Programación en Base de Datos

DICCIONARIO DE DATOS

Semana 02:

OBJETIVO DEL LABORATORIO

Diseña e implementa modelos de datos y diccionarios, mediante PL/SQL.

MARCO TEÓRICO

El diccionario de datos pertenece a SYS y se almacena en el tablespace SYSTEM. Se crea en el momento de la creación de la base de datos y es actualizado automáticamente por Oracle cuando se ejecutan sentencias SQL DDL (Data Definition Language) (CREATE, ALTER, DROP).

Para oracle es importante tener una bitácora de todo lo que se realiza en su motor de base de datos, de esa manera el tendrá el control de los componentes reciente mente configurados en Oracle.

RECURSOS

Hardware

* Sistema. Oracle en Windows requiere un PC Intel x86, AMD64 o Intel EM64T
* Memoria. Al menos 1 GB de RAM y el doble en virtual.
* Espacio en disco duro. Al menos 6 GB para la instalación (algunas instalaciones requieren menos). Además, necesitamos poder almacenar 500 MB en la carpeta TEMP del sistema.
* Tarjeta gráfica. Debe de ser capaz de mostrar 1024 por 768 píxeles como mínimo y 256 colores.

Software

Sistema Operativo.

* + Windows 2003 Server y 2003 Server R2.
  + Windows XP Professional.
  + Windows Vista, pero no la versión Home Edition.
  + Windows Server 2008 y 2012. No la versión Server Core.

Compiladores. Se usan para la gente que crean aplicaciones en Oracle usando lenguajes como Pro C, Pro COBOL, JAVA.

* + Visual C++.NET 2005 8.0 o Intel 10.1 C, .Net Express.

Navegador. Para configurar algunos servicios de Oracle. Debe de ser navegador moderno (Internet Explorer 6 o superior, Firefox 2.0 o superior, Safari 3.1 o superior, Chrome 3.0 o superior)

Usar dirección IP única en la máquina en la que se instala Oracle. Es decir, no usar DHCP para direccionar la IP en el servidor de Oracle. No es un requisito obligatorio, pero es muy recomendable.

IMPORTANTE

**LA ESTRUCTURA Y DATOS DE LAS TABLAS, SE ENCUENTRAN**

**EN LOS ARCHIVOS SCRIPT DE BASE DE DATOS.**

**Semana02\_Diccionarios.sql,**

**Semana02\_Usuarios.sql,**

**FAVOR DE DESCARGAR TODOS LOS ARCHIVOS.**

Implementa modelos de datos que incorporen reglas o restricciones.

**El diccionario de datos es un conjunto de tablas y vistas que dan información sobre el contenido de una base de datos:**

* **las estructuras de almacenamiento;**
* **las usuarios y sus derechos;**
* **los objetos (tablas, vistas, índices, procedimientos, funciones, etc.).**

**El diccionario de datos pertenece a SYS y se almacena en el tablespace SYSTEM. Se crea en el momento de la creación de la base de datos y es actualizado automáticamente por Oracle cuando se ejecutan sentencias SQL DDL (Data Definition Language) (CREATE, ALTER, DROP).**

**Para el usuario, es suficiente con ejecutar consultas SELECT. Salvo excepción, toda la información se almacena en mayúsculas en el diccionario de datos; ¡conviene tener esto en cuenta en el momento de escribir las cláusulas WHERE!**

**El diccionario de datos se carga en memoria en la parte Dictionary Cache de la Shared Pool y Oracle lo utiliza internamente para el tratamiento de consultas.**

**El diccionario de datos se crea en el momento de la creación de la base de datos. Desde un punto de vista práctico, las tablas propiamente dichas del diccionario de datos no son documentadas por Oracle y son difíciles de utilizar. En cambio, gracias a los scripts aportados por Oracle, es posible crear las vistas (y los sinónimos públicos) que, ellas sí, son documentadas y permiten efectivamente explotar el diccionario de datos.**

**Tablas de la Base de Datos:**

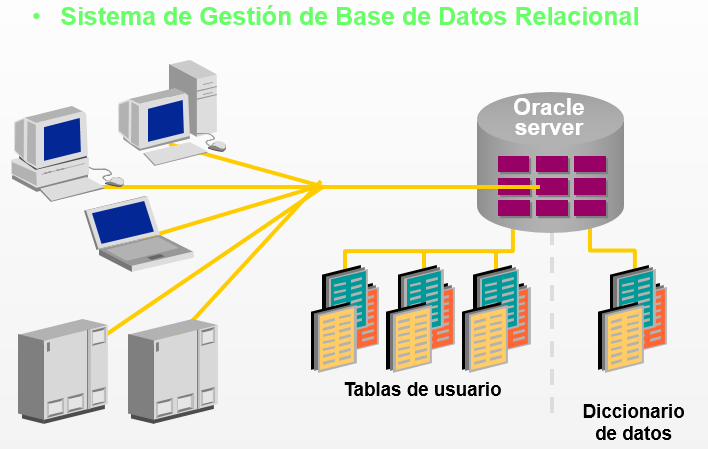
**Tablas de Usuario:**

* **Son una recopilación de tablas creadas y mantenidas por el usuario**
* **Contienen información de usuario**

