



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10

FC-FISC-1-8-2016)



TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

Facilitador(a): Ing. Carmen Ortega Asignatura: Base de Datos II

Fecha: 16/11/2020 Grupo: 1IL131

Estudiantes: Yui Lo

A. TÍTULO DE LA EXPERIENCIA: CAPITULO VI CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y CONTROL DE LA CONCURRENCIA

B. TEMAS:

1. Introducción a las transacciones y bloqueos
 - 1.1 Administración de las transacciones
 - 1.2 Bloqueos en SQL (según el gestor a usar)
 - 1.3 Administración de los bloqueos
2. Conceptos sobre Control de la Concurrencia
3. Control de la concurrencia con métodos de bloqueo
 - Granularidad
 - Tipos de bloqueo
 - Seriabilidad
 - Interbloqueos
4. Otros métodos de control de concurrencia

C. OBJETIVO(S):

Realizar investigación sobre conceptos de conceptos básicos sobre transacciones y control de la concurrencia y aplicarlo en ejemplos prácticos. La actividad le permite al estudiante enriquecer su experiencia en el desarrollo y construcción de conocimientos del curso de Base de Datos II.

D. METODOLOGÍA:

1. Divida el salón en grupos.
2. Distribuya el taller a cada grupo.
3. Cada equipo contará con un período de laboratorio (2 períodos de 45 min) para dar respuesta a las preguntas.
4. Primero realice el taller de forma individual y una vez desarrollado haga una reunión video llamada en equipo para que colaborativamente den respuesta en equipo las preguntas del ejercicio.
5. Se deben recibir en sección tareas de la plataforma de clases virtuales las respectivas asignaciones resuelta o realizar el test online del taller.

E. PROCEDIMIENTO O ENUNCIADO DE LA EXPERIENCIA:

- a. Revisar el material de la ppt Guía Taller – Capítulo VI suministrada por los equipos de trabajos del salón.



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10

FC-FISC-1-8-2016)



TALLER No 5

**CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA**

- b. Desarrolla los ejercicios dados ppt Guía Taller y colocar los resultados, respuestas en este documento Word en la sección G.
- c. Coloque evidencia de cada ejercicio realizado del taller con sus respuestas correspondientes.

NOTA: DESARROLLE SUS RESPUESTAS EN LA SECCION G.

F. RECURSOS:

Laptop o PC con los recursos, Internet, material de clases preparado por el docente.

G. RESULTADOS:

Caso 1

Ejercicio 1

```
17 • START TRANSACTION;
18 • INSERT INTO T (id, s) VALUES (4, 'fourth');
19 • SELECT * FROM T;
20 • ROLLBACK;
21 • SELECT * FROM T;
22
```

Result Grid

	id	s	si
▶	1	first	NULL
	2	second	NULL
	3	third	NULL
	4	fourth	NULL
*	NULL	NULL	NULL

El primer select muestra que la tabla T contiene 4 filas, mientras que el segundo select después de la sentencia ROLLBACK se deshace la inserción de la cuarta fila.

Ejercicio 2



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



```
23 • INSERT INTO T (id, s) VALUES (5, 'fifth');
24 • ROLLBACK;
25 • SELECT * FROM T;
26
```

< Filter Rows: Edit:

	id	s	si
▶	1	first	NULL
	2	second	NULL
	3	third	NULL
	5	fifth	NULL
*	NULL	NULL	NULL

Obtiene como resultado el valor insertado 5 y fifth, y la sentencia ROLLBACK no lo deshace. Podemos concluir que MySQL hace autocommit, que en cada transacción debemos iniciarlo mediante el comando “STARTTRANSACTION”.

