

# Fundamentos de Bases de Datos.

## Práctica 6.

Profesor: M.I. Gerardo Avilés Rosas

*gar@ciencias.unam.mx*

Laboratorio: Luis Eduardo Castro Omaña

*lalo\_castro@ciencias.unam.mx*

Laboratorio: Carlos Augusto Escalona Navarro

*caen@ciencias.unam.mx*

14 de marzo de 2018

Se dan a conocer especificaciones de entrega para la practica 6.

### 1. DDL

Un esquema de base de datos se especifica mediante un conjunto de definiciones expresadas mediante un lenguaje especial llamado lenguaje de definición de datos (DDL)<sup>1</sup>.

Un diccionario de datos contiene metadatos, es decir, datos acerca de los datos. El esquema de una tabla es un ejemplo de metadatos. Un sistema de base de datos consulta el diccionario de datos antes de leer o modificar los datos reales.

Especificamos el almacenamiento y los métodos de acceso usados por el sistema de bases de datos por un conjunto de instrucciones en un tipo especial de DDL denominado lenguaje de almacenamiento y definición de datos. Estas instrucciones definen los detalles de implementación de los esquemas de base de datos, que se ocultan usualmente a los usuarios.

### 2. SQL PLUS

La utilidad SQL PLUS es una herramienta de línea de comandos para la ejecución ad hoc e interactiva de instrucciones y scripts de PL-SQL y para la

---

<sup>1</sup>Siglas en inglés de Data Definition Language

automatización de tareas de creación de scripts de PL-SQL. Es posible utilizar la herramienta de las siguientes formas:

- Los usuarios escriben instrucciones PL-SQL interactivamente de una forma similar al modo en que trabajan con el símbolo del sistema. Los resultados se muestran en el símbolo del sistema.
- Los usuarios envían un trabajo SQL PLUS especificando la ejecución de una instrucción PL-SQL individual o dirigiendo la utilidad hacia un archivo de texto que contiene las instrucciones PL-SQL que se van a ejecutar. El resultado se dirige normalmente hacia un archivo de texto, aunque también se puede mostrar en el símbolo del sistema.

Algunos de los comandos mas utilizados en PL-SQL son:

- Poder loggarse a SQL PLUS

```
SQLPLUS [ { username[/password] [@connect_identifier] | / }  
        [ AS { SYSDBA | SYSOPER } ]  
        | /NOLOG  
        ]
```

- Lista de ayuda

```
HELP [ INDEX | topic ]
```

- Mostrar variables del sistema y configuración de entorno

```
SHOW { ALL | ERRORS | USER | system_variable | ... }
```

- Cambiar variables de entorno

```
SET system_variable value
```

- Iniciar una base de datos

```
STARTUP PFILE = filename  
        [ MOUNT [ dbname ] | NOMOUNT | ... ]
```

- Conectar a base de datos

```
CONNECT [ [ username [ /password ] [ @connect_identifier ]  
        [ / AS { SYSOPER | SYSDBA } ]  
        ]
```

- Listar definición de tabla, vista, procedimiento almacenado, etc..

```
DESCRIBE [ schema. ] object
```

- Salir SQL PLUS

```
{ EXIT | QUIT }  
[ SUCCESS | FAILURE | WARNING | ... ]  
[ COMMIT | ROLLBACK ]
```

Para mas información sobre SQL PLUS puedes revisar la documentación en:  
[https://docs.oracle.com/cd/B10501\\_01/server.920/a90842/ch13.html](https://docs.oracle.com/cd/B10501_01/server.920/a90842/ch13.html)

### 3. Instrucciones DDL

Instrucción	Función
CREATE DATABASE	Crea una nueva base de datos y el archivo usado para almacenarla.
CREATE TABLESPACE	Crea una nuevo tablespace para almacenar las tablas de la base de datos.
CREATE TABLE	Crea una nueva tabla.
ALTER TABLE	Modifica la definición de una tabla alterando, agregando o eliminando columnas y restricciones.
CREATE INDEX	Crea un índice en una tabla determinada.
DROP TABLE	Quita la definición de una tabla y todos sus datos, índices y restricciones.

#### 3.1. Crear Base De Datos

Crea una nueva base de datos y los archivos que se usan para almacenar la base de datos, una instantánea de base de datos, o adjunta una base de datos a partir de los archivos separados de una base de datos creada anteriormente.

```
CREATE DATABASE [ database ]  
{ USER SYS IDENTIFIED BY password  
| USER SYSTEM IDENTIFIED BY password  
| CONTROLFILE REUSE  
| MAXDATAFILES integer  
| MAXINSTANCES integer  
| CHARACTER SET charset  
| NATIONAL CHARACTER SET charset  
| SET DEFAULT  
  { BIGFILE | SMALLFILE } TABLESPACE  
| database_logging_clauses  
| tablespace_clauses  
| set_time_zone_clause  
}... ;
```

database: Es el nombre de la nueva base de datos. Nombres de base de datos deben ser únicos dentro de una instancia de Oracle DB y cumplir las reglas de identificadores.

