



Universidad Autónoma de Chihuahua

Facultad de ingeniería

Computo Paralelo y Distribuido

3.25 Proyecto 2a. Evaluación

Asesor: José Saul de Lira Miramontes

Ángel Eduardo Garibay Valenzuela 348775

Jair Alejandro Gaytán Espíndola 353205

Grupo: 8CC2

Noviembre 14, 2024

Crear una base de datos distribuida para un Sistema Bancario, utilizando el servidor de base de datos Oracle

Operaciones:

1. Instalación de dos servidores de base de datos Oracle (Virtuales/Docker)(Site A y B)

Container CPU usage ⓘ
No containers are running.

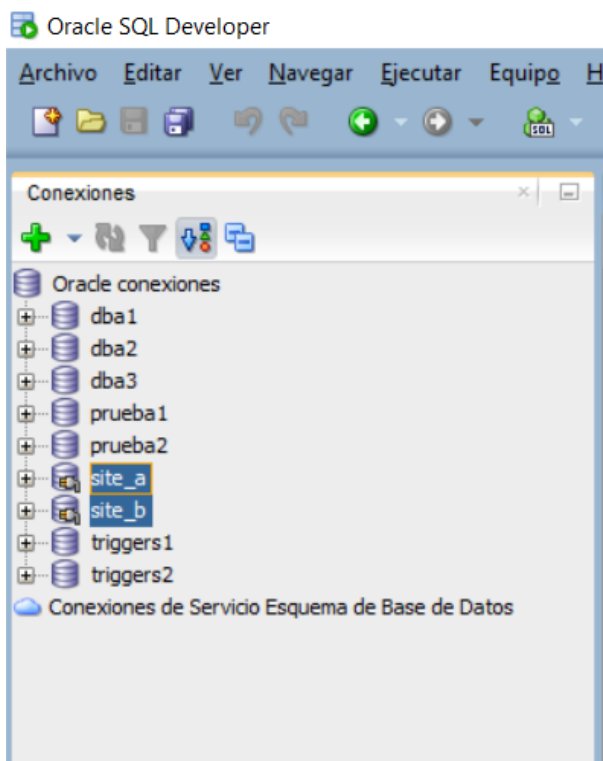
Container memory usage ⓘ
No containers are running.

[Show charts](#)