Ejercicio 3

```
27 • SET AUTOCOMMIT = 0;
28 • DELETE FROM T WHERE id > 1;
29
30 • INSERT INTO T (id, s) VALUES (2, 'second');
31 • INSERT INTO T (id, s) VALUES (3, 'third');
32 • SELECT * FROM T;
33
34 • ROLLBACK;
35 • SELECT * FROM T;
36
```

< Filter Rows: Edit:

	id	s	si
▶	1	first	NULL
	2	second	NULL
	3	third	NULL
	5	fifth	NULL
*	NULL	NULL	NULL



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10

FC-FISC-1-8-2016)



TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

Para el uso de SET AUTOCOMMIT=0 se desactiva el Autocommit de MySQL para toda la sección, mientras que START TRANSACTION solo desactiva el Autocommit para la instrucción. Una de las ventajas de usar SET AUTOCOMMIT = 0 es que en cualquier momento puede retroceder con la sentencia ROLLBACK.

Ejercicio 4

```
37 • SET AUTOCOMMIT=0;
38 • INSERT INTO T (id, s) VALUES (9, '¿Esto se consolidará?');
39 • CREATE TABLE T2 (id INT);
40 • INSERT INTO T2 (id) VALUES (1);
41 • SELECT * FROM T2;
42 • ROLLBACK;
43
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
id			
1			

La tabla T si se guardó los datos insertado, mientras que la tabla T2 no se guardó los datos insertado. Podemos concluir que la sentencia ROLLBACK no deshace tabla creada, solo deshace los cambios realizadas a la tabla y que solo afectará las modificaciones realizada en la última tabla.

Ejercicio 5

```
57 • SET AUTOCOMMIT=0;
58 • INSERT INTO T (id, s) VALUES (2, 'La prueba de errores');
59 -- la división por cero debería fallar
60 • SELECT (1/0) AS dummy FROM DUAL;
61 -- Actualizamos una tupla inexistente
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
dummy			
NULL			

la división 1 entre 0 no es un error



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



```
66 • INSERT INTO T (id, s) VALUES (2, 'Esto es un duplicado');
67 • INSERT INTO T (id, s) VALUES (3, 'Probamos a introducir una cadena de
68 • INSERT INTO T (id, s, si) VALUES (4, 'Smallint overflow para 32769?',
69 • INSERT INTO T (id, s) VALUES (5, '¿Sigue active la transacción?');
70 • SELECT * FROM T;
71 • COMMIT;
```

Result Grid				Filter Rows:	Edit:	Export/Import:	Wr
id	s	si					
1	first	NULL					
2	La prueba de errores	NULL					
NULL	NULL	NULL					

El valor 2 es duplicado.

Si desactivamos el autocommit, cuando ocurre un error no se deshace los cambio realizada.

Caso 2



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10

FC-FISC-1-8-2016)



TALLER No 5

**CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA**

Conceptos	Resp.	Descripción
1. Tipo de primitiva de sincronización	9	Capacidad de los sistemas de administración de base de datos para ejecutar procesos al mismo tiempo.
2. Los protocolos	12	Aquellos algoritmos que están basados en acceso mutuamente exclusivo a datos compartidos
3. Pesimista y optimista	1	Clasificación más común de control de concurrencia
4. Pesimista	11	Actualización pérdida, dependencia no confirmada, análisis inconsistente
5. Optimista	8	Para evitar esta ejecución incorrecta, un sincronizador debe decidir rechazar "Escribir1" provocando que la transacción T1 sea cancelada.
6. Sincronizador	10	Multiprogramación, aplicaciones estructuradas.
7. Protocolos basados en bloqueos	13	Fases de las transacciones de tipo optimistas
8. Leer1(Cuentas[7]) devuelve el valor de \$1000 Leer2(Cuentas[7]) devuelve el valor de \$1000 Escribir2(Cuentas[7],\$10 1000) Commit2 Escribir1(Cuentas[7],\$1100) Commit1	6	Es un programa que controla la ejecución concurrente de las transacciones; su función radica en ordenar las operaciones que forman parte de las transacciones que se requieren ejecutar concurrentemente, de tal manera que la ejecución resultante sea correcta.
9. Concurrencia	3	Puntos de vista que utilizan las primitivas de sincronización en sus algoritmos
10. Casos de aplicación de concurrencia	4	Tan pronto como un usuario empieza a actualizar un registro, se coloca un bloqueo sobre el mismo.
11. Forma en que una transacción produce respuestas incorrectas.	5	Los algoritmos retrasan la sincronización de las transacciones hasta su terminación.
12. Los bloqueos	2	Aquellos algoritmos que intentar ordenar la ejecución de las transacciones de acuerdo con un conjunto de reglas
13. Inicio, modificar, validar, commit/rollback	7	Protocolos más utilizados por los DBMS comerciales

