

# Administración de Base de Datos

Guía de Trabajo  
*Administración de Base de Datos*

Primera edición digital  
Huancayo, 2022

De esta edición

© Universidad Continental, Oficina de Gestión Curricular  
Av. San Carlos 1795, Huancayo-Perú  
Teléfono: (51 64) 481-430 anexo 7361  
Correo electrónico: [recursosucvirtual@continental.edu.pe](mailto:recursosucvirtual@continental.edu.pe)  
<http://www.continental.edu.pe/>

**Cuidado de edición**

Fondo Editorial

**Diseño y diagramación**

Fondo Editorial

Todos los derechos reservados.

La *Guía de Trabajo*, recurso educativo editado por la Oficina de Gestión Curricular, puede ser impresa para fines de estudio.

# Contenido

---

Presentación	5
<b>Primera Unidad</b>	<b>7</b>
Práctica 1	
Implementación de base de datos	8
Práctica 2	
Índices	11
Práctica 3	
Disparadores	20
<b>Segunda Unidad</b>	<b>25</b>
Práctica 4	
Automatización de trabajos	26
Práctica 5	
Automatización de trabajos	30
Práctica 6	
Monitoreo y rendimiento	31
<b>Tercera Unidad</b>	<b>35</b>
Práctica 7	
Transacciones y bloqueos	36
Práctica 8	
Inicios de sesión y usuarios	41
Práctica 9	
Copias de seguridad	44

<b>Cuarta Unidad</b>	<b>47</b>
Práctica 10	
Migración de datos	48
Práctica 11	
Bases de datos distribuidas	49
Práctica 12	
Auditoría de bases de datos	50
Referencias	57

# Presentación

---

La presente guía, brinda al estudiante de la asignatura Administración de Base de Datos, los casos y ejercicios prácticos de laboratorio, usando los conocimientos, metodología, técnicas y herramientas instruidas que le permitirán realizar una adecuada administración de bases de datos en las organizaciones.





## Primera Unidad



## Implementación de base de datos

---

### 1. Práctica de laboratorio

- Para la práctica, utilizar dos discos duros (o dos particiones). Por ejemplo, "C:\BD" y "D:\BD".
- Restaurar la base de datos '**Comercial**'. Utilizar el Management Studio. Restaurarla en la carpeta "C:\BD".
- Incrementar el tamaño del archivo de datos, y del archivo de registro de transacciones, ambos a 30 Mb. Luego comprobar físicamente el tamaño de ambos archivos y también en la ficha de propiedades.

```
ALTER DATABASE Comercial
MODIFY FILE
(NAME = Comercial,
SIZE = 30,
MAXSIZE = 30)
```

```
ALTER DATABASE Comercial
MODIFY FILE
(NAME = Comercial_log,
SIZE = 30,
MAXSIZE = 30)
```

- Agregar un segundo archivo de base de datos para la BD **Comercial** en la carpeta "D:\BD":

```
ALTER DATABASE Comercial
ADD FILE
(NAME = Comercial1,
FILENAME = 'D:\BD\Comercial1.ndf',
SIZE = 2,
MAXSIZE = 10,
FILEGROWTH = 2)
```

- e) Reducir la base de datos, para ahorrar espacio en disco. Luego comprobar el tamaño físico de los archivos.

**DBCC SHRINKDATABASE (Comercial)**

- f) Separar la base de datos Comercial. Apuntar a la base de datos **Master**. Asegurarse que no haya conexiones abiertas.

**EXEC sp\_detach\_db 'Comercial'**

- g) Mover los archivos de la base de datos **Comercial**, a la unidad "D:\BD". Desde esa nueva ubicación, adjuntar la base de datos nuevamente al servidor.

**EXEC sp\_attach\_db 'Comercial',  
'D:\BD\Comercial.mdf',  
'D:\BD\Comercial\_log.ldf',  
'D:\BD\Comercial1.ndf'**

- h) Restaurar la base de datos **'Farmacia'**. Utilizar el Management Studio. Restaurarla en la carpeta "C:\BD".

- i) Incrementar el tamaño del archivo de datos, y del archivo de registro de transacciones, ambos a 100 Mb. Luego comprobar físicamente el tamaño de ambos archivos y también en la ficha de propiedades.

- j) Agregar un segundo archivo de base de datos para la BD Farmacia en la carpeta "D:\BD":

- k) Reducir la base de datos, para ahorrar espacio en disco. Luego comprobar el tamaño físico de los archivos.

- l) Separar la base de datos **Farmacia**. Apuntar a la base de datos **Master**. Asegurarse que no haya conexiones abiertas.



- m) Mover los archivos de la base de datos **Farmacia**, a la unidad "D:\BD". Desde esa nueva ubicación, adjuntar la base de datos nuevamente al servidor.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Práctica 2

### Índices

---

#### 1. Práctica de laboratorio

- Para la práctica, crear y utilizar la carpeta "C:\BD".
- Restaurar la base de datos '**ClassNorthwind.bak**'. Utilizar el Management Studio. Restaurarla en la carpeta "C:\BD".
- Crear un índice no agrupado llamado OrdersCustomerslink en la columna CustomerID de la tabla Orders con un valor de FILLFACTOR de 75.

```
CREATE NONCLUSTERED INDEX Orders_Customers_link ON Orders(CustomerID)
WITH FILLFACTOR = 75
GO
```

- Comprobar en la ventana *browser* del Management Studio, que se haya creado el índice.
- Crear índices agrupados y no agrupados para todas las referencias de clave foránea a la tabla **Products**, y comprobar su creación mediante comandos.

```
CREATE CLUSTERED INDEX Products_CategoryID_link ON Products(CategoryID)
CREATE NONCLUSTERED INDEX Products_SupplierID_link ON Products(SupplierID)
GO
```

```
PRINT 'CREATED INDEXES:'
SELECT name
FROM sysindexes
WHERE name IN ('Products_CategoryID_link', 'Products_SupplierID_link')
GO
```

- Restaurar la base de datos '**CreditDB.bak**'. Utilizar el Management Studio. Restaurarla en la carpeta "C:\BD".



- g) Obtener información estadística de la tabla **member** de la base de datos **credit**.

**USE credit**

**GO**

**EXEC sp\_spaceused member**

**SELECT \* FROM sysindexes WHERE id = OBJECT\_ID('member')**

**DBCC SHOWCONTIG ('member')**

La información a recolectar debe ser (revisar en la pestaña mensajes):

Información	Origen	Resultado
Número de filas.	sp_spaceused: filas	10.000
Número de índices.	SELECT * FROM sysindexes WHERE id = OBJECT_ID('member')	Ninguna. Una fila en sysindexes con un valor de 0 en <i>indid</i> representa la tabla propiamente dicha.
Número de páginas.	SHOWCONTIG: Páginas recorridas.	145
Número de filas por página.	Cálculo y redondeo de los resultados. (n° de filas/n° de páginas) = n° de filas por página.	69
Número de extensiones.	SHOWCONTIG: Cambios de extensión.	18
Promedio de relleno de extensión.	SHOWCONTIG: Promedio de páginas por extensión.	7,6
Promedio de relleno de página.	SHOWCONTIG: Promedio de densidad de página (completa).	98.82%

- h) Crear un índice agrupado único y observará los cambios en la estructura de la tabla. También obtener información acerca de la estructura del índice.