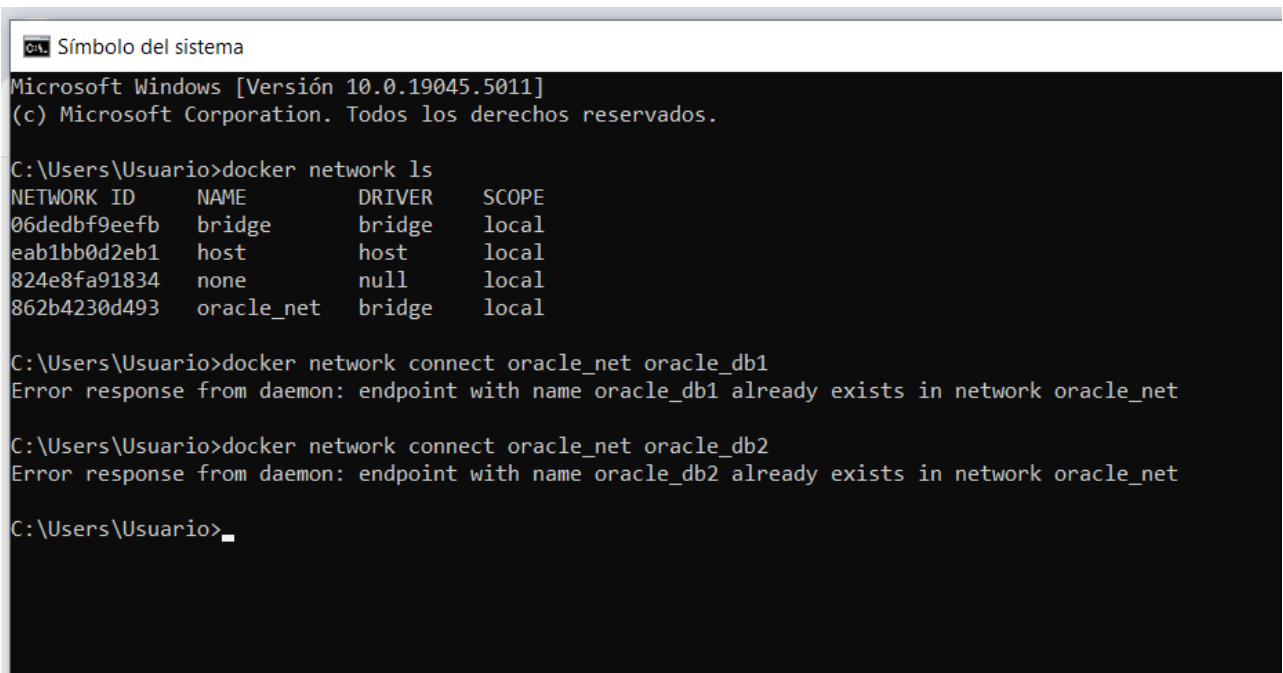
Search

Only show running containers

	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	oracle_db3	45c389f7e2db	oracleinanutshell/oracle-xe	1523:1521	N/A	6 hours ago	
<input type="checkbox"/>	oracle_db2	e50516de270c	oracleinanutshell/oracle-xe	1522:1521 ↗	N/A	3 seconds ago	
<input type="checkbox"/>	oracle_db1	d39909a09c67	oracleinanutshell/oracle-xe	1521:1521 ↗	N/A	4 seconds ago	



2. Site A y B => Crear un servicio de Oracle entre cada uno de los servidores (opcional)



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.5011]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Usuario>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
06dedbf9eefb        bridge              bridge              local
eab1bb0d2eb1        host                host                local
824e8fa91834        none                null                local
862b4230d493        oracle_net          bridge              local

C:\Users\Usuario>docker network connect oracle_net oracle_db1
Error response from daemon: endpoint with name oracle_db1 already exists in network oracle_net

C:\Users\Usuario>docker network connect oracle_net oracle_db2
Error response from daemon: endpoint with name oracle_db2 already exists in network oracle_net

C:\Users\Usuario>
```

3. Site A y B => Crear un database link entre cada uno de los servidores

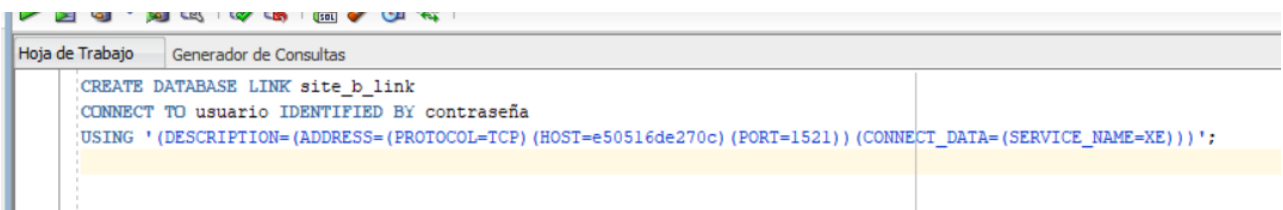
site_a:

CREATE DATABASE LINK site_b_link

CONNECT TO usuario IDENTIFIED BY contraseña

USING

'(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=e50516de270c)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=XE)))';



```
Hoja de Trabajo  Generador de Consultas

CREATE DATABASE LINK site_b_link
CONNECT TO usuario IDENTIFIED BY contraseña
USING '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=e50516de270c)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=XE)))';
```

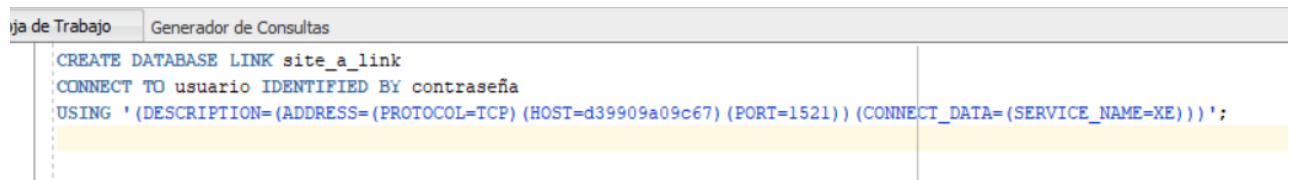
site_b:

CREATE DATABASE LINK site_a_link

CONNECT TO usuario IDENTIFIED BY contraseña

USING

'(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=d39909a09c67)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=XE)))';

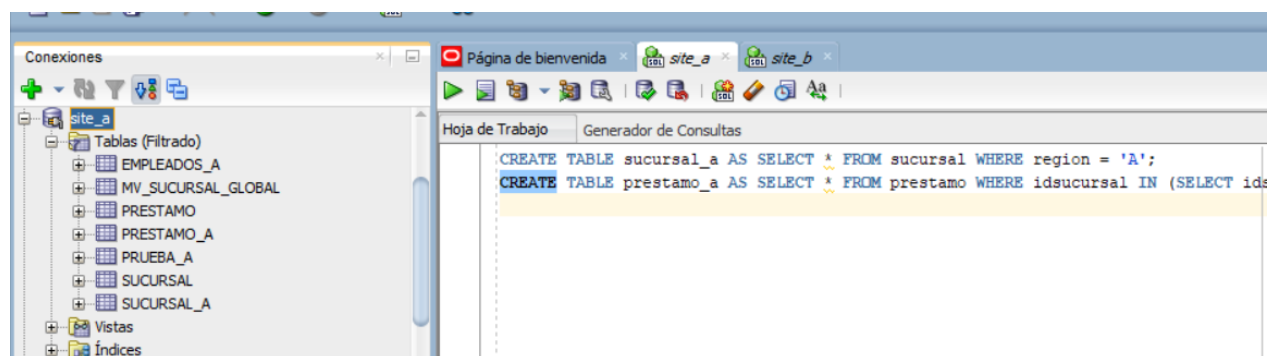


4. Fragmentar las tablas de sucursales(branch) y prestamos(loan), el criterio de fragmentación: región.
5. Asignar los fragmentos de la región 1 en Site A y de la región 2 en el Site B

Site_a:

CREATE TABLE sucursal_a AS SELECT * FROM sucursal WHERE region = 'A';

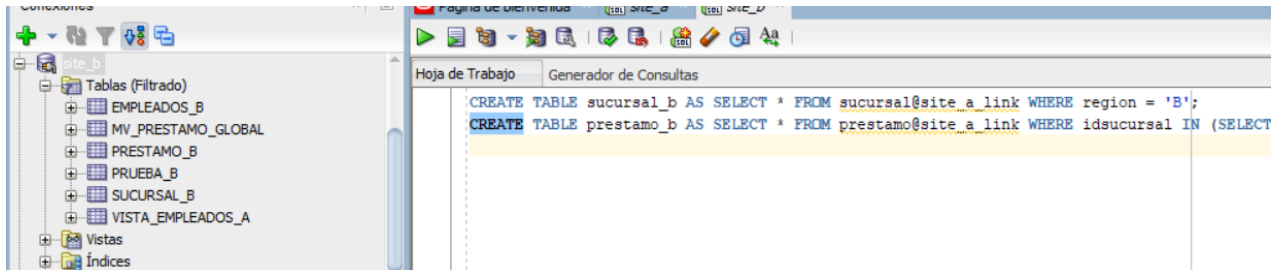
CREATE TABLE prestamo_a AS SELECT * FROM prestamo WHERE idsucursal IN
(SELECT idsucursal FROM sucursal WHERE region = 'A');



Site_b (usando el database link):

