



**INSTITUTO POLITÉCNICO NACIONAL**  
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA  
Y TECNOLOGÍAS AVANZADAS



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

NOMBRE: \_\_\_\_\_ Sánchez Ramírez Noel Adán \_\_\_\_\_. GRUPO: \_\_3TV2\_\_

Calificación: \_\_\_\_\_ FECHA: 25 de enero de 2021

**INSTRUCCIONES:** RESOLVER TODOS LOS REACTIVOS Y PROBLEMAS.

Escribir de forma clara y concisa (indicar procedimiento y resultado).  
(el examen consta de 2 páginas).

**FAVOR DE APAGAR EL CELULAR O COMPUTADORA, DE NO ATENDER LA INDICACIÓN SE LE CANCELARA EL EXAMEN Y SU CALIFICACIÓN SERA DE CERO.**

1. Crear un Login llamado Fernando, el cual debe ser mapeado a la cuenta de usuarios, asignarle una contraseña  
`CREATE LOGIN [Fernando] WITH PASSWORD=N'123456'`  
`CREATE USER usuarios`  
`FOR LOGIN Fernando`

2. El usuario Fernando debe tener autorización para consultar el esquema base\_upiita  
`CREATE SCHEMA base_upiita AUTHORIZATION Usuarios;`

3. Escriba las sentencias necesarias para cambiar el archivo Log de una base de datos distribuidas

```
ALTER DATABASE [UPIITA_BASE] MODIFY FILE (NAME = UPIITA_BASE, FILENAME = 'E:\SQLData\UPIITA_BASE.mdf');
```

```
ALTER DATABASE tempdb MODIFY FILE (NAME = UPIITA_BASELOG, FILENAME = 'E:\SQLData\UPIITA_BASELOG.ldf');
```

4. Escriba las sentencias necesarias para redimensionar el crecimiento de la base de datos distribuidas

```
ALTER DATABASE [UPIITA_BASE] MODIFY FILE ( NAME = N'UPIITA_BASE', SIZE = 16384KB, MAXSIZE = 204800KB , FILEGROWTH = 131072KB )
```

```
ALTER DATABASE [UPIITA_BASE] MODIFY FILE ( NAME = N'UPIITA_BASE_log', SIZE = 16384KB, FILEGROWTH = 131072KB )
```

5. ¿Cuál es la sentencia para REVOCAR los permisos de administrador al usuario: user\_conta  
Permisos expuestos uno por uno  
`REVOKE ALL user_conta FROM [UPIITA_BASE];`  
Visto desde un rol.  
`ALTER ROLE [db_ddladmin] DROP MEMBER user_conta;`



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

ALTER          ROLE          [db\_accessadmin]          DROP          MEMBER          user\_conta;

6. Como se le asignarían permisos de SOLO LECTURA al usuario: user\_audit  
Con esquema dbo

```
use [UPIITA_BASE]
GO
GRANT SELECT ON [dbo].[sysdiagrams] TO [usuarios]
```

Sin esquema dbo

```
use [UPIITA_BASE]
GO
GRANT SELECT ON [base_upiita].[sysdiagrams] TO [usuarios]
```

O para cualquier cosa

```
USE [UPIITA_BASE]; ALTER ROLE [db_datareader] ADD MEMBER [usuarios];
```