**Diccionario de Datos:**

* **Es una recopilación de tablas creadas y mantenidas por Oracle Server**
* **Contienen información de la base de datos**

**Sistema de Gestión de Base de Datos Relacional.**

****

UTILIZA EL GESTOR DE BASE DE DATOS ORACLE PARA SELECCIONAR INFORMACIÓN A TRAVÉS DE LOS OBJETOS DEL DICCIONARIO DE DATOS.

Consulta del diccionario de Datos.

* Es fundamental para cualquier base de datos
* Describe la base de datos y sus objetos
* Contiene tablas y vistas de sólo lectura
* Es propiedad del usuario SYS
* Lo mantiene Oracle Server
* Se accede a él con SELECT

Información en Diccionario de Datos.

* Estructuras lógicas y físicas de la base de datos
* Usuarios registrados en la base de datos
* Definiciones e integridad de todos los esquemas
* Información de Auditoría
* Información de espacio utilizado por los objetos
* Roles y Privilegios de los usuarios

Estructura del Diccionario.

Tablas Base

* Almacena información de la base de datos
* Solo Oracle Server puede leer y escribir estas tablas.
* Se crean al momento de crear la base de datos

Vistas

* Utilizadas para mostrar de forma resumida la información contenida en las tablas base.
* Se acceden a ellas a través de sinónimos públicos.
* Ver todas las vistas del diccionario de datos

**Sql> SELECT \* FROM Dictionary;**

* Ver las columnas de las vistas del diccionario de datos

**Sql> SELECT \* FROM Dict\_columns;**

Ejemplos:

* Ver objetos propiedad del usuario

**Sql> SELECT \* FROM User\_Objects;**

* Ver tablespaces de la base de datos

**Sql> SELECT \* FROM Dba\_tablespaces;**

* Vea tipos de objeto distintos propiedad del usuario

**Sql> SELECT table\_name FROM user\_tables;**

* Vea tablas, vistas, sinónimos y secuencias propiedad del usuario.

**Sql> SELECT DISTINCT object\_type FROM user\_objects;**

* Vea los nombres de tablas propiedad del usuario.

**Sql> SELECT \* FROM user\_catalog;**

* Se pueden ver los archivos control file

**Sql> SELECT \* FROM V$CONTROLFILE;**

* Se pueden ver los constraints definidos para una tabla específica en la tabla

**Sql> SELECT \* FROM USER\_CONSTRAINTS;**

* Tablespace agrupada por tipo de segmento y ordenada por tablespace.

**Sql> Select segment\_type, tablespace\_name, count(\*) from dba\_segments**

**group by segment\_type,tablespace\_name order by 2 desc**

Las Vistas tienen Categorías:

* DBA: Contiene información de todos los objetos que hay en todos los esquemas de la base de datos.
* ALL: Contiene información de todos los objetos a los que puede acceder el usuario en sesión.
* USER: Contiene información de objetos de propiedad del usuario en sesión.

**Uso del Diccionario de Datos:**

* Oracle consulta el diccionario de datos para hallar la información acerca de los usuarios, esquemas y estructuras almacenadas.
* Oracle modifica el diccionario de datos cada vez que una instrucción DDL es usada.
* Cualquier usuario puede utilizar el diccionario de datos.
* Vista maestra que es el punto de partida para conocer los objetos de la BD.
* Campos:
  + OWNER: Dueño del objeto
  + OBJECT\_NAME: el nombre del objeto
  + OBJETC\_TYPE: el tipo de objeto: tabla, índice, package, etc.
  + CREATED: fecha de creación del objeto.

Ejemplos:

* Para las tablas del usuario scott

SELECT TABLE\_NAME FROM ALL\_TABLES WHERE OWNER ='SCOTT';