**USE credit**

**CREATE UNIQUE CLUSTERED INDEX mem\_no\_CL  
ON member (member\_no)**

**USE credit**

**SELECT \* FROM sysindexes WHERE id = OBJECT\_ID('member')  
DBCC SHOWCONTIG ('member')**

Información	Origen	Resultado
Número de páginas del índice agrupado.	fila de sysindexes: used	147
Número de páginas de datos en el índice agrupado.	fila de sysindexes: dpages	145
Número de páginas sin datos en el índice agrupado.	(used - dpages)	147-145 = 2
Número de índices.	SELECT * FROM sysindexes	Uno. Una fila en sysindexes con un valor de 1 en <i>indid</i> representa el índice agrupado.
Número de páginas.	SHOWCONTIG: Páginas recorridas.	145
Número de filas por página.	Cálculo y redondeo de los resultados. (n° de filas/ n° de páginas) = n° de filas por página.	69
Número de extensiones.	SHOWCONTIG: Cambios de extensión.	18
Promedio de relleno de extensión.	SHOWCONTIG: Promedio de páginas por extensión.	7,6
Promedio de relleno de página.	SHOWCONTIG: Promedio de densidad de página (completa).	98.82%

- i) En este procedimiento, crear un índice no agrupado único y obtener información acerca de la estructura del índice. Escribir y ejecutar esta instrucción para eliminar el índice creado antes.

```
USE credit  
EXEC index_cleanup member
```

- j) Escribir y ejecutar esta instrucción para crear un índice no agrupado con la columna **firstname** de la tabla **Member**, sin especificar un valor de FILLFACTOR:

```
USE credit  
CREATE NONCLUSTERED INDEX indx_fname  
ON member(firstname)
```

```
USE credit  
SELECT * FROM sysindexes WHERE id = OBJECT_ID('member')
```

Información	Origen	Resultado
Número de páginas del Índice no agrupado en la columna <i>firstname</i> .	fila de sysindexes: used	39
Número de páginas en el nivel de hoja.	fila de sysindexes: dpages	37
Número aproximado de filas por página de hoja.	(n° de filas en la tabla/n° de páginas de nivel de hoja)	(10000 / 37) = 270

- k) En este procedimiento, crear un índice no agrupado y observar los cambios en la estructura de la tabla. Escribir y ejecutar esta instrucción para eliminar el índice no agrupado de la tabla Member:

```
USE credit  
EXEC index_cleanup member
```

- l) Escribir y ejecutar esta instrucción para crear el mismo índice, pero ahora con un valor de FILLFACTOR del 25 por ciento:

```
USE credit
CREATE NONCLUSTERED INDEX indx_fname
ON member(firstname)
WITH FILLFACTOR=25
```

- m) Escribir y ejecutar esta instrucción SELECT que devuelve las filas de sysindexes de la tabla Member:

```
USE credit
SELECT * FROM sysindexes WHERE id = OBJECT_ID('member')
```

Información	Origen	Resultado
Número de páginas en este índice.	fila de sysindexes: used	293
Número de páginas en el nivel de hoja.	fila de sysindexes: dpages	145
Número aproximado de filas por página de hoja.	(n° de filas en la tabla/n° de páginas de nivel de hoja)	(10000 / 145) = 70

**Nota:** Con un valor de FILLFACTOR de 0 (predeterminado), pueden caber 270 filas en cada página de nivel de hoja. Multiplicar 270 por el 25 por ciento y el resultado es aproximadamente 70 filas. Este es el número de filas que cabrán en una página de nivel de hoja que solo está llena un 25 por ciento.

- n) En este ejercicio crear varios índices con la tabla **Member**, obtener sus estadísticas y evaluar si un índice es útil para el optimizador de consultas según su selectividad. Eliminar los índices existentes de la tabla **Member**:

```
USE credit
EXEC index_cleanup member
```



- o) Escribir y ejecutar estas instrucciones para crear tres índices basados en la tabla **Member**:

**USE credit**

```
CREATE UNIQUE INDEX indx_member_no ON member (member_no)
CREATE INDEX indx_corp_lname ON member (corp_no,lastname)
CREATE INDEX indx_lastname ON member (lastname)
GO
```

- p) En este procedimiento obtener estadísticas de los índices nuevos, registrar la información estadística y evaluar la selectividad de los índices. Escribir y ejecutar esta instrucción para mostrar la información estadística de índices acerca de la columna **member\_no** de la tabla **Member**:

**USE credit**

```
DBCC SHOW_STATISTICS (member,indx_member_no)
```

Información	Resultado
Rows	10.000
Steps	3
Density	1
All density	0.0001

¿Qué selectividad tiene el índice de la columna *member\_no*?

Es muy selectivo. El índice *indx\_member\_no* se crea con la columna *member\_no*, que contiene valores únicos. Si una consulta especifica un número de miembro en la cláusula WHERE con una igualdad, solo se devolverá una fila.

- q) Escribir y ejecutar esta instrucción para mostrar la información estadística de índices de un índice compuesto basado en las columnas *corp\_no* y *lastname* de la tabla **Member**:

**USE credit**

```
DBCC SHOW_STATISTICS (member,indx_corp_lname)
```

Información	Resultado
Rows	10,000
Steps	201
Density	0.3257
All density (corp_no)	0.0024
All density (corp_no, lastname)	0.0006
All density	0.0001

**¿Qué selectividad tiene este Índice?**

Sigue siendo selectivo, pero no tanto como la columna *member\_no*, ya que existen valores duplicados.

- r) Escribir y ejecutar esta instrucción para mostrar la información estadística de índices acerca de la columna **lastname** de la tabla **Member**:

**USE credit**

**DBCC SHOW\_STATISTICS (member,indx\_lastname)**

Información	Resultado
Rows	10,000
Steps	26
Density	0
All density	0.0384

**¿Qué selectividad tiene este Índice?**

Este índice no es muy selectivo. El uso de la fórmula para calcular la densidad muestra que este índice tiene baja selectividad. Debido a que no puede dividir por cero, el resultado es cero. All density indica la densidad total.

$$((26/0)/10000) = 0$$

- s) Restaurar la base de datos "Supermercado.bak"
- t) Crear un índice no agrupado para el campo NombreArticulo de la tabla Artículo.

**CREATE NONCLUSTERED INDEX IDXNombreArticulo  
ON Artículo (NombreArticulo)**



- u) Crear un índice con valores únicos, no agrupado, para el campo *FechaEmision* de la tabla *DocVenta*.

**CREATE UNIQUE NONCLUSTERED INDEX IDXFechaEmision  
ON DocVenta (Fecha)**

- v) Crear un índice con valores no únicos, no agrupado, de múltiples campos, para los campos *ApPatContacto* y *ApMatContacto*, de la tabla *Cientes*.