7. Cree un procedimiento que reciba como parámetro el nombre de usuario, contraseña, base de datos y una opción:

1 -> El usuario tendrá permisos de administrador

2 -> El usuario tendrá permisos de sólo lectura

3 -> El usuario tendrá permisos de lectura y escritura

Este procedimiento, debe, con base en la opción recibida crear el login y el usuario solicitado y dar los permisos señalados en la base que recibió como parámetro.

```
CREATE OR ALTER PROCEDURE SP_ROLES @USUARIO NVARCHAR(20), @PASS NVARCHAR(20), @BASE
NVARCHAR(20), @OPCION INT
AS
BEGIN
DECLARE @QUERY NVARCHAR(MAX) = 'USE ' + @BASE + '; CREATE LOGIN ' + @USUARIO + '
WITH PASSWORD = ''' + @PASS + '''; CREATE USER ' + @USUARIO + ' FOR LOGIN ' +
@USUARIO + ' WITH DEFAULT_SCHEMA = base_upiita;';
IF @OPCION = 1
BEGIN
SET @QUERY = @QUERY + ' ALTER ROLE db_ddladmin ADD MEMBER ' + @USUARIO
+ ';' ;
END
ELSE IF @OPCION = 2
BEGIN
SET @QUERY = @QUERY + ' ALTER ROLE db_datareader ADD MEMBER ' +
@USUARIO + ';' ;
END
ELSE IF @OPCION = 3
BEGIN
SET @QUERY = @QUERY + ' ALTER ROLE db_datareader ADD MEMBER ' +
@USUARIO + '; ALTER ROLE db_datawriter ADD MEMBER ' + @USUARIO + ';' ;
END
PRINT @QUERY;
```



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

```
EXEC(@QUERY);  
END
```

```
EXEC SP_ROLES 'P', '1234', 'UPIITA_BASE', 3
```

8. Se tiene un sistema, en el que se añadirá un control de usuarios que genere los usuarios a las bases de datos y los correspondientes logins, en la ventana de registro se solicitará:
  - a) Solicitud: Alta, Baja, Cambio en permisos
  - b) Nombre de Usuario
  - c) Password asignado
  - d) Nombre de base de datos
  - e) Permisos asignados
- Crear la o las tablas que recibirán dicha información, deberá incluir una tabla Bitacora, la cual será la última en llenarse.
- ```
CREATE DATABASE [EXAMEN];
```
- ```
USE [EXAMEN]
```
- ```
GO
```
- ```
-- Tablas
```
- ```
CREATE TABLE CAT_USUARIOS_TYPE(  
    CAT_UST_ID INT PRIMARY KEY,  
    CAT_UST_DESC NVARCHAR(20)  
);
```
- ```
CREATE TABLE CAT_USUARIOS(  
    CAT_USU_ID INT PRIMARY KEY,  
    CAT_USU_NOMBRE NVARCHAR(20) NOT NULL,  
    CAT_USU_PASS NVARCHAR(10) NOT NULL,  
    CAT_USU_TYPE INT NOT NULL DEFAULT 2,  
    CONSTRAINT FK_USUARIOS_TYPE FOREIGN KEY (CAT_USU_TYPE) REFERENCES  
CAT_USUARIOS_TYPE(CAT_UST_ID)  
);
```
- ```
CREATE TABLE HIST_REGISTROS_BITACORA(  
    HIST_REG_ID INT PRIMARY KEY,  
    HIST_USUARIO_ID INT DEFAULT 1,  
    HIST_REG_PERMISOS NVARCHAR(MAX),  
    HIST_REG_BASE NVARCHAR(500) DEFAULT 'EXAMEN',  
    HIST_REG_INIT DATETIME DEFAULT GETDATE(),  
    HIST_REG_END DATETIME,  
    CONSTRAINT FK_USUARIOS FOREIGN KEY (HIST_USUARIO_ID) REFERENCES  
CAT_USUARIOS(CAT_USU_ID)  
);
```
- Crear un trigger en la última tabla, Bitacora, el cual debe realizar la petición del usuario (Crear usuario, agregar o quitar permisos, etc)
- ```
CREATE OR ALTER TRIGGER TG_PERMISOS  
ON  
HIST_REGISTROS_BITACORA  
FOR INSERT  
AS
```