* Usuarios conectados en este momento

SELECT COUNT(\*), USERNAME FROM V$SESSION GROUP BY USERNAME;

|  |  |
| --- | --- |
| INFORMACION | VISTAS CORRESPONDIENTES |
| Errores de compilación | all\_errors, dba\_errors, user\_errors |
| Trabajos en la BD | all\_jobs, dba\_jobs,user\_jobs |
| Usuarios | all\_users, dba\_users, user\_users |
| Objetos de la BD: Tablas, programas, vistas secuencias, índices, etc. | all\_objects, dba\_objects, user\_objects |
| Dependencias entre los objetos (PROCEDURE, FUNCTION, PACKAGE, PACKAGE BODY) | all\_dependencies, dba\_dependencies, user\_dependencies |
| Tablas de la base de datos | All\_tables, dba\_tables, user\_tables |
| Vistas en la BD | All\_views, dba\_views, user\_views |
| Secuencias | All\_sequences, dba\_sequences, user\_sequences |
| Nombres de tablas, vistas, secuencias, sinónimos | All\_catalog, dba\_catalog, user\_catalog |
| Columnas de las tablas y vistas | All\_tab\_columns, dba\_tabs\_colunms, user\_tab\_colunms |
| Comentarios sobre tablas o columnas | All\_tab\_comments, dba\_tab\_comments, user\_tab\_comments, All\_col\_comments, dba\_col\_comments, user\_col\_comments |

|  |  |
| --- | --- |
| INFORMACION | VISTAS CORRESPONDIENTES |
| Restricciones de integridad | All\_constraints, dba\_constraints, user\_constrains, all\_cons\_columns, dba\_cons\_columns, user\_cons\_columns |
| Triggers en la BD | All\_triggers, dba\_triggers, user\_triggers |
| Uso de las columnas en un trigger | All\_trigger\_cols, dba\_trigger\_cols, user\_trigger\_cols |

**Tablas y vistas:**

* **ALL\_ALL\_TABLES**
* **DBA\_ALL\_TABLES**
* **USER\_ALL\_TABLES**
* **ALL\_COL\_COMMENTS**
* **DBA\_COL\_COMMENTS**
* **USER\_COL\_COMMENTS**
* **ALL\_PARTIAL\_DROP\_TABS**
* **DBA\_PARTIAL\_DROP\_TABS**
* **USER\_PARTIAL\_DROP\_TABS**
* **ALL\_REFS**
* **DBA\_REFS**
* **USER\_REFS**
* **ALL\_TAB\_COLUMNS**
* **DBA\_TAB\_COLUMNS**
* **USER\_TAB\_COLUMNS**
* **ALL\_TAB\_COMMENTS**
* **DBA\_TAB\_COMMENTS**
* **USER\_TAB\_COMMENTS**
* **ALL\_TABLES**
* **DBA\_TABLES**
* **USER\_TABLES**
* **ALL\_UNUSED\_COL\_TABS**
* **DBA\_UNUSED\_COL\_TABS**
* **USER\_UNUSED\_COL\_TABS**
* **ALL\_UPDATABLE\_COLUMNS**
* **DBA\_UPDATABLE\_COLUMNS**
* **USER\_UPDATABLE\_COLUMNS**
* **ALL\_VIEWS**
* **DBA\_VIEWS**
* **USER\_VIEWS**

**Secuencias**

* **ALL\_SEQUENCES**
* **DBA\_SEQUENCES**
* **USER\_SEQUENCES**

**Sinónimos**

* **ALL\_SYNONYMS**
* **DBA\_ SYNONYMS**
* **USER\_ SYNONYMS**

**Índices**

* ALL\_IND\_COLUMNS
* DBA\_IND\_COLUMNS
* USER\_IND\_COLUMNS
* ALL\_IND\_EXPRESSIONS
* DBA\_IND\_EXPRESSIONS
* USER\_IND\_EXPRESSIONS
* ALL\_INDEXES
* DBA\_INDEXES
* USER\_INDEXES
* ALL\_INDEXTYPES
* DBA\_INDEXTYPES
* USER\_INDEXTYPES
* ALL\_INDEXTYPE\_OPERATORS
* DBA\_INDEXTYPE\_OPERATORS
* USER\_NDEXTYPE\_OPERATORS

**Subrogramas**

* ALL\_ARGUMENTS
* USER\_ARGUMENTS

**Disparadores**

* ALL\_INTERNALS\_TRIGGERS
* DBA\_INTERNAL\_TRIGGERS
* USER:INTERNAL\_TRIGGERS
* ALL\_TRIGGERS
* DBA\_TRIGGERS
* USER\_TRIGGERS
* ALL\_TRIGGER\_COLS
* DBA\_TRIGGER\_COLS
* USER\_TRIGGER\_COLS
* Código fuente y errores de compilación
* ALL\_ERRORS
* DBA\_ERRORS
* USER\_ERRORS
* ALL\_SOURCE
* DBA\_SOURCE
* USER\_SOURCE

**Restricciones**

* ALL\_CONS\_COLUMNS
* DBA\_CONS\_COLUMNS
* USER\_CONS\_COLUMNS
* ALL\_CONSTRAINTS
* DBA\_CONSTRAINTS
* USER\_CONSTRAINTS