**CREATE NONCLUSTERED INDEX IDXContacto  
ON Cliente (ApPatContacto,ApMatContacto)**

- w) Visualizar y registrar en una tabla, la información y estadísticas más importantes de las tablas e índices creados. Escribir un análisis cualitativo del significado de la información encontrada por cada índice.

**SELECT \* FROM sysindexes WHERE id = OBJECT\_ID('articulo')  
DBCC SHOW\_STATISTICS (articulo,idxnombrearticulo)  
DBCC SHOWCONTIG ('articulo')**

Nota: Falta extraer la información para las tablas *DocVenta* y *Cliente*.

- x) Restaurar la base de datos "Farmacia.bak"
- y) Crear un índice no agrupado para el campo *Descripción* de la tabla *Artículos*.
- z) Crear un índice con valores únicos no agrupado, para el campo *Fecha* de la tabla *Ventas*.
- aa) Crear un índice con valores no únicos, no agrupado, de múltiples campos, para los campos *Apellidos* y *Nombres*, de la tabla *Cientes*.
- bb) Visualizar y registrar en una tabla, la información y estadísticas más importantes de las tablas e índices creados. Escribir un análisis cualitativo del significado de la información encontrada por cada índice.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



**1. Práctica de laboratorio**

- a) Restaurar la base de datos proporcionada "Supermercado.bak".
- b) El siguiente disparador alerta si se ha agregado algún registro en la tabla Cliente:

```
CREATE TRIGGER SeAgregoCliente
ON Cliente
FOR INSERT
AS
IF UPDATE(IDCliente)
PRINT 'Se agregó un nuevo cliente... Trigger disparado'
```

Probar el disparador con la siguiente instrucción:

```
INSERT INTO Cliente (IDCliente, RazonSocial, ApPatContacto,
ApMatContacto, NombreContacto)
VALUES ('JPF','Restaurante El Remanso','Parker','Ferrer','José')
```

- c) Analizar y ejecutar el siguiente disparador que no permita ingresar un Artículo con precio negativo:

```
CREATE TRIGGER VerificaPrecio
ON Articulo
FOR INSERT
AS
IF (SELECT PrecioUnidad FROM Inserted)<0
BEGIN
PRINT 'No se permite precios negativos... Tigger Disparado...'
ROLLBACK
END
```

Probar el disparador con la siguiente instrucción:

```
INSERT INTO Articulo(IDArticulo,NombreArticulo,UnidExis-  
tencia,PrecioUnidad,UnidMinimo)  
VALUES(114,'Spaguetti',50,-5,10)
```

- d) Crear un disparador que no permita modificar las Descripciones almacenadas en la tabla unidad.

```
CREATE TRIGGER NoCambiosUnidad  
ON Unidad  
FOR UPDATE  
AS  
IF UPDATE(Descripcion)  
BEGIN  
PRINT 'No se deben cambiar las descripciones... Trigger Disparado...'  
ROLLBACK  
END
```

Probar el disparador con la siguiente instrucción:

```
UPDATE Unidad  
SET Descripcion='Carnes/Embutidos'  
WHERE IDUnidad=6
```

- e) Restaurar la base de datos ClassNorthwind desde el archivo Northwind.bak
- f) Este desencadenador actualiza la columna UnitsInStock de la tabla Products cuando se inserta una fila en la tabla Order Details (cuando se recibe un pedido).

```
CREATE TRIGGER OrdDet_Insert  
ON [Order Details]  
FOR INSERT  
AS  
UPDATE P SET  
UnitsInStock = (P.UnitsInStock - I.Quantity)  
FROM Products AS P INNER JOIN Inserted AS I  
ON P.ProductID = I.ProductID  
GO
```



- g) Ejecutar la instrucción SELECT siguiente para seleccionar una fila de la tabla Products y determinar las unidades de un producto que hay en almacén:

```
SELECT * FROM Products WHERE ProductID = 22
```

La columna UnitsInStock debe contener el valor de 104. Si la columna UnitsInStock muestra un número diferente, anótelos.

- h) Insertar una fila en la tabla Order Details que pida 50 unidades del producto:

```
INSERT [Order Details] (OrderID, ProductID, UnitPrice, Quantity, Discount)  
VALUES (11077, 22, 21.00, 50, 0.0)
```

- i) Hacer una consulta en la tabla Products para comprobar que el valor de la columna UnitsInStock para el ProductID determinado ha cambiado a 54.

- j) En este ejercicio crear dos tablas nuevas denominadas NewCategories y NewProducts. Ninguna de las dos tiene las restricciones PRIMARY y FOREIGN KEY de las tablas Categories y Products. Crear un desencadenador en la tabla NewCategories. Este desencadenador actualiza la columna Discontinued de la tabla NewProducts cuando se elimina una categoría (cuando se elimina un registro de la tabla NewCategories). Todos los productos afectados se marcan con 1, lo que indica que ya no se suministran.

- k) Escribir y ejecutar la consulta siguiente para crear dos tablas nuevas denominadas NewCategories y NewProducts:

```
USE ClassNorthwind  
GO  
--Crea una tabla NewCategories  
SELECT * INTO NewCategories FROM Categories  
--Crea una tabla NewProducts  
SELECT * INTO NewProducts FROM Products  
GO
```

- l) Escribir y ejecutar la consulta siguiente para crear un desencadenador en la tabla `NewCategories`. Este desencadenador actualiza la columna `Discontinued` de la tabla `NewProducts` a 1 cuando se elimina la categoría principal de un producto (cuando se elimina una fila de la tabla `NewCategories`).

```
CREATE TRIGGER Category_Delete
ON NewCategories
FOR DELETE
AS
UPDATE P SET Discontinued = 1
FROM NewProducts AS P INNER JOIN Deleted AS d
ON P.CategoryID = D.CategoryID
```

- m) Escribir y ejecutar la siguiente instrucción `SELECT` que consulta la tabla `NewProducts` para determinar si se siguen suministrando los productos con `CategoryID` "7".

```
SELECT ProductID, CategoryID, Discontinued
FROM NewProducts WHERE CategoryID = 7
```

- n) Escribir una instrucción `DELETE` que quite una fila de la tabla `NewCategories` y comprobar que el desencadenador se ejecuta correctamente.

```
DELETE NewCategories WHERE CategoryID = 7
```

- o) Restaurar la base de datos `Farmacia` desde el archivo `FARMACIA.BAK`.
- p) Crear y ejecutar un disparador que alerte si se ha agregado algún registro en la tabla `Cliente`.
- q) Crear y ejecutar un disparador que alerte si se ha ingresado un `Artículo` con stock mayor de 100 unidades.
- r) Crear y ejecutar un disparador que no permita ingresar un `Artículo` con stock mínimo menor de 10 unidades.



