**MODELOS Y BASES DE DATOS**

**SQL Developer**

**Guía de autoestudio 3/6**

**2019-02**

# OBJETIVOS

Desarrollar competencias básicas para usar una herramienta simple para crear y consultar una base de datos.

# TÓPICOS

* *SQL Developer* y sus funcionalidades básicas
* Creación, modificación, eliminación de tablas de una base de datos  Control básico de integridad: claves y chequeos.
* Adición y retiro de información de una base de datos
* Consulta de la información contenida en una base de datos: datos y metadatos.

# ENTREGA

Publicar las respuestas en el espacio correspondiente en un archivo .zip , el nombre de este archivo debe ser la concatenación en orden alfabético de los primeros apellidos de cada uno de los miembros.

# A. ESTUDIO

**1. Aprendiendo a crear, actualizar y borrar tablas**

**a adicionar, actualizar y eliminar tuplas**

**a crear y borrar vistas a crear y borrar índices consultar los datos sobre los datos**

Estudie las siguientes secciones del tutorial interactivo SQLzoo.net.

|  |
| --- |
| **Reference: how to...**  2 [CREATE and DROP](http://sqlzoo.net/wiki/CREATE_and_DROP_Reference)  How to create tables, indexes, views and other things. How to get rid of them. 3 [INSERT and DELETE](http://sqlzoo.net/wiki/INSERT_and_DELETE_Reference)  How to put records into a table, change them and how to take them out again.  7 [Meta Data](http://sqlzoo.net/wiki/Meta_Data_Reference)  How to find out what tables and columns exist. How to count and limit the rows return. |

## CREATE: La instrucción CREATE TABLE se usa para crear una nueva tabla en una base de datos.

**DROP:** La instrucción DROP TABLE se usa para descartar una tabla existente en una base de datos.

**INSERT:** La instrucción INSERT INTO se usa para insertar nuevos registros en una tabla.

**DELETE:** La instrucción DELETE se usa para eliminar registros existentes en una tabla.

**METADATA:** La metadata son información guardada en SQL acerca de las propias tablas del mismo, por ejemplo (tablas, campos, store procedures).

## 2. Revisando los tipos de datos

Completen la siguiente tabla de equivalencia de tipos de datos. Escriba los tipos correspondientes en el lenguaje del modelo conceptual, en SQL y en SQL ORACLE.

|  |  |  |
| --- | --- | --- |
| **MODELO** | **SQL:2008** | **ORACLE** |
| ENTERO(N) | int | INTEGER |
| REAL(D,M) | decimal (*p*, *s* ) | decimal(p,e) |
| CARACTER | char | char |
| CADENA(N)  Fija  Flexible | varchar | varchar2 |
| HORA  FECHA  FECHA+HORA | time  date  datetime | date  timestamp |

## 3. Aprendiendo a definir las restricciones sobre atributos y de clave de forma independiente

1. Investigue la sentencia **ALTER TABLE ADD CONSTRAINT**  que le va a permitir adicionar las restricciones de atributos y claves a las tablas con nombre separando la creación de tablas de la definición de restricciones.

**R/:** El comando ADD CONSTRAINT se usa para crear una restricción después de que una tabla ya se haya creado.

1. ¿Cómo quedarían definidas las siguientes tablas separando la definición de datos de la definición de restricciones? (Un ALTER TABLE por cada restricción)