```
CREATE TABLE sucursal_b AS SELECT * FROM sucursal@site_a_link WHERE region = 'B';
```

```
CREATE TABLE prestamo_b AS SELECT * FROM prestamo@site_a_link WHERE idsucursal IN (SELECT idsucursal FROM sucursal@site_a_link WHERE region = 'B');
```



6. Site A y B => Crear una vista global de las tablas sucursales(branch) y loan(prestamos)

Site_a

```
CREATE VIEW sucursal_global AS
```

```
SELECT * FROM sucursal_a
```

```
UNION ALL
```

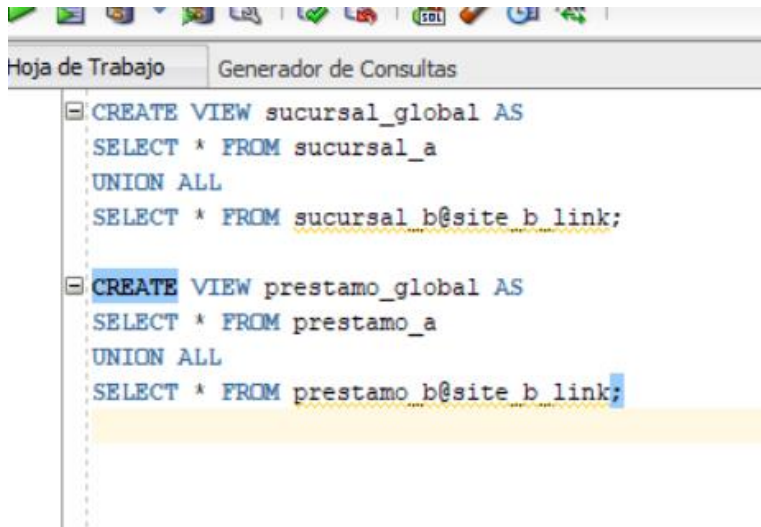
```
SELECT * FROM sucursal_b@site_b_link;
```