- s) Crear y ejecutar un disparador que no permita modificar el stock de un artículo.
- t) Crear y ejecutar un disparador que no permita eliminar un cliente.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Segunda Unidad



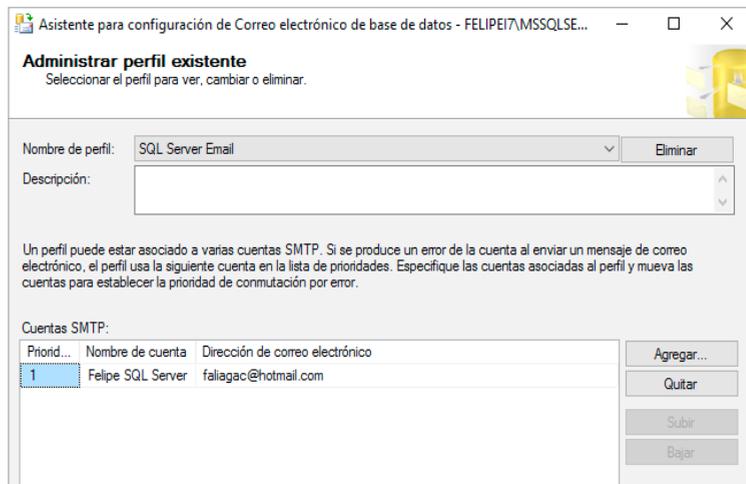
## Práctica 4

### Automatización de trabajos

---

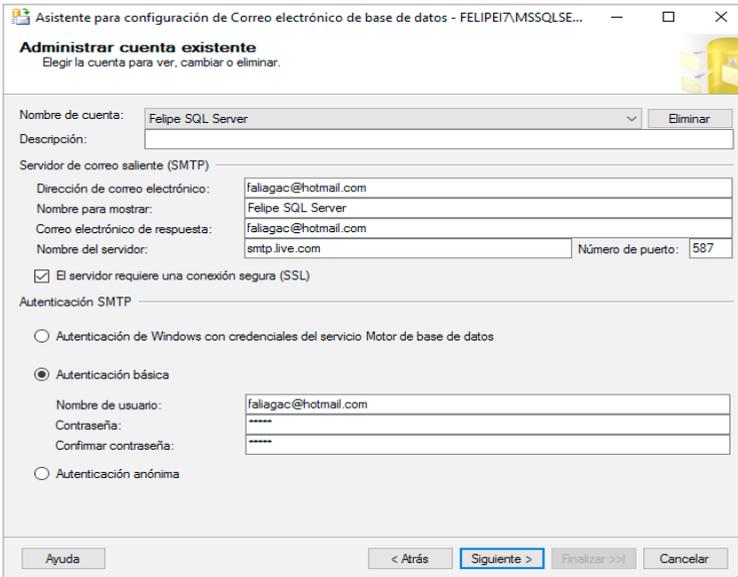
#### 1. Práctica de laboratorio

- a) Utilizando la herramienta de SQL Server, Correo electrónico de Bases de Datos, configurar un perfil nuevo de correo electrónico, para SQL Server:

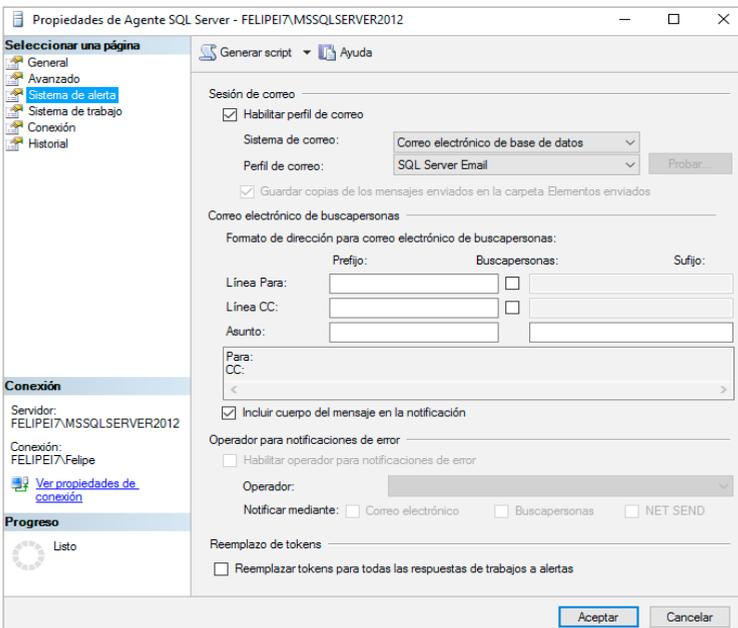


- b) Utilizando la herramienta de SQL Server, Correo electrónico de Bases de Datos, agregar una cuenta de correo electrónico (de su uso personal), en el perfil creado anteriormente:

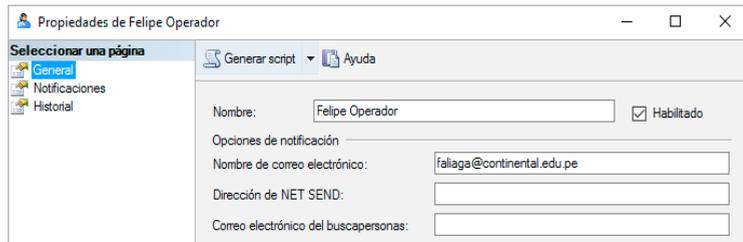
*Ver muestra en la siguiente página.*



c) En las propiedades del Agente SQL Server, configurar el uso del correo electrónico, creado en los pasos anteriores. Seleccionar el perfil creado.



- d) Restaurar la base de datos Supermercado.bak para realizar pruebas.
- e) Crear en el Agente de SQL Server, un operador que reciba las notificaciones de correo electrónico. Debe utilizar su propio correo electrónico de Continental.



- f) Crear en el Agente de SQL Server, un trabajo que liste los clientes de la base de datos supermercado, y establecer una notificación vía e-mail al operador antes creado, en cuanto el trabajo termine.
- g) Ejecutar el trabajo y comprobar si llegó el e-mail de notificación al operador; vale decir a su cuenta de Continental.
- h) Crear en el Agente de SQL Server, un trabajo que extraiga un *backup* de la base de datos supermercado, y crear una notificación vía e-mail al operador antes creado, en cuanto el trabajo termine. El paso incluido en el trabajo debe ser:
 

```
BACKUP DATABASE Supermercado to disk='c:\bd\super.bak' WITH INIT
```
- i) Ejecutar el trabajo y comprobar si llegó el *e-mail* de notificación al operador; vale decir a su cuenta de Continental. Comprobar también si se extrajo el *backup* respectivo.
- j) Crear en el Agente de SQL Server, una alerta que notifique vía e-mail al operador antes creado, una alerta de tipo (14-Permisos insuficientes).
- k) Probar la alerta tratando con el siguiente comando:

```
raiserror('Este es un error simulado de SQL Server',14,1) with log
```

- l) Verificar que haya llegado *e-mail* de la alerta.
- m) Automatizar el trabajo de *backup* del ejercicio (h), para que se repita diariamente a una hora determinada.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Práctica 5

### Automatización de trabajos

---

#### 1. Práctica de laboratorio

- a) Utilizando la herramienta de SQL Server, Correo electrónico de Bases de Datos, configurar un perfil nuevo de correo electrónico, para SQL Server.
- b) Utilizando la herramienta de SQL Server, Correo electrónico de Bases de Datos, agregar una cuenta de correo electrónico (de su uso personal de Gmail u Hotmail), en el perfil creado anteriormente.
- c) En las propiedades del Agente SQL Server, configurar el uso del correo electrónico, creado en los pasos anteriores. Seleccionar el perfil creado.
- d) Restaurar la base de datos Farmacia.bak.
- e) Crear en el Agente de SQL Server, un operador que reciba las notificaciones de correo electrónico. Debe utilizar el correo del profesor (**faliagac@hotmail.com**).
- f) Crear en el Agente de SQL Server, un trabajo que extraiga un backup de la base de datos farmacia, y crear una notificación vía e-mail al operador antes creado, en cuanto el trabajo termine.
- g) Ejecutar el trabajo. El profesor calificará el presente laboratorio, de acuerdo al correo electrónico recibido.