|  |  |
| --- | --- |
| CREATE TABLE games  (yr INT NOT NULL PRIMARY KEY  ,city VARCHAR(20) UNIQUE  );  La restricción de llave primaria se debe llamar PK\_GAMES  La restricción de llave única se debe llamar UK\_GAMES | CREATE TABLE games  (yr INT NOT NULL, city VARCHAR(20));  ALTER TABLE games  ADD CONSTRAINT PK\_GAMES PRIMARY KEY (yr);  ALTER TABLE games  ADD CONSTRAINT UK\_GAMES UNIQUE KEY (city); |
| CREATE TABLE Orderses  (  O\_Id int NOT NULL PRIMARY KEY,  OrderNo int NOT NULL,  P\_Id int FOREIGN KEY REFERENCES Persons(P\_Id)  )  La restricción de llave primaria de debe llamar PK\_ORDENES  La restricción de llave foránea debe llamarse  FK\_ORDENES\_PERSONAS | CREATE TABLE Orderses  (O\_Id int NOT NULL,  OrderNo int NOT NULL,  P\_Id int);  ALTER TABLE Orderses  ADD CONSTRAINT PK\_ORDENES PRIMARY KEY (O\_Id);  ALTER TABLE Orderses  ADD CONSTRAINT FK\_ORDENES\_PERSONAS  FOREIGN KEY (P\_Id) REFERENCES Persons(P\_Id); |

# B. INVESTIGANDO SQL Developer

Considerando la herramienta SQL Developer

1. Investigue las funcionalidades básicas de la herramienta.

R/:

* Crear y probar conecciones
* Crear, examinar y modificar objetos
* Consultar y actualizar datos
* Exportar datos y DDL
* Importar datos
* Esquema de copia y comparación
* Comandos de proceso
* Editar PL/SQL
* Ejecutar PL/SQL con su Debug
* Ejecutar y crear reportes
* Unidad de pruebas de PL/SQL
* Visor y modelador de datos
* Migrar a bases de datos de terceros

**Instalando**

Instale la herramienta SQL Developer. ¿Son claras las instrucciones de instalación? ¿Se le presentó algún problema?

**R/:** Las instrucciones son claras, aunque el principal inconveniente fue que para poder descargarlo fue necesario crear una nueva cuenta en Oracle, puesto que al colocar los mismos datos bd# para ingresar al developer indicaba que era un inicio de sesión no válido.

**Arrancando**

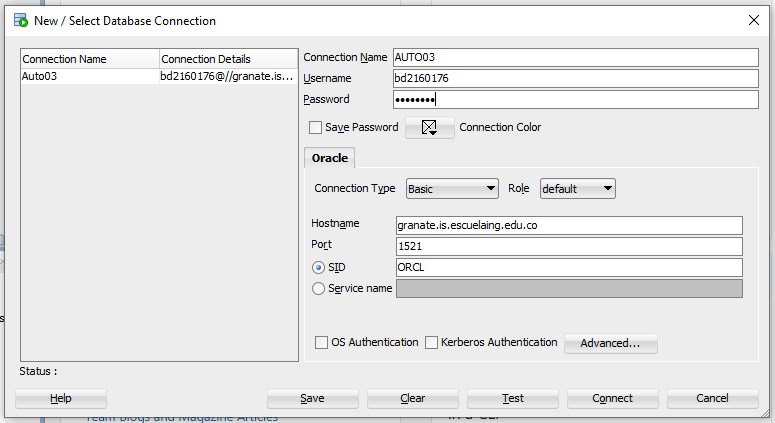
Realice y explique cómo se deben realizar las siguientes acciones:

* + Establecer una conexión con el motor ORACLE de la ESCUELA

**R/:** En una primera instancia se debe de crear una conección manualmente,para poder establecer una conexión es necesario especificar:

* HostName: En el caso de la escuela, su HostName es *granate.is.escuelaing.edu.co*
* Port: El listener port por defecto para la base de datos de ORACLE es 1521
* SID:El identificador del sistema es ORCL para la base de datos de 10*g* y 11*g*, o xe para la versión 10*g express*

Además, se requiere un usuario y contraseña para poder acceder, y establecer un nombre a la conexión.



* + Consultar toda la información posible que hay en su cuenta

**R/: No supimos a que se refería esta pregunta, además de que no encontramos ningun tipo de información acerca de la cuenta.**

## C. musicians

Para escribir el primer archivo de comandos vamos a crear la base de datos **musicians.** Los comandos deben quedar en un archivo separado con los siguientes comentarios.