Caso 3

Se produce un interbloqueo cuando 2 procesos compiten por el acceso exclusivo a un recurso, pero no son capaces de poder obtener acceso exclusivo a él, porque el otro proceso lo impide. Esto da como resultado un enfrentamiento donde ninguno de los procesos puede continuar.

En este ejemplo, tenemos 2 tablas: Las facturas y InvoiceDetails. Construiremos estas tablas basadas en la base de datos AdventureWorks.



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



```
-- Invoices
SELECT *
INTO dbo.Invoices
FROM [AdventureWorks].[Sales].[SalesOrderHeader]
-- InvoiceDetails
SELECT *
INTO dbo.InvoiceDetails
FROM [AdventureWorks].[Sales].[SalesOrderDetail]
```

En la actualidad, la primera sesión, que representa al UserA en el ejemplo, se ejecutará a continuación de la instrucción correspondiente. Tenga en cuenta que no hay previsto ningún compromiso.

```
DECLARE @SalesOrderId BIGINT;
DECLARE @SumValue BIGINT;
SET @SalesOrderId = 43659;

--SELECT * From dbo.Invoices where SalesOrderId = @SalesOrderId ;

BEGIN TRAN

SELECT @SumValue = SUM(LineTotal) -- we will round up to next int
FROM dbo.InvoiceDetails
WHERE SalesOrderId = @SalesOrderId

UPDATE dbo.Invoices
SET
    SubTotal = @SumValue ,
    TaxAmt   = @SumValue * .21
WHERE SalesOrderId = @SalesOrderId
;
```

Podemos verificar que la transacción aún esté abierta, por lo que no se revierte ni se confirma mediante la siguiente consulta:

```
SELECT * FROM sys.sysprocesses WHERE open_tran = 1
```

Ahora, usted debe verificar que ejecutemos la siguiente instrucción en otra sesión la que desempeñará el papel de UserBen nuestro ejemplo.

```
DECLARE @SalesOrderId BIGINT;
DECLARE @SumValue BIGINT;
SET @SalesOrderId = 43659;

BEGIN TRAN

UPDATE dbo.InvoiceDetails SET LineTotal = 0 where SalesOrderId = @SalesO
rderId ;
SELECT * FROM dbo.Invoices where SalesOrderId = @SalesOrderId ;
```

Si ahora, ejecutamos la siguiente declaración en la primera sesión (UserA):



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



```
select * from dbo.InvoiceDetails;
```

Luego, cualquiera de las sesiones (UserA o UserB)) se marcará como interbloqueo. En mi caso, fue la sesión del usuario.

```
Msg 1205, Level 13, State 49, line 1  
Transaction (Process ID 59) was deadlocked on lock resources with another process and has been chosen as the deadlock victim. Rerun the transaction.
```

En el Caso de que, Si retrocedemos, nada debería haber cambiado para SalesOrderId #43659.

Caso 4

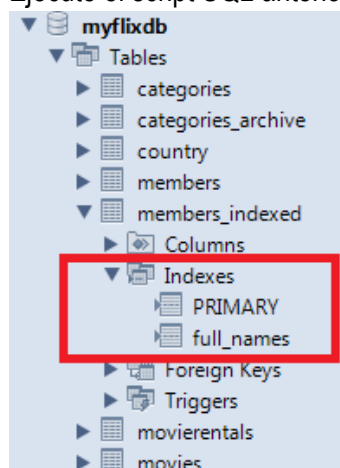
Para nuestro myflixdb, esperamos muchas búsquedas en la base de datos con el nombre completo.