```
CREATE VIEW prestamo_global AS
```

```
SELECT * FROM prestamo_a
```

```
UNION ALL
```

```
SELECT * FROM prestamo_b@site_b_link;
```



Site_b

CREATE VIEW sucursal_global AS

SELECT * FROM sucursal_a@site_a_link

UNION ALL

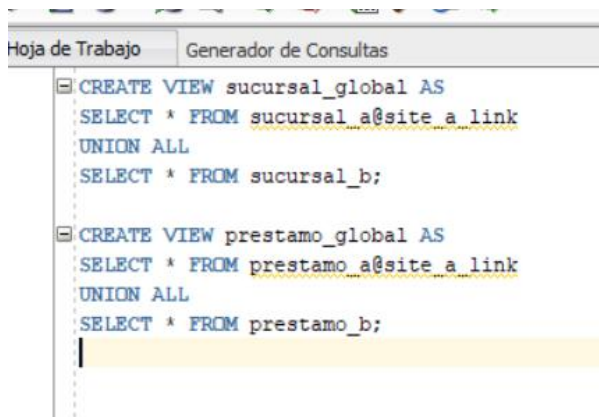
SELECT * FROM sucursal_b;

CREATE VIEW prestamo_global AS

SELECT * FROM prestamo_a@site_a_link

UNION ALL

SELECT * FROM prestamo_b;

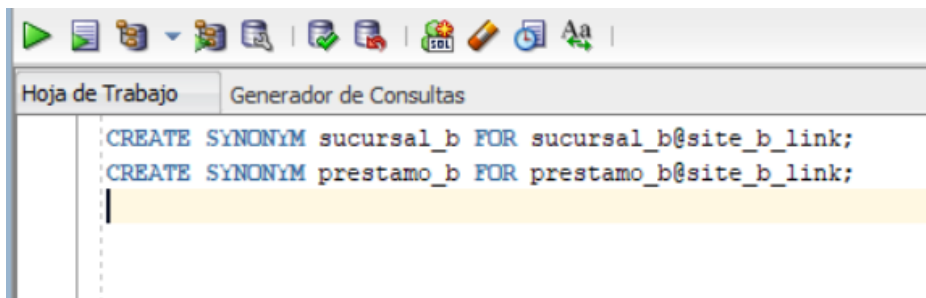


7. Site A y B => Crear un sinónimo para cada fragmento

Site_a

CREATE SYNONYM sucursal_b FOR sucursal_b@site_b_link;

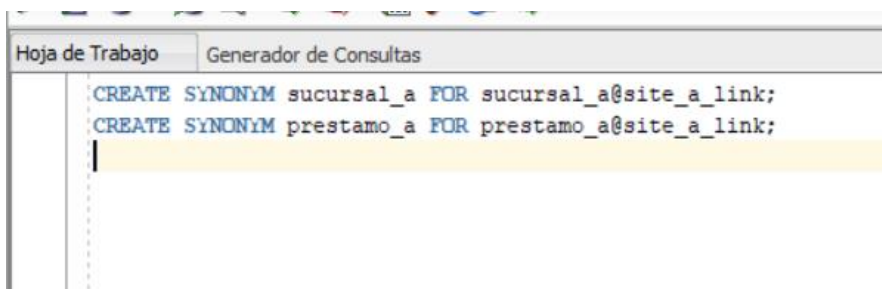
CREATE SYNONYM prestamo_b FOR prestamo_b@site_b_link;



Site_b

CREATE SYNONYM sucursal_a FOR sucursal_a@site_a_link;

CREATE SYNONYM prestamo_a FOR prestamo_a@site_a_link;



8. Site A y B => Crear un procedimiento almacenado que permita dar de alta sucursales(branch) en el Site A o B dependiendo a que región pertenezca

```
CREATE OR REPLACE PROCEDURE alta_sucursal(  
    p_idsucursal VARCHAR2,  
    p_nombre VARCHAR2,  
    p_ciudad VARCHAR2,  
    p_activos NUMBER,  
    p_region VARCHAR2  
) AS  
  
BEGIN  
  
    IF p_region = 'A' THEN  
  
        INSERT INTO sucursal_a VALUES (p_idsucursal, p_nombre, p_ciudad,  
p_activos, p_region);  
  
    ELSE  
  
        INSERT INTO sucursal_b@site_b_link VALUES (p_idsucursal, p_nombre,  
p_ciudad, p_activos, p_region);  
  
    END IF;  
  
END;  
  
/
```


9. Site A y B => Crear un procedimiento almacenado que permita dar de alta prestamos(loan) en el site A o B dependiendo de qué sucursal otorgo el préstamo

```
CREATE OR REPLACE PROCEDURE alta_prestamo(  
    p_noprestamo VARCHAR2,  
    p_idsucursal VARCHAR2,  
    p_cantidad NUMBER  
) AS  
    v_region VARCHAR2(2);  
BEGIN  
    SELECT region INTO v_region FROM sucursal WHERE idsucursal =  
    p_idsucursal;  
  
    IF v_region = 'A' THEN  
        INSERT INTO prestamo_a VALUES (p_noprestamo, p_idsucursal,  
    p_cantidad);  
    ELSE  
        INSERT INTO prestamo_b@site_b_link VALUES (p_noprestamo,  
    p_idsucursal, p_cantidad);  
    END IF;  
END;  
  