Paso a paso escriban y prueben cada una de sus secciones de la base de datos.

* Crear la base de datos sin restricciones (Tablas) **YA**
* Adicionar las restricciones a la base de datos

(Atributos, Primarias, Únicas, Foraneas) **YA**

* Poblar la base de datos con los datos iniciales (PoblandoOK) Automaticen la generación de las instrucciones INSERT.

Dejen en el archivo las consultas correspondientes en comentarios.**YA**

* Probar las restricciones con los casos definidos en NoOK (PoblandoNoOK) **YA**
* Probar cada una de las consultas (Consultas) **YA**
* Despoblar la base de datos (XPoblar)

**DELETE FROM performances;**

**DELETE FROM has\_composed;**

**DELETE FROM composer ;**

**DELETE FROM composition;**

**DELETE FROM concert;**

**DELETE FROM plays\_in;**

**DELETE FROM band;**

**DELETE FROM performer;**

**DELETE FROM musician;**

**DELETE FROM place;**

* Eliminar toda la información de la base de datos (XTablas)

**DROP TABLE band CASCADE CONSTRAINTS;**

**DROP TABLE composer CASCADE CONSTRAINTS;**

**DROP TABLE composition CASCADE CONSTRAINTS;**

**DROP TABLE concert CASCADE CONSTRAINTS;**

**DROP TABLE has\_composed CASCADE CONSTRAINTS;**

**DROP TABLE musician CASCADE CONSTRAINTS;**

**DROP TABLE performance\_ CASCADE CONSTRAINTS;**

**DROP TABLE performer CASCADE CONSTRAINTS;**

**DROP TABLE place CASCADE CONSTRAINTS;**

**DROP TABLE plays\_in CASCADE CONSTRAINTS;**

**Opcion 2 para borrar todo**

**BEGIN**

**FOR cur\_rec IN (SELECT object\_name, object\_type**

**FROM user\_objects**

**WHERE object\_type IN**

**('TABLE',**

**'VIEW',**

**'PACKAGE',**

**'PROCEDURE',**

**'FUNCTION',**

**'SEQUENCE'**

**))**

**LOOP**

**BEGIN**

**IF cur\_rec.object\_type = 'TABLE'**

**THEN**

**EXECUTE IMMEDIATE 'DROP '**

**|| cur\_rec.object\_type**

**|| ' "'**

**|| cur\_rec.object\_name**

**|| '" CASCADE CONSTRAINTS';**

**ELSE**

**EXECUTE IMMEDIATE 'DROP '**

**|| cur\_rec.object\_type**

**|| ' "'**

**|| cur\_rec.object\_name**

**|| '"';**

**END IF;**

**EXCEPTION**

**WHEN OTHERS**

**THEN**

**DBMS\_OUTPUT.put\_line ( 'FAILED: DROP '**

**|| cur\_rec.object\_type**

**|| ' "'**

**|| cur\_rec.object\_name**

**|| '"'**

**);**

**END;**

**END LOOP;**

**END;**

|  |  |
| --- | --- |
| **COMPONENTE** | **CONTENIDO** |
| Tablas | Creación de tablas.  No olvide indicar las condiciones para valores desconocidos. |
| Atributos | Creación de restricciones sobre los atributos de tablas  Restricciones con nombre CK\_nombreTabla\_nombreAtributo |
| Primarias | Creación de claves primarias.  Restricciones con nombre PK\_nombreTabla |
| Únicas | Creación de claves únicas.  Restricciones con nombre UK\_nombreTabla\_nombreAtributo |
| Foráneas | Creación de claves foránea.  Restricciones con nombre FK\_nombreTabla\_nombreTablaReferenciada |
| Consultas | Las consultas propuestas de la base de datos |
| PoblarOK | Adicionar información a la tabla |
| PoblarNoOK | Ejemplares que ilustran la protección de la base de datos |
| XPoblar | Eliminar los datos de la base de datos |
| XTablas | Eliminar las tablas de la base de datos |