#### 2. Fecha de entrega

..... / ..... / .....

#### 3. Forma de entrega

- El trabajo es individual.
- El profesor calificará el presente trabajo, de acuerdo al correo electrónico recibido.

## Práctica 6

### Monitoreo y rendimiento

#### 1. Práctica de laboratorio

- a) Restaurar la base de datos "Northwindcopy.bak" en el servidor SQL Server.
- b) Copiar la carpeta "\Monitor" a la unidad C:\
- c) Abrir la herramienta "Monitor de Rendimiento" (de Windows), y agregar los siguientes contadores:

Contador	Instancia	Objeto
Lecturas de pagina/ s		SQLServer: Administrador de buffer
Transacciones activas	NorthWindCopy	SQLServer: Databases
Porcentaje de registro usado	NorthWindCopy	SQLServer: Databases
Transacciones/s	NorthWindCopy	SQLServer: Databases
Procesos bloqueados	NorthWindCopy	SQLServer: General Statistics
Transacciones		SQLServer: General Statistics
Búsquedas de índices/s		SQLServer: Métodos de acceso

- d) Observar el valor que reflejan los contadores. (Capturar la pantalla del monitor de rendimiento).
- e) Desde una ventana de comandos, ejecutar el archivo "Monitor.bat" de la carpeta "C:\Monitor". Este conjunto de archivos en lotes, simularán actividades de los usuarios en el servidor SQL Server, en la base de datos "NorthwindCopy".
- f) Volver a observar el valor que reflejan ahora los contadores. ¿Qué tendencias se observan? (Capturar la pantalla del monitor de rendimiento).
- g) Cerrar la herramienta Monitor de rendimiento, para eliminar los contadores seleccionados.
- h) Abrir la herramienta "Monitor de Rendimiento" (de Windows), y agregar los siguientes contadores:



Contador	Instancia	Objeto
Errores de página/s		Memoria
Páginas/s	Total	Memoria
Mbytes disponibles	Total	Memoria
% de tiempo de procesador		Proceso
Errores de página/s		Proceso
Memoria de conexión (KB)		SQLServer: Memory Manager
Memoria total del servidor (KB)		SQLServer: Memory Manager

- i) Observar el valor que reflejan ahora los contadores. ¿Qué tendencias se observan? - (Capturar la pantalla del monitor de rendimiento).
- j) Iniciar el monitor de actividad del servidor SQL Server.
- k) Visualizar e interpretar la información de los rubros del monitor de actividad: (Capturar las pantallas del monitor de actividad).

#### Información general

#### Procesos

#### Esperas de recursos

#### E/S de archivo de datos

#### Consultas costosas recientes

- l) Visualizar e interpretar los informes estándar de la base de datos "Northwindcopy": (Capturar las pantallas de cada reporte).

*Ver muestra en la siguiente página.*

Uso de disco  
Uso de disco por tablas superiores  
Uso de disco por tabla  
Uso de disco por partición  
Eventos de copias de seguridad y restauración  
Todas las transacciones  
Todas las transacciones de bloqueo  
Principales transacciones por edad  
Principales transacciones por recuento de transacciones bloqueadas  
Principales transacciones por recuento de bloqueos  
Estadísticas de bloqueo de recursos por objetos  
Estadísticas de ejecución de objetos  
Historial de coherencia de la base de datos  
Uso de memoria por objetos con optimización para memoria  
Estadísticas de uso de índice  
Estadísticas físicas del índice  
Historial de cambios de esquema  
Estadísticas de usuario

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.





## Tercera Unidad



## Transacciones y bloqueos

---

### 1. Práctica de laboratorio

- Restaurar la base de datos proporcionada "Northwind.bak".
- Ejecutar el siguiente Script:

```
BEGIN TRAN
PRINT 'trancount value:'
PRINT @@trancount
PRINT 'Before update:'
SELECT ContactName FROM Customers WHERE CustomerID = 'GREAL'
UPDATE Customers SET ContactName = 'Howard Snyder_Updated' WHERE CustomerID = 'GREAL'
PRINT 'After update:'
SELECT ContactName FROM Customers WHERE CustomerID = 'GREAL'
PRINT 'trancount value:'
PRINT @@trancount
-- COMMIT TRANSACTION
```

- En este momento, ¿se han confirmado los cambios efectuados con la instrucción UPDATE en esta transacción? ¿Cómo se puede determinar esto?

No, la transacción se debe completar con la instrucción COMMIT TRAN. La transacción sigue estando activa y sigue manteniendo los bloqueos que haya adquirido. El valor de la variable global @@trancount es 1, lo que indica que en esta sesión se ha enviado una instrucción BEGIN TRAN.

- ¿Podrían otras transacciones consultar o actualizar los datos modificados?

Otras transacciones no podrían consultar o modificar los datos modificados hasta que se haya confirmado (o deshecho) la transacción.

- c) Escribir una instrucción COMMIT TRANSACTION en la ventana de consulta y, después, resáltela y ejecútela para completar la transacción y hacer que los cambios sean definitivos.
- d) Resaltar y ejecutar una de las instrucciones SELECT de la tabla Customers para comprobar que ahora sí se ha completado el cambio.
- e) Ejecutar el siguiente *script*:

```
BEGIN TRAN
PRINT 'trancount value:'
PRINT @@trancount
PRINT 'Before update:'
SELECT ContactName FROM Customers WHERE CustomerID = 'GREAL'
UPDATE Customers SET ContactName = 'Howard Snyder' WHERE CustomerID = 'GREAL'
PRINT 'After update:'
SELECT ContactName FROM Customers WHERE CustomerID = 'GREAL'
PRINT 'trancount value:'
PRINT @@trancount
```

```
ROLLBACK TRAN
```

```
Print 'ROLLBACK issued'
SELECT ContactName FROM Customers WHERE CustomerID = 'GREAL'
PRINT 'trancount value:'
PRINT @@trancount
```

- ¿Se ha almacenado de forma permanente en la base de datos la modificación efectuada por la instrucción UPDATE?

No. La transacción se ha cancelado, de modo que los cambios efectuados durante la transacción se han deshecho.

- ¿Se ha completado la transacción?