/
```

10. Site A y B => Crear un trigger para cada fragmento, que permita replicarlo en el servidor remoto

Site_a (Replicando a site_b)

CREATE OR REPLACE TRIGGER replicar_sucursal

AFTER INSERT ON sucursal_a

FOR EACH ROW

BEGIN

**INSERT INTO sucursal_b@site_b_link VALUES (:NEW.idsucursal,
:NEW.nombresucursal, :NEW.ciudadsucursal, :NEW.activos, :NEW.region);**

END;

/

11. Site A => Crear una vista materializada que recostruya la tabla global de sucursales(branch)

CREATE MATERIALIZED VIEW mv_sucursal_global AS

SELECT * FROM sucursal_global;

12. Site B => Crear una vista materializada que reconstruya la tabla global de préstamos(loan)

CREATE MATERIALIZED VIEW mv_prestamo_global AS

SELECT * FROM prestamo_global;

13. Site A => Crear una vista que obtenga la cantidad total otorgada en préstamos por sucursal

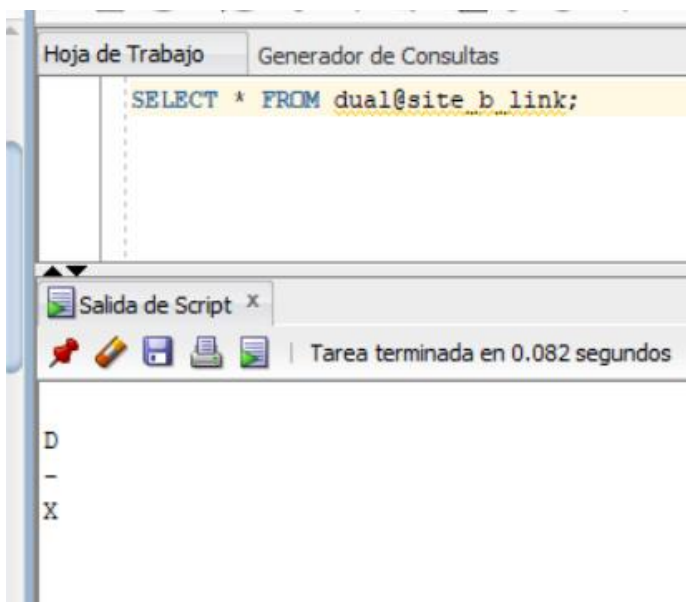
```
CREATE VIEW total_prestamos_por_sucursal AS  
SELECT idsucursal, SUM(cantidad) AS total_prestamos  
FROM prestamo_global  
GROUP BY idsucursal;
```

Comandos para verificar

Database link

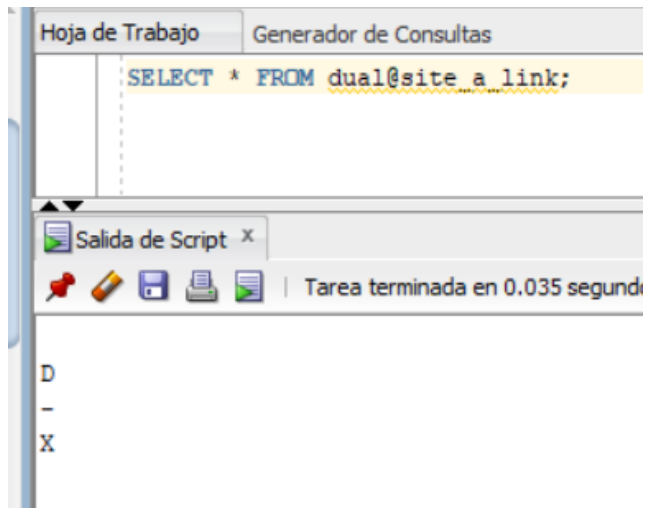
Site_a

```
SELECT * FROM dual@site_b_link;
```



Site_b

SELECT * FROM dual@site_a_link;

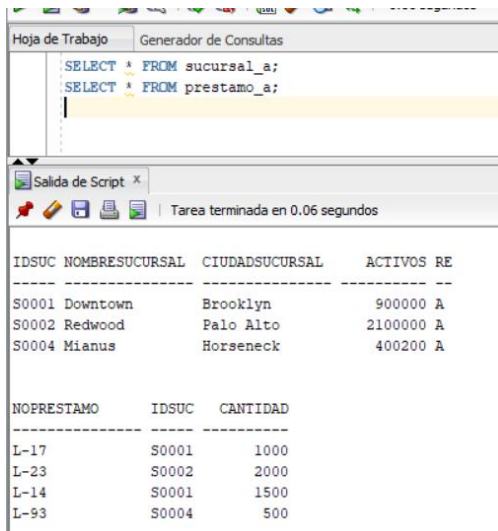


Consultar tablas fragmentadas:

Site_A

SELECT * FROM sucursal_a;

SELECT * FROM prestamo_a;



Hoja de Trabajo Generador de Consultas