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

- BEGIN
- DECLARE @ID\_USUARIO INT;
- DECLARE @USUARIO NVARCHAR(30);
- DECLARE @PERMISOS NVARCHAR(20);
- DECLARE @BASE NVARCHAR(20);
- DECLARE @QUERY NVARCHAR(MAX);
- DECLARE @EVENTO NVARCHAR(20);
- 
- SET @ID\_USUARIO = (SELECT HIST\_USUARIO\_ID FROM inserted);
- SET @USUARIO = (SELECT CAT\_USU\_NOMBRE FROM CAT\_USUARIOS);
- SET @BASE = (SELECT HIST\_REG\_BASE FROM inserted);
- SET @QUERY = 'USE ' + @BASE + ';' + @PERMISOS + ' ON SCHEMA ::[dbo] to ' + @USUARIO + ';;';
- EXEC(@QUERY);
- SET @EVENTO = 'EXITOSO ' + @USUARIO;
- INSERT INTO HIST\_AUDITORIA(HIST\_AUDI\_ID, HIST\_EVENTO)VALUES((SELECT ISNULL(MAX(HIST\_AUDI\_ID),0) + 1 FROM HIST\_AUDITORIA), @EVENTO)
- END
- El resultado (Exitoso, no exitoso) deberá registrarse en esa misma tabla.

```
USE [EXAMEN1]
EXECUTE AS USER = 'CONSULTOR';
INSERT INTO CAT_USUARIOS_TYPE(CAT_UST_ID,CAT_UST_DESC) VALUES(3,'ANALISTA');
```

98 %

Messages

(1 row affected)

Completion time: 2021-01-26T14:30:19.7139760-06:00



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

```
USE[EXAMEN]
SELECT TOP (1000) [HIST_AUDI_ID]
,[HIST_EVENTO]
FROM [EXAMEN].[dbo].[HIST_AUDITORIA]
```

Results Messages

HIST_AUDI_ID	HIST_EVENTO
1	EXITOSO CONSULTOR

```
/****** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [HIST_REG_ID]
,[HIST_USUARIO_ID]
,[HIST_REG_PERMISOS]
,[HIST_REG_BASE]
,[HIST_REG_INIT]
,[HIST_REG_END]
,[HIST_REG_FOLIO]
FROM [EXAMEN].[dbo].[HIST_REGISTROS_BITACORA]
```

Results Messages

HIST_REG_ID	HIST_USUARIO_ID	HIST_REG_PERMISOS	HIST_REG_BASE	HIST_REG_INIT	HIST_REG_END	HIST_REG_FOLIO
1	1	GRANT INSERT	EXAMEN1	2021-01-26 14:13:13.803	NULL	1

9. Crear un script que cree una base de datos con tres tablas y tres usuarios:

- a) Administrador
- b) Consultor
- c) Analista



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

El usuario Administrador tendrá todos los privilegios, Consultor solo tendrá permisos de SELECT y Analista solo podrá realizar INSERT, UPDATE y DELETE.

```
CREATE DATABASE [EXAMEN];
USE [EXAMEN]
GO
-- Tablas
CREATE TABLE CAT_USUARIOS_TYPE(
    CAT_UST_ID INT PRIMARY KEY,
    CAT_UST_DESC NVARCHAR(20)
);
CREATE TABLE CAT_USUARIOS(
    CAT_USU_ID INT PRIMARY KEY,
    CAT_USU_NOMBRE NVARCHAR(20) NOT NULL,
    CAT_USU_PASS NVARCHAR(10) NOT NULL,
    CAT_USU_TYPE INT NOT NULL DEFAULT 2,
    CONSTRAINT FK_USUARIOS_TYPE FOREIGN KEY (CAT_USU_TYPE) REFERENCES
CAT_USUARIOS_TYPE(CAT_UST_ID)
);
CREATE TABLE HIST_REGISTROS_BITACORA(
    HIST_REG_ID INT PRIMARY KEY,
    HIST_USUARIO_ID INT DEFAULT 1,
    HIST_REG_PERMISOS NVARCHAR(MAX),
    HIST_REG_BASE NVARCHAR(500) DEFAULT 'EXAMEN',
    HIST_REG_INIT DATETIME DEFAULT GETDATE(),
    HIST_REG_END DATETIME,
    CONSTRAINT FK_USUARIOS FOREIGN KEY (HIST_USUARIO_ID) REFERENCES
CAT_USUARIOS(CAT_USU_ID)
);
-- Usuario y logins
USE [EXAMEN]
CREATE LOGIN ADMINISTRADOR WITH PASSWORD='PASSWORD'
CREATE USER ADMINISTRADOR FOR LOGIN ADMINISTRADOR WITH
DEFAULT_SCHEMA= ESQUEMA_ADMINISTRADOR

CREATE LOGIN CONSULTOR WITH PASSWORD='PASSWORD'
CREATE USER CONSULTOR FOR LOGIN CONSULTOR WITH
DEFAULT_SCHEMA= ESQUEMA_CONSULTOR

CREATE LOGIN ANALISTA WITH PASSWORD='PASSWORD'
CREATE USER ANALISTA FOR LOGIN ANALISTA WITH
DEFAULT_SCHEMA= ESQUEMA_ANALISTA
-- Permisos
GRANT INSERT ON SCHEMA :: [dbo] to ADMINISTRADOR
GRANT SELECT ON SCHEMA :: [dbo] to ADMINISTRADOR
GRANT UPDATE ON SCHEMA :: [dbo] to ADMINISTRADOR
GRANT DELETE ON SCHEMA :: [dbo] to ADMINISTRADOR

GRANT SELECT ON SCHEMA :: [dbo] to CONSULTOR

GRANT INSERT ON SCHEMA :: [dbo] to ANALISTA
GRANT UPDATE ON SCHEMA :: [dbo] to ANALISTA
```