Añadiremos la columna "full_names" a Index en una nueva tabla "members_indexed".

La secuencia de comandos que se muestra a continuación nos ayuda a lograr eso.

```
CREATE TABLE `members_indexed` (  
  `membership_number` int (11) NOT NULL AUTO_INCREMENT,  
  `full_names` varchar (150) DEFAULT NULL,  
  `gender` varchar (6) DEFAULT NULL,  
  `date_of_birth` date DEFAULT NULL,  
  `physical_address` varchar (255) DEFAULT NULL,  
  `postal_address` varchar (255) DEFAULT NULL,  
  `contact_number` varchar (75) DEFAULT NULL,  
  `email` varchar (255) DEFAULT NULL,  
  PRIMARY KEY (`membership_number`),  
  INDEX (full_names)  
) ENGINE = InnoDB;
```

Ejecute el script SQL anterior en MySQL workbench contra el "myflixdb".



Al actualizar myflixdb, se muestra la tabla recién creada llamada members_indexed.



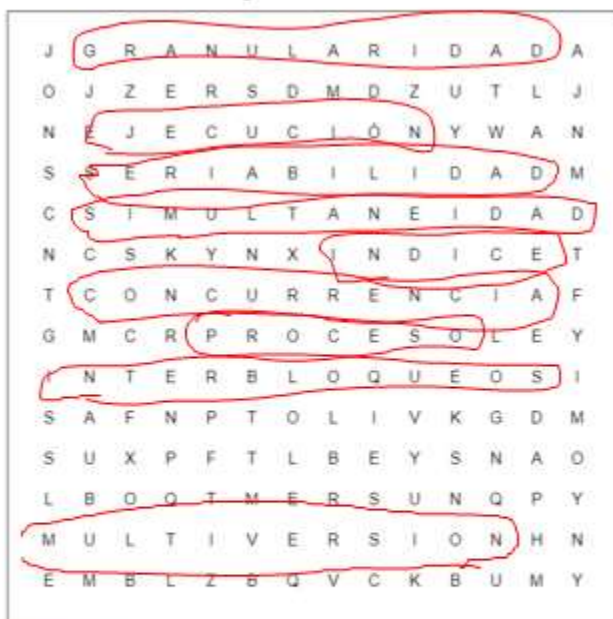
UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



A medida que la base de miembros se expande y aumenta el número de registros, las consultas de búsqueda en la tabla members_indexed que usan las cláusulas WHERE y ORDER BY serán mucho más rápidas en comparación con las realizadas en la tabla de miembros sin el índice definido.

Sopa de letras



www.educma.com

Concurrencia	Ejecución
Granularidad	Índice
Interbloques	Multiversión
Proceso	Seriabilidad
Simultaneidad	

H. CONSIDERACIONES FINALES – CONCLUSIONES:

En este taller pude aprender y aplicar los conocimientos que se obtuvo en la presentación de mis compañeros sobre las transacciones y bloqueo en MySQL y control de concurrencia, estos ayuda a mantener los datos ordenado para un funcionamiento óptimo en la base de datos.

BIBLIOGRAFIA:



UNIVERSIDAD TECNOLÓGICA DE PANAMÁ
FACULTAD DE INGENIERÍA DE SISTEMAS COMPUTACIONALES
SISTEMAS DE INFORMACION
GUÍA DE ACTIVIDAD N° 10
TALLER No 5
CAPÍTULO VI -CONCEPTOS BÁSICOS SOBRE TRANSACCIONES Y
CONTROL DE LA CONCURRENCIA

FC-FISC-1-8-2016)



RÚBRICAS:

<i>Detalle - Evaluar</i>	<i>Valor de la respuesta</i>
<i>Sección G – (100 PUNTOS)</i>	<i>100 pts</i>