### 3.2. Crear Tablespace

Un tablespace es una unidad lógica de almacenamiento dentro de una Oracle DB. Es un puente entre el sistema de ficheros del sistema operativo y la base de datos. Cada tablespace se compone de, al menos, un datafile y un datafile solo puede pertenecer a un tablespace. Cada tabla o índice de oracle pertenece a un tablespace, es decir cuando se crea una tabla o índice se crea en un tablespace determinado.

```
CREATE [UNDO] TABLESPACE tablespace_name
        DATAFILE Datafile_Options Storage_Options ;
```

Se puede ver mas de este tema en el manual de TABLESPACES <http://computacion.fciencias.unam.mx/~gar/fbd/laboratorio/uploaden/CreacionUsuariosTableSpace.pdf>

### 3.3. Crear Tablas

Crea una nueva tabla en ORACLE DB

```
CREATE [ GLOBAL TEMPORARY ] TABLE [ schema. ]table_name
        [ (relational_properties) ]
        [ ON COMMIT { DELETE | PRESERVE } ROWS ]
        [ physical_properties ]
        [ table_properties ] ;
```

table\_name: Es el nombre de la nueva tabla. Los nombres de tabla deben seguir las reglas para identificadores. table\_name puede tener un máximo de 128 caracteres, excepto para los nombres de tablas temporales locales (nombres precedidos de un solo signo de número (#)) que no puede superar los 116 caracteres.

### 3.4. Editar Tablas

Modifica una definición de tabla al alterar, agregar o quitar columnas y restricciones, reasignar y regenerar particiones, o deshabilitar o habilitar restricciones y desencadenadores.

```
ALTER TABLE [ schema. ] table
        [ alter_table_properties
        | column_clauses
        | constraint_clauses
        | alter_table_partitioning
```

```

| alter_external_table_clauses
| move_table_clause
]
[ enable_disable_clause
| { ENABLE | DISABLE } { TABLE LOCK | ALL TRIGGERS }
] ...
;

```

### 3.5. Tipos de datos

En Oracle DB, cada columna, variable local, expresión y parámetro tiene un tipo de datos relacionado. Un tipo de datos es un atributo que especifica el tipo de datos que el objeto puede contener: datos de enteros, datos de caracteres, datos de moneda, datos de fecha y hora, cadenas binarias, etc.

Tipos de datos de Oracle DB se organizan en las siguientes categorías:

- Numéricos exactos:
  - Enteros: Tipos de datos numéricos exactos que utilizan datos enteros.

Tipo de datos	Intervalo	Storage
bigint	De $-2^{63}(-9,223,372,036,854,775,808)$ a $2^{63} - 1(9,223,372,036,854,775,807)$	8 bytes
int	De $-2^{31}(-2,147,483,648)$ a $2^{31} - 1(2,147,483,647)$	4 bytes
smallint	De $-2^{15}(-32,768)$ a $2^{15} - 1(32,767)$	2 bytes
tinyint	De 0 a 255	1 byte

- Decimal y numéricos: Tipos de datos numéricos que tienen precisión y escala fijas. Decimal y numeric son sinónimos y se pueden usar indistintamente.

decimal[ (p[ ,s] )] y numeric[ (p[ ,s] )] Números de precisión y escala fijas. Cuando se utiliza la precisión máxima, los valores válidos se sitúan entre  $-10^{38} + 1$  y  $10^{38} - 1$ . numérico es funcionalmente equivalente a decimal.

p (precisión): El número total máximo de dígitos decimales que almacenará, tanto a la izquierda como a la derecha del separador decimal. La precisión debe ser un valor comprendido entre 1 y la precisión máxima de 38. La precisión predeterminada es 18.

s (escala): El número de dígitos decimales que se almacenará a la derecha del separador decimal. Este número se resta de p para determinar el número máximo de dígitos a la izquierda del separador

decimal. El número máximo de dígitos decimales que se puede almacenar a la derecha del separador decimal. Escala debe ser un valor comprendido entre 0 y p. Solo es posible especificar la escala si se ha especificado la precisión. La escala predeterminada es 0.

- Caracteres: Son tipos de datos de cadena de longitud fija o de longitud variable.

`char [ ( n ) ]` Datos de cadena no Unicode de longitud fija. n define la longitud de cadena y debe ser un valor entre 1 y 8.000. El tamaño de almacenamiento es n bytes.

`varchar [ ( n | max ) ]` Datos de cadena no Unicode de longitud variable. n define la longitud de cadena y puede ser un valor entre 1 y 8.000. max indica que el tamaño máximo de almacenamiento es  $2^{31} - 1$  bytes (2 GB). El tamaño de almacenamiento es la longitud real de los datos especificados + 2 bytes.

## 4. Actividad

Deberán crear las instrucciones DDL para crear el esquema de bases de datos utilizando como base el diagrama relacional que realizaron en la práctica pasada.

Deben especificar las llaves foráneas, compuestas y primarias de cada tabla; así como las restricciones con las que puede contar la tabla.

## 5. Entregables

Se debe crear un script DDL.sql el cual contendrá las instrucciones para la creación de la base de datos, sus tablas y sus restricciones.

Como todo se debe hacer por línea de comando, el archivo README.txt debe tener una explicación muy detallada de como debo correr su script, si al seguir las instrucciones no es posible crear la base de datos no se calificará la práctica.

La entrega deberá ser el día miércoles 21 de Marzo de 2018.