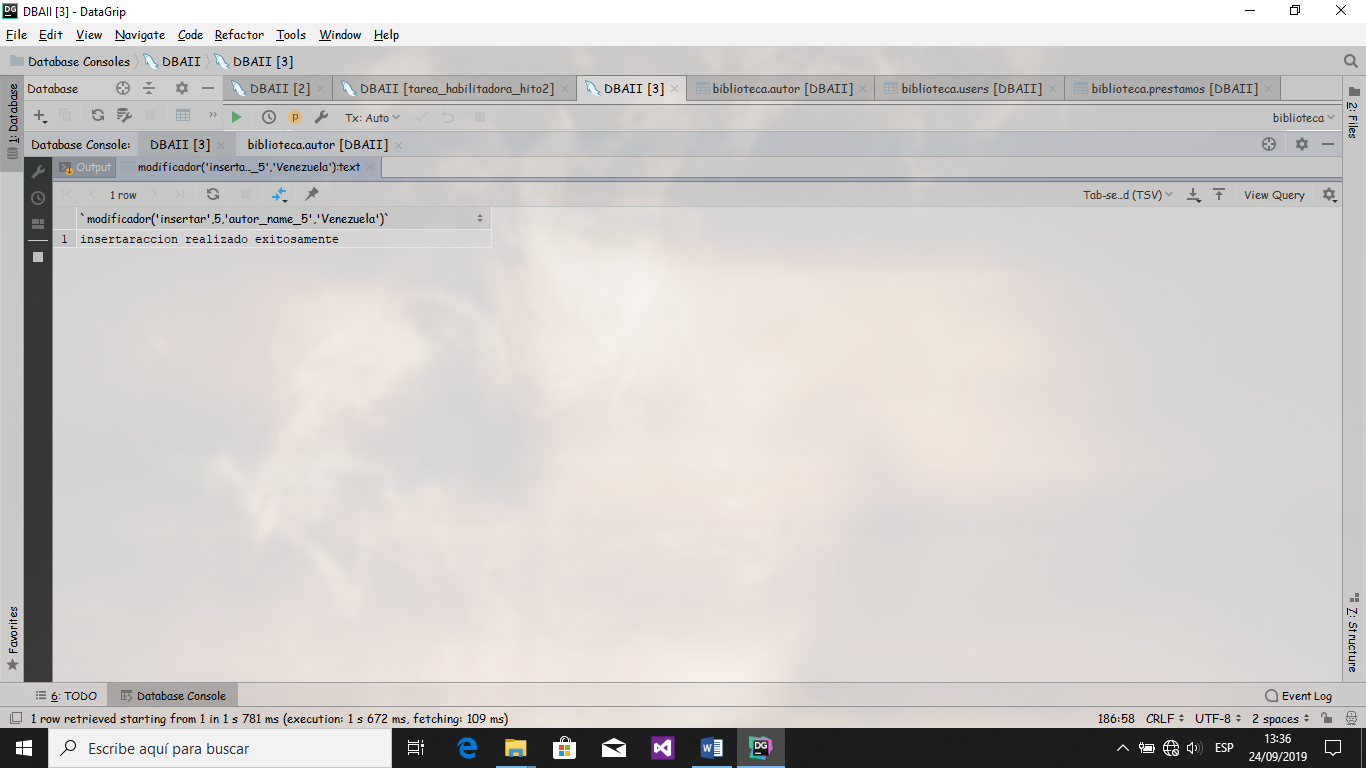
**select** *modificador*(**'delete'**,9,**'autor\_name\_5'**,**'chile'**);

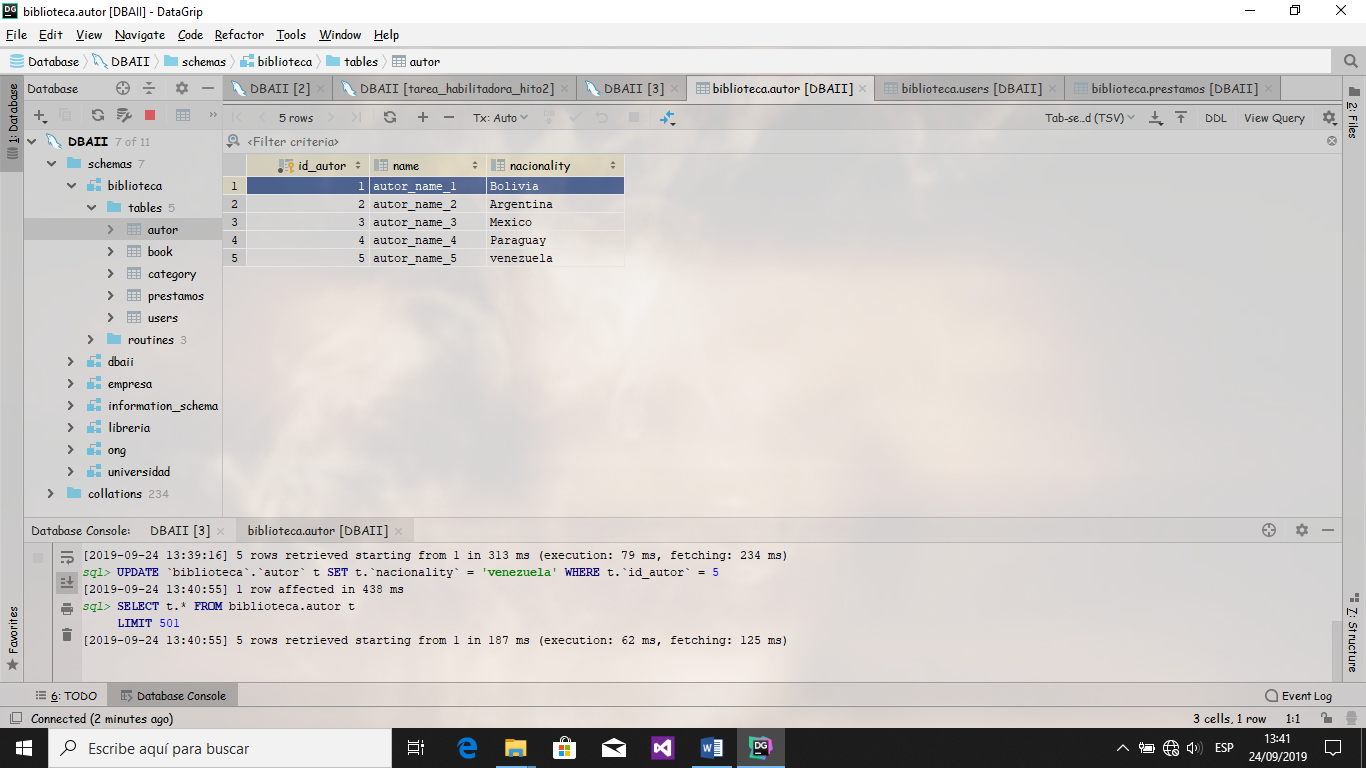
**Resultado esperado:**

Tabla antes de ser modificada:



Respuesta del Query:



 Actualización de la tabla:

**Pregunta 5**

Se desea saber cuántos libros fueron prestados en la gestión 2018.  
 Crear una función que permita saber esa cantidad.  
 Comportamiento esperado.

**Consulta:**

**select** *count*(pre.**id\_prestamo**)**as** cantidad\_2018  
 **from** prestamos **as** pre  
 **where** pre.**fec\_prestamo**>**'2018-01-01'and** pre.**fec\_prestamo**<**'2018-12-30'**;

**Resultado esperado:**