Sí. La instrucción ROLLBACK TRAN completa la transacción y libera los bloqueos adquiridos por la transacción. La consulta de la variable global @@trancount devuelve cero.

- f) Cerrar los *scripts* anteriores, y abrir dos nuevas consultas, apuntando a la base de datos Northwind2. En el primer *script*, ejecutar el procedimiento almacenado de sistema:

```
exec sp_lock
```

Describir los bloqueos activos.

- g) En el segundo *script*, ejecutar lo siguiente y revisar los resultados:

```
BEGIN TRAN
UPDATE Customers
SET ContactName = 'Howard Snyder'
WHERE CustomerID = 'GREAL'
-- For the purpose of the exercise, COMMIT TRAN or ROLLBACK TRAN are not used.

PRINT 'Server Process ID (spid)'
PRINT @@spid
-- Use the SPID to identify the connection when using sp_lock.
```

- h) En el primer *script*, ejecutar el procedimiento almacenado de sistema:

```
exec sp_lock
```

Describir los bloqueos activos actuales, y explicar con los bloqueos que existían en el ejercicio 6.

- i) En el segundo *script*, ejecutar ROLLBACK TRAN.  
 j) En el primer *script*, ejecutar el procedimiento almacenado de sistema y revisar los bloqueos liberados.

```
exec sp_lock
```

Describir los bloqueos activos.

- l) En el segundo *script*, ejecutar lo siguiente y revisar los resultados:

```

BEGIN TRAN
  SELECT * FROM Customers(TABLOCKX) WHERE CustomerID = 'GREAL'
-- TABLOCKX bloquea la tabla Customers
-- For the purpose of the exercise, COMMIT TRAN or ROLLBACK TRAN are not used.

PRINT 'Server Process ID (spid)'
PRINT @@spid
-- Use the SPID to identify the connection when using sp_lock.

```

m) En el tercer *script*, ejecutar lo siguiente y revisar los resultados:

```

BEGIN TRAN
  UPDATE Customers
  SET ContactName = 'Howard Snyder'
  WHERE CustomerID = 'GREAL'
-- For the purpose of the exercise, COMMIT TRAN or ROLLBACK TRAN are not used.

PRINT 'Server Process ID (spid)'
PRINT @@spid

```

- ¿Hay una transacción que no se puede ejecutar? Si es así, ¿por qué?

Sí, la segunda transacción está intentando actualizar los datos que están bloqueados por la primera transacción, de modo que la segunda transacción debe esperar a que termine la primera. Puede determinar si una transacción no se puede ejecutar debido a un conflicto de bloqueo; para ello, utilice el procedimiento almacenado de sistema `sp_lock` y busque la palabra `WAIT` en la columna de estado. En este caso, la transacción de la conexión 3 debe esperar a que termine la transacción de la conexión 2.

- ¿Cuánto tiempo espera una transacción un recurso bloqueado?

Una transacción espera indefinidamente un recurso bloqueado, siempre y cuando no haya interbloqueos. Se pue-

de establecer el período de tiempo de espera de la sesión para controlar cuánto tiempo esperará SQL Server a que se liberen los recursos bloqueados.

n) En el segundo *script*, cancelar la transacción:

```
ROLLBACK TRAN
```

o) En el primer *script*, ejecutar el procedimiento almacenado de sistema:

```
exec sp_lock
```

Revise si el bloqueo ha sido desactivado y si se ejecutó el tercer *script*.

p) Restaurar la base de datos supermercado. Crear un *script* donde se bloquee la tabla **Cliente**. Crear un segundo *script* donde se trate de modificar la tabla Cliente, pero no se pueda porque se encuentra bloqueada. Capturar las pantallas necesarias para demostrar el bloqueo.

q) Liberar el bloqueo para que se ejecutar el segundo *script*. Capturar las pantallas necesarias para demostrar el desbloqueo.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Práctica 8

**Inicios de sesión y usuarios**

---

**1. Práctica de laboratorio**

- a) Restaurar la base de Datos Supermercado desde el archivo SUPERMERCADO.BAK.
- b) Crear los siguientes inicios de sesión (login), con sus respectivas contraseñas:

```
exec sp_addlogin 'usuario1', 'abc123'  
exec sp_addlogin 'usuario2', 'abc123'  
exec sp_addlogin 'usuario3', 'abc123'
```

- c) Crear los usuarios que representen a los tres inicios de sesión en la base de datos Supermercado:

```
use Supermercado  
exec sp_grantdbaccess 'usuario1'  
exec sp_grantdbaccess 'usuario2'  
exec sp_grantdbaccess 'usuario3'
```

- d) Verificar los usuarios creados:

```
sp_helpuser
```

- e) Conceder la función de lectura de datos **db\_datareader**, al **usuario1**:

```
sp_addrolemember 'db_datareader','usuario1'
```

- f) Denegar la función de lectura de datos **db\_denydatareader**, al **usuario2**:

```
sp_addrolemember 'db_denydatareader','usuario2'
```

- g) Conceder la función de lectura y escritura de datos **db\_datareader** y **db\_datawriter**, al **usuario3**:

```
exec sp_addrolemember 'db_datareader','usuario3'
exec sp_addrolemember 'db_datawriter','usuario3'
```

- h) Desde el object explorer, conectar una instancia del servidor SQL para el **usuario1**, otra para el **usuario2**, y otra para el **usuario3**.
- i) Seleccionar al **usuario1**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- j) Seleccionar al **usuario2**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- k) Seleccionar al **usuario3**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- l) Otorgar al **usuario2**, permisos de lectura sobre la tabla **Cliente**:

```
GRANT select ON cliente to usuario2
```

- m) Otorgar al **usuario1**, y al **usuario2**, permisos de INSERT Y UPDATE, sobre la tabla **Cajero**.

```
GRANT insert,update ON cajero to usuario1,usuario2
```

- n) Seleccionar al **usuario1**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- o) Seleccionar al **usuario2**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- p) Restaurar la base de Datos Farmacia desde el archivo FARMACIA.BAK.
- q) Crear tres inicios de sesión (usuario1, usuario2 y usuario3), con sus respectivas contraseñas.
- r) Crear los usuarios que representen a los tres inicios de sesión en la base de datos Farmacia.

- s) Conceder la función de lectura de datos **db\_datareader**, al **usuario1**.
- t) Denegar la función de lectura de datos **db\_denydatareader**, al **usuario2**.
- u) Conceder las funciones de lectura y escritura de datos **db\_datareader** y **db\_datawriter**, al **usuario3**.
- v) Desde el object explorer conectar una instancia del servidor SQL para el **usuario1**, otra para el **usuario2**, y otra para el **usuario 3**.
- w) Seleccionar al **usuario1**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- x) Seleccionar al **usuario2**. Comprobar si el usuario puede leer y escribir datos en las tablas.
- y) Seleccionar al **usuario3**. Comprobar si el usuario puede leer y escribir datos en las tablas.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar las instrucciones en T-SQL y las capturas de pantalla en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Práctica 9

### Copias de seguridad

---

#### 1. Práctica de laboratorio

- a) Iniciar un nuevo query. Crear un dispositivo de copia de seguridad para la base de datos **Supermercado**:

```
sp_addumpdevice 'disk' , 'Supermercado_backup', 'C:\Supermercado.bak'
```

- b) Realizar una copia de seguridad completa de la base de datos **Supermercado**, inicializando el dispositivo:

```
BACKUP DATABASE Supermercado to Supermercado_backup WITH INIT
```

- c) Ingresar un nuevo registro para la tabla **UNIDAD**.