```
SELECT * FROM sucursal_a;
SELECT * FROM prestamo_a;
```

Salida de Script x

Tarea terminada en 0.06 segundos

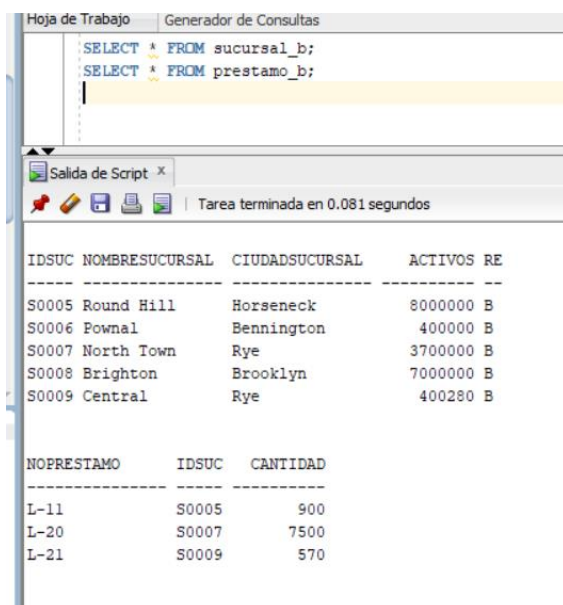
IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0001	Downtown	Brooklyn	900000	A
S0002	Redwood	Palo Alto	2100000	A
S0004	Mianus	Horseneck	400200	A

NOPRESTAMO	IDSUC	CANTIDAD
L-17	S0001	1000
L-23	S0002	2000
L-14	S0001	1500
L-93	S0004	500

Site_b

SELECT * FROM sucursal_b;

SELECT * FROM prestamo_b;



Hoja de Trabajo Generador de Consultas

```
SELECT * FROM sucursal_b;
SELECT * FROM prestamo_b;
```

Salida de Script x

Tarea terminada en 0.081 segundos

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0005	Round Hill	Horseneck	8000000	B
S0006	Pownal	Bennington	400000	B
S0007	North Town	Rye	3700000	B
S0008	Brighton	Brooklyn	7000000	B
S0009	Central	Rye	400280	B

NOPRESTAMO	IDSUC	CANTIDAD
L-11	S0005	900
L-20	S0007	7500
L-21	S0009	570

Consultar las vistas globales:

Site_a

```
SELECT * FROM sucursal_global;
```

```
SELECT * FROM prestamo_global;
```

The screenshot shows a database management interface with two tabs: 'Hoja de Trabajo' and 'Generador de Consultas'. The 'Generador de Consultas' tab is active, displaying two SQL queries:

```
SELECT * FROM sucursal_global;  
SELECT * FROM prestamo_global;
```

Below the queries, a 'Salida de Script' window shows the results of the queries. The first query returns 8 rows from the 'sucursal_global' table, and the second query returns 7 rows from the 'prestamo_global' table.

8 filas seleccionadas.

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0001	Downtown	Brooklyn	900000	A
S0002	Redwood	Palo Alto	2100000	A
S0004	Mianus	Horseneck	400200	A
S0005	Round Hill	Horseneck	8000000	B
S0006	Pownal	Bennington	400000	B
S0007	North Town	Rye	3700000	B
S0008	Brighton	Brooklyn	7000000	B
S0009	Central	Rye	400280	B

7 filas seleccionadas.

NOPRESTAMO	IDSUC	CANTIDAD
L-17	S0001	1000
L-23	S0002	2000
L-14	S0001	1500
L-93	S0004	500
L-11	S0005	900
L-20	S0007	7500
L-21	S0009	570

Site_b

SELECT * FROM sucursal_global@site_a_link;

SELECT * FROM prestamo_global@site_a_link;

The screenshot shows a database query tool interface. The top window, titled 'Hoja de Trabajo', contains two SQL queries: `SELECT * FROM sucursal_global@site_a link;` and `SELECT * FROM prestamo_global@site_a link;`. Below this, a 'Salida de Script' window displays the results of these queries. The first query result is a table with 5 columns: `IDSUC`, `NOMBRESUCURSAL`, `CIUDADSUCURSAL`, `ACTIVOS`, and `RE`. It contains 9 rows of data. The second query result is a table with 3 columns: `NOPRESTAMO`, `IDSUC`, and `CANTIDAD`. It contains 7 rows of data.

8 filas seleccionadas.

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0001	Downtown	Brooklyn	900000	A
S0002	Redwood	Palo Alto	2100000	A
S0004	Mianus	Horseneck	400200	A
S0005	Round Hill	Horseneck	8000000	B
S0006	Pownal	Bennington	400000	B
S0007	North Town	Rye	3700000	B
S0008	Brighton	Brooklyn	7000000	B
S0009	Central	Rye	400280	B

7 filas seleccionadas.

NOPRESTAMO	IDSUC	CANTIDAD
L-17	S0001	1000
L-23	S0002	2000
L-14	S0001	1500
L-93	S0004	500
L-11	S0005	900
L-20	S0007	7500
L-21	S0009	570

Verificar los sinónimos

Site_a

SELECT * FROM sucursal_b;

SELECT * FROM prestamo_b;

The screenshot shows a web application with a browser window displaying 'Página de bienvenida' and tabs for 'site_a' and 'site_b'. The 'Generador de Consultas' (Query Generator) tab is active, showing two SQL queries: `SELECT * FROM sucursal_b;` and `SELECT * FROM prestamo_b;`. Below the queries, a 'Salida de Script' (Script Output) window shows the execution time: 'Tarea terminada en 0.057 segundos'. The results are displayed in two tables.

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0005	Round Hill	Horseneck	8000000	B
S0006	Powmal	Bennington	4000000	B
S0007	North Town	Rye	3700000	B
S0008	Brighton	Brooklyn	7000000	B
S0009	Central	Rye	400280	B

NOPRESTAMO	IDSUC	CANTIDAD
L-11	S0005	900
L-20	S0007	7500
L-21	S0009	570

Site_b

SELECT * FROM sucursal_a;

SELECT * FROM prestamo_a;

The screenshot shows the same web application interface as the previous one, but for Site_b. The 'Generador de Consultas' tab shows two SQL queries: `SELECT * FROM sucursal_a;` and `SELECT * FROM prestamo_a;`. The 'Salida de Script' window shows the execution time: 'Tarea terminada en 0.067 segundos'. The results are displayed in two tables.

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS	RE
S0001	Downtown	Brooklyn	900000	A
S0002	Redwood	Palo Alto	2100000	A
S0004	Mianus	Horseneck	400200	A

NOPRESTAMO	IDSUC	CANTIDAD
L-17	S0001	1000
L-23	S0002	2000
L-14	S0001	1500
L-93	S0004	500

Procedimientos almacenados:

Para insertar una nueva sucursal

Site_a