**INSTITUTO POLITÉCNICO NACIONAL**  
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA  
Y TECNOLOGÍAS AVANZADAS



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

`GRANT DELETE ON SCHEMA :: [dbo] to ANALISTA`

- Escribir la sentencia para agregarle el permiso de insertar al Consultor  
`GRANT INSERT ON SCHEMA :: [dbo] to CONSULTOR`
- Escribir la sentencia para quitarle el permiso de eliminar al Analista  
`REVOKE DELETE ON SCHEMA :: [dbo] to ANALISTA`
- 

10. Agregar una columna a la tabla Bitacora del ejercicio 4 llamada Folio, el cual se alimentará por una secuencia (SecBitac) que comenzará en 1 e irá incrementando en 1. Con un valor máximo de 999. (Nota: En caso de ejecutar la secuencia de forma externa a la tabla, ese valor se “ocupará”, es decir, la tabla lleva el folio 10, pero ejecutamos la secuencia fuera de la tabla y nos da el valor 11, al hacer un nuevo registro en la tabla deberá mostrar el folio 12)

```
ALTER TABLE HIST_REGISTROS_BITACORA ADD HIST_REG_FOLIO INTEGER DEFAULT 1;  
CREATE SEQUENCE dbo.SECBITAC  
AS INT  
START WITH 1  
INCREMENT BY 1  
MINVALUE 1 MAXVALUE 999  
CYCLE;
```

11. Crear una consulta a la tabla BITACORA con el usuario Analista, mostrar el mensaje enviado por SQL y posterior a eso crear la sentencia necesaria para llevar a cabo la consulta sin que muestre el mensaje de error.

```
USE [EXAMEN]  
EXECUTE AS USER = 'ANALISTA'  
SELECT * FROM HIST_REGISTROS_BITACORA;
```

8 %

Messages

Msg 229, Level 14, State 5, Line 63  
The SELECT permission was denied on the object 'HIST\_REGISTROS\_BITACORA', database 'EXAMEN', schema 'dbo'.

Completion time: 2021-01-26T13:45:11.5081362-06:00



**INSTITUTO POLITÉCNICO NACIONAL**  
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA  
Y TECNOLOGÍAS AVANZADAS



INGENIERÍA TELEMÁTICA  
UNIDAD DE APRENDIZAJE: BASE DE DATOS DISTRIBUIDAS  
EXAMEN FINAL PRACTICO DEPARTAMENTAL

USE [EXAMEN]

GRANT SELECT ON SCHEMA :: [dbo] to ANALISTA  
EXECUTE AS USER = 'ANALISTA'  
SELECT \* FROM HIST\_REGISTROS\_BITACORA;

98 %

Results

Messages

HIST_REG_ID	HIST_USUARIO_ID	HIST_REG_PERMISOS	HIST_REG_BASE	HIST_REG_INIT	HIST_REG_END	HIST_REG_FOLIO
-------------	-----------------	-------------------	---------------	---------------	--------------	----------------