- d) Realizar la primera copia de seguridad diferencial de la base de datos **Supermercado**, en el dispositivo **Supermercado\_backup**:

```
BACKUP DATABASE Supermercado to Supermercado_backup WITH differential, NOINIT
```

- e) Ingresar un nuevo registro para la tabla **UNIDAD**.

- f) Realizar la segunda copia de seguridad diferencial:

```
BACKUP DATABASE Supermercado to Supermercado_backup WITH differential, NOINIT
```

- g) Ingresar un último registro a la tabla **UNIDAD**. Realizar una tercera copia de seguridad diferencial.

- h) Generar un segundo dispositivo para realizar la copia de seguridad del registro de transacciones:

```
sp_addumpdevice 'disk' , 'Supermercado_log', 'C:\SupermercadoLog.bak'
```

- i) Realizar la copia de seguridad del registro de transacciones:

**BACKUP DATABASE Supermercado TO Supermercado\_log WITH INIT  
BACKUP LOG Supermercado TO Supermercado\_log WITH NOINIT**

- j) Verificar la base de datos y archivo de transacciones que se respaldaron en el dispositivo de copia de seguridad **Supermercado\_backup**:

**RESTORE FILELISTONLY FROM Supermercado\_backup  
RESTORE VERIFYONLY FROM Supermercado\_backup**

- k) Restaurar la primera copia de seguridad diferencial:

**RESTORE DATABASE Supermercado FROM Supermercado\_backup WITH NORECOVERY,REPLACE  
RESTORE DATABASE Supermercado FROM Supermercado\_backup WITH FILE=2,RECOVERY**

- l) Verificar el contenido de la tabla **Unidad**.

- m) Restaurar la segunda copia de seguridad diferencial

- n) Verificar el contenido de la tabla **Unidad**.

- o) Restaurar la tercera copia de seguridad diferencial

- p) Verificar el contenido de la tabla **Unidad**.

- q) Restaurar la base de datos completa y el registro de transacciones:

**RESTORE DATABASE Supermercado FROM Supermercado\_LOG WITH NORECOVERY,REPLACE  
RESTORE LOG Supermercado FROM Supermercado\_LOG WITH RECOVERY**

- r) Verificar el contenido de la tabla **Unidad**.

- s) Iniciar un nuevo query. Crear un dispositivo de copia de seguridad para la base de datos **Farmacia**.

- t) Realizar una copia de seguridad completa de la base de datos **Farmacia**, inicializando el dispositivo.



- u) Generar un segundo dispositivo para realizar la copia de seguridad del registro de transacciones de la base de datos **Farmacia**.
- v) Realizar la copia de seguridad del registro de transacciones de la base de datos **Farmacia**.
- w) Eliminar la base de datos **Farmacia**.
- x) Restaurar la base de datos **Farmacia** completa desde el primer dispositivo.
- y) Eliminar la base de datos **Farmacia**.
- z) Restaurar la base de datos **Farmacia** completa desde la copia de seguridad del registro de transacciones.

## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar el código T-SQL en un documento de Microsoft Word.
- Subir al enlace en el aula virtual.



## Cuarta Unidad



### 1. Práctica de laboratorio

- a) Utilizando la herramienta importación-exportación de datos de SQL Server, importar las bases de datos "COMERCIAL" y "BOTICA", desde los archivos de Microsoft Excel proporcionados. Implementar también las relaciones para cada tabla de base de datos.
- b) Exportar hacia Microsoft Access, ambas bases de datos implementadas: "COMERCIAL" y "BOTICA".
- c) Descargar de internet, una base de datos de tema libre (3 tablas en promedio) en cualquier formato, que no sea de SQL Server.
- d) Implementar e importar la base de datos descargada hacia SQL Server. Implementar también las relaciones para cada tabla de base de datos.

### 2. Fecha de entrega

..... / ..... / .....

### 3. Forma de entrega

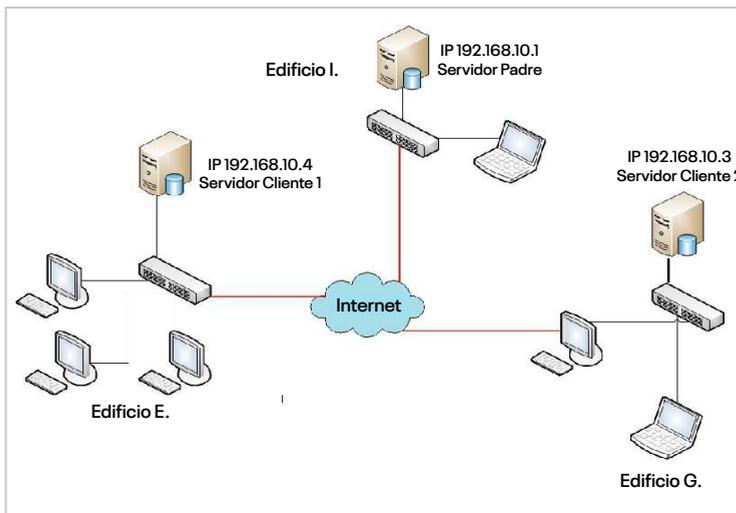
- El trabajo es individual.
- Entregar en archivos de Microsoft Access, base de datos original y backup de SQL Server, todo en un solo archivo comprimido.
- Subir al enlace en el aula virtual.

## Práctica 11

### Bases de datos distribuidas

#### 1. Práctica de laboratorio

a) Instalar las instancias necesarias del servidor SQL Server, para conformar la siguiente base de datos distribuida:



#### 2. Fecha de entrega

..... / ..... / .....

#### 3. Forma de entrega

- El trabajo es individual.
- Entregar capturas de pantallas, todo en un solo archivo de Microsoft Word.
- Subir al enlace en el aula virtual.



**Auditoría de bases de datos**

---

**1. Práctica de laboratorio**

a) **CASO:** Usted es el auditor informático de una importante institución financiera. Un día lo convocan a una reunión con el gerente general, donde se tratan temas relativos a la problemática originada por la utilización de cajeros automáticos por parte de los clientes. En particular se han detectado los siguientes inconvenientes:

1. Actualmente el banco entrega una tarjeta de plástico a cada uno de los clientes que contiene el número de cuenta (codificado en forma magnética) y un PIN. Desafortunadamente, hay clientes que escriben el PIN en la tarjeta, y en consecuencia cuando pierden la tarjeta o son robados han ocurrido extracciones de fondos no autorizadas. Este número está en aumento.
2. Los cajeros automáticos permiten hasta tres intentos de ingresar el PIN actual. Luego la máquina retiene la tarjeta. Muchos clientes olvidan sus PIN. Recientemente, un cliente enojado intentó recuperar su tarjeta utilizando una palanca.
3. El banco envía por correo las tarjetas y los PIN, sin embargo, recientemente los correos han sido robados y se han producido extracciones de fondo no autorizadas.