```
BEGIN alta_sucursal('S0010', 'Branch1', 'New City', 500000, 'A');  
  
END;  
  
/
```

Site_b

```
BEGIN alta_sucursal('S0011', 'Branch2', 'Minessota', 300000, 'B');  
  
END;  
  
/
```

Para insertar un nuevo préstamo

```
BEGIN  
  
alta_prestamo('L-99', 'S0001', 2500);  
  
END;  
  
/
```

Verificar triggers

Inserta en sucursal_a en site_a y verifica en sucursal_b en site_b

```
INSERT INTO sucursal_a VALUES ('S0012', 'Test Branch', 'Test City', 400000, 'A');
```

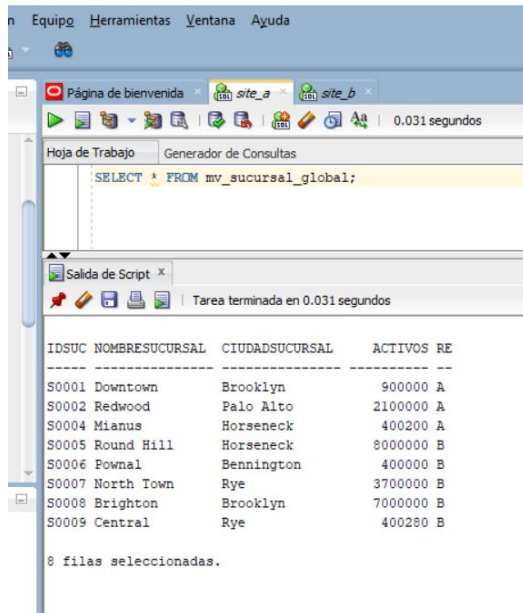
En site_b:

```
SELECT * FROM sucursal_b WHERE idsucursal = 'S0012';
```

Vista materializada

Site_a

SELECT * FROM mv_sucursal_global;



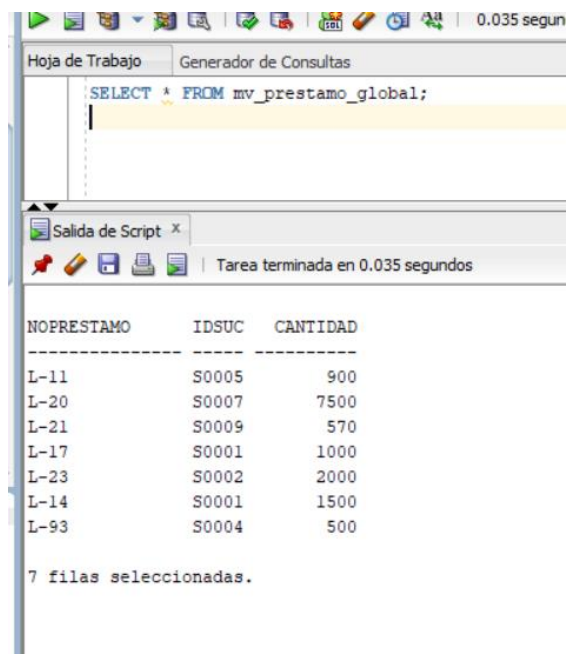
The screenshot shows a database query tool interface. The top menu bar includes 'Equipg', 'Herramientas', 'Ventana', and 'Ayuda'. The toolbar contains various icons for file operations and execution. The 'Hoja de Trabajo' (Worksheet) tab is active, displaying the query: `SELECT * FROM mv_sucursal_global;`. Below the query editor, a status bar indicates 'Tarea terminada en 0.031 segundos'. The results pane shows a table with four columns: 'IDSUC', 'NOMBRESUCURSAL', 'CIUDADSUCURSAL', and 'ACTIVOS RE'. The table contains 9 rows of data. At the bottom, it states '8 filas seleccionadas.'

IDSUC	NOMBRESUCURSAL	CIUDADSUCURSAL	ACTIVOS RE
S0001	Downtown	Brooklyn	900000 A
S0002	Redwood	Palo Alto	2100000 A
S0004	Mianus	Horseneck	400200 A
S0005	Round Hill	Horseneck	8000000 B
S0006	Pownal	Bennington	400000 B
S0007	North Town	Rye	3700000 B
S0008	Brighton	Brooklyn	7000000 B
S0009	Central	Rye	400280 B

8 filas seleccionadas.

Site_b

SELECT * FROM mv_prestamo_global;



The screenshot shows a database query tool interface. The top menu bar includes 'Equipg', 'Herramientas', 'Ventana', and 'Ayuda'. The toolbar contains various icons for file operations and execution. The 'Hoja de Trabajo' (Worksheet) tab is active, displaying the query: `SELECT * FROM mv_prestamo_global;`. Below the query editor, a status bar indicates 'Tarea terminada en 0.035 segundos'. The results pane shows a table with three columns: 'NOPRESTAMO', 'IDSUC', and 'CANTIDAD'. The table contains 7 rows of data. At the bottom, it states '7 filas seleccionadas.'

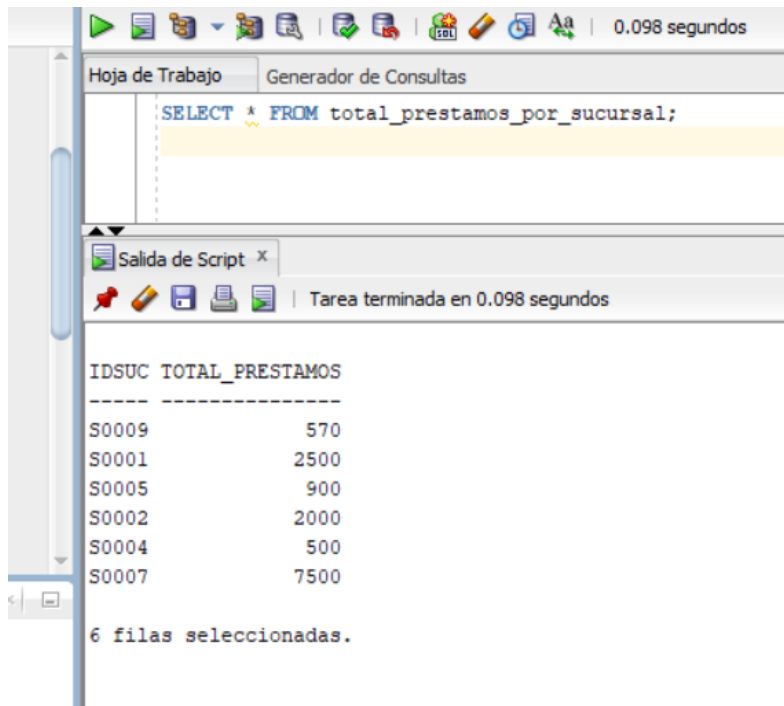
NOPRESTAMO	IDSUC	CANTIDAD
L-11	S0005	900
L-20	S0007	7500
L-21	S0009	570
L-17	S0001	1000
L-23	S0002	2000
L-14	S0001	1500
L-93	S0004	500

7 filas seleccionadas.

Vista:

Site_a

SELECT * FROM total_prestamos_por_sucursal;



The screenshot shows a SQL query execution window. The top toolbar includes icons for running, saving, and other database operations, along with a timer showing 0.098 segundos. The main window is titled 'Hoja de Trabajo' and 'Generador de Consultas'. The query entered is 'SELECT * FROM total_prestamos_por_sucursal;'. Below the query, a 'Salida de Script' window shows the results of the query. The results are displayed in a table with two columns: 'IDSUC' and 'TOTAL_PRESTAMOS'. The table contains 6 rows of data. At the bottom, a status message indicates '6 filas seleccionadas.'

IDSUC	TOTAL_PRESTAMOS
S0009	570
S0001	2500
S0005	900
S0002	2000
S0004	500
S0007	7500

6 filas seleccionadas.