**El gerente general le solicita a encontrar por lo menos cinco soluciones respecto a los temas presentados. Escribir un reporte breve exponiendo sus cinco soluciones.**

b) Para el caso anterior, se le solicita redactar el plan de auditoría de bases de datos.

---

**PLAN DE AUDITORÍA DE BASE DE DATOS**

---

**1. Antecedentes**

(Describir los datos generales de la organización tomada como modelo, y resumir los síntomas iniciales detectados, que justifiquen realizar una auditoría informática)

**2. Objetivos**

(Plantear un objetivo general y cuatro objetivos específicos)

**3. Cronograma**

(Diseñar un cronograma de actividades genérico, donde se estipulen las actividades de auditoría. Incluir fechas tentativas)

**4. Presupuesto**

(Formular una lista de recursos humanos, materiales y otros que se necesitarían para llevar a cabo la auditoría, con sus respectivos costos tentativos)

**5. Anexos**

(Considerar la información adicional relevante, que considere pertinente)

---

c) Seleccionar una organización de tamaño mediano real o virtual, que requiera una auditoría de base de datos. (Puede ser la organización donde usted trabaja o hace prácticas pre profesionales). Aplique el siguiente cuestionario de auditoría, y encuentre las observaciones y recomendaciones que sean necesarias.

1. ¿Existen procedimientos formales para la operación del sistema de cómputo?

Sí ( ) NO ( )

2. ¿Están actualizados los procedimientos?

Sí ( ) NO ( )

3. Indique la periodicidad de la actualización de los procedimientos:

Semestral ( )

Anual ( )

Cada vez que haya cambio de equipo ( )



4. Indique el contenido de los instructivos de operación para cada aplicación:

Identificación del sistema ( )

Identificación del programa ( )

Periodicidad y duración de la corrida ( )

Especificación de formas especiales ( )

Fechas de creación y expiración Instructivo sobre datos de entrada y salida ( )

Altos programados y la acciones requeridas ( )

Instructivos específicos a los operadores en caso de falla del equipo ( )

Instructivos de reinicio ( )

Procedimientos de recuperación para proceso de gran duración o criterios ( )

Identificación de todos los dispositivos de la máquina a ser usados ( )

Especificaciones de resultados (cifras de control, registros de salida por archivo, etc.) ( )

5. ¿Existen órdenes de proceso para cada corrida en la computadora (incluyendo pruebas, compilaciones y producción)?

Sí ( ) NO ( )

6. ¿Son suficientemente claras para los operadores estas órdenes?

Sí ( ) NO ( )

7. ¿Existe una estandarización de las ordenes de proceso?

Sí ( ) NO ( )

8. ¿Existe un control que asegure la justificación de los procesos en el computador?

Sí ( ) NO ( )



9. ¿Cómo programan los operadores los trabajos dentro del departamento de cómputo?  
Primero que entra, primero que sale ( )  
Se respetan las prioridades, ( )  
Otra (especifique) ( )
10. ¿Los retrasos o incumplimiento con el programa de operación diaria, se revisa y analiza?  
Sí ( ) NO ( )
11. ¿Quién revisa este reporte en su caso?
12. Analice la eficiencia con que se ejecutan los trabajos dentro del departamento de cómputo, tomando en cuenta equipo y operador, a través de inspección visual, y describir sus observaciones.
13. ¿Existen procedimientos escritos para la recuperación del sistema en caso de falla?
14. ¿Cómo se actúa en caso de errores?
15. ¿Existen instrucciones específicas para cada proceso, con las indicaciones pertinentes?
16. ¿Se tienen procedimientos específicos que indiquen al operador que hacer cuando un programa interrumpe su ejecución u otras dificultades en proceso?
17. ¿Puede el operador modificar los datos de entrada?
18. ¿Se prohíbe a analistas y programadores la operación del sistema que programo o analizo?
19. ¿Se prohíbe al operador modificar información de archivos o bibliotecas de programas?



20. ¿El operador realiza funciones de mantenimiento diario en dispositivos que así lo requieran?

21. Las intervenciones de los operadores:

¿Son muy numerosas? Sí ( ) NO ( )

Se limitan los mensajes esenciales? Sí ( ) NO ( )

Otras (especifique)

---

22. ¿Se tiene un control adecuado sobre los sistemas y programas que están en operación?

Sí ( ) NO ( )

23. ¿Cómo controlan los trabajos dentro del departamento de cómputo?

24. ¿Se rota al personal de control de información con los operadores procurando un entrenamiento cruzado y evitando la manipulación fraudulenta de datos?

Sí ( ) NO ( )

25. ¿Cuentan los operadores con una bitácora para mantener registros de cualquier evento y acción tomada por ellos?

Sí ( )

Por máquina ( )

Escrita manualmente ( )

NO ( )

26. ¿Existen procedimientos para evitar las corridas de programas no autorizados?

Sí ( ) NO ( )

27. ¿Existe un plan definido para el cambio de turno de operaciones que evite el descontrol y discontinuidad de la operación?

28. Verificar que sea razonable el plan para coordinar el cambio de turno.

29. ¿Se hacen inspecciones periódicas de muestreo?

Sí ( ) NO ( )

30. Enuncie los procedimientos mencionados en el inciso anterior:

31. ¿Se controla estrictamente el acceso a la documentación de programas o de aplicaciones rutinarias?

Sí ( ) NO ( )

¿Cómo?

---

32. Verificar que los privilegios del operador se restrinjan a aquellos que le son asignados a la clasificación de seguridad de operador.

33. ¿Existen procedimientos formales que se deban observar antes de que sean aceptados en operación, sistemas nuevos o modificaciones a los mismos?

Sí ( ) NO ( )

34. ¿Estos procedimientos incluyen corridas en paralelo de los sistemas modificados con las versiones anteriores?

Sí ( ) NO ( )

35. ¿Qué precauciones se toman durante el periodo de implantación?

36. ¿Cómo se controlan las llaves de acceso (Password)?



## 2. Fecha de entrega

..... / ..... / .....

## 3. Forma de entrega

- El trabajo es individual.
- Entregar todo lo trabajado en un archivo de Microsoft Word.
- Subir al enlace en el aula virtual.



# Referencias

---

## **Básica**

Pakhira, M. (2013). *Database management system*. PHI Learning Private Limited.

## **Complementaria**

Gabillaud, J. (2015). *SQL Server 2014: administración de una base de datos transaccional*. ENI.

Rajiv, E. (2013). *Database management system (DBMS)* (3.<sup>a</sup> ed.). S. Chand & Company Ltd.

Ricardo, C. (2015). *Bases de datos*. McGraw-Hill.

Silberschatz, A., Korth, H. y Sudarshan, S. (2013). *Fundamentos de bases de datos*. McGraw-Hill