**Privilegios y concesiones**

* ALL\_COL\_PRIVS
* DBA\_COL\_PRIVS
* USER\_COL\_PRIVS
* ALL\_COL\_PRIVS\_MADE
* USER\_COL\_PRIVS\_MADE
* ALL\_COL\_PRIVS\_RECD
* USER\_COL\_PRIVS\_RECD

**Provilegios de Tablas**

* ALL\_TABS\_PRIVS
* DBA\_TABS\_PRIVS
* USER\_TABS\_PRIVS
* ALL\_TABS\_PRIVS\_MADE
* DBA\_TABS\_PRIVS\_MADE
* USER\_TABS\_PRIVS\_MADE

**Privilegios del sistema**

* DBA\_SYS\_PRIVS
* USER\_SYS\_PRIVS

**Dependencias**

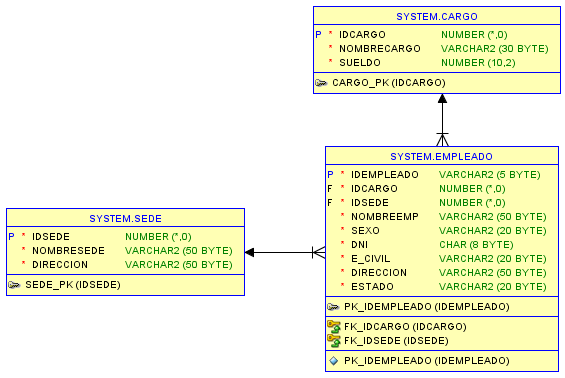
* ALL\_DEPENDENCIES
* DBA\_DEPENDENCIES
* USER\_DEPEDENCIES

**Estructuras de a base de datos**

* V$TABLESPACE
* USER\_TABLESPACES
* DBA\_DATA\_FILES
* V$DATAFILE

IMPLEMENTA ÍNDICES, SECUENCIAS Y SINÓNIMOS EN LAS TABLAS CREADAS CON UN USUARIO DE ORACLE.

Se tiene el siguiente Modelo Relacional:



AHORA PRUEBE UD. MISMO,

PASO 1:

Inicie sesión en PL/SQL usando al usuario SYSTEM y la contraseña: oracle, proceda a redactar y ejecutar el siguiente Script de Base de Datos que corresponde al diagrama de base de datos propuesto, donde se crearan las tablas: EMPLEADO, CARGO y SEDE. Que están debidamente relacionados.

**Create Table Cargo (**

**IdCargo Integer Not Null,**

**NombreCargo Varchar2(30) Not Null**

**CHECK (NombreCargo IN ('ADMINISTRADOR', 'GERENTE', 'SERVICIOS',**

**'CONTADOR','SECRETARIA')),**

**Sueldo Number(10,2) Not Null CHECK (Sueldo IN (980, 1500, 2500, 3000, 4000)),**

**CONSTRAINT PK\_IdCargo Primary Key (IdCargo),**

**CONSTRAINT CHK\_Mayuscula\_NombreCargo CHECK (NombreCargo = NLS\_UPPER(NombreCargo)));**

**Create Table Sede (**

**IdSede Integer Not Null,**

**NombreSede Varchar2(50) Not Null,**

**Direccion Varchar2(50) Not Null,**

**CONSTRAINT PK\_IdSede Primary Key(IdSede),**

**CONSTRAINT CHK\_Nombre\_Sede\_Mayusculas CHECK (NombreSede = NLS\_UPPER(NombreSede)),**

**CONSTRAINT CHK\_Direccion\_Mayusculas CHECK (Direccion = NLS\_UPPER(Direccion)));**

**Create Table Empleado(**

**IdEmpleado Varchar2(5) Not Null,**

**IdCargo Integer Not Null,**

**IdSede Integer Not Null,**

**NombreEmp Varchar2(50) Not Null,**

**Sexo Varchar2(20) Not Null CHECK (Sexo IN ('MASCULINO','FEMENINO')),**

**Dni Char(8) Not Null,**

**E\_Civil Varchar2(20) Not Null CHECK (E\_Civil IN ('SOLTERO', 'CASADO',**

**'VIUDO','DIVORCIADO')),**

**Direccion Varchar2(50) Not Null,**

**Estado Varchar2(20) Not Null,**

**CONSTRAINT PK\_IdEmpleadoX Primary Key (IdEmpleado),**

**CONSTRAINT CHK\_Nombre\_Empleado\_MayusX CHECK (NombreEmp = NLS\_UPPER(NombreEmp)),**

**CONSTRAINT CHK\_Direccion\_MayusX CHECK (Direccion = NLS\_UPPER(Direccion)),**

**CONSTRAINT CHK\_EstadoX CHECK (Estado IN ('ACTIVO', 'BLOQUEADO', 'CESADO',**

**'SUSPENDIDO','DESPEDIDO')),**

**CONSTRAINT FK\_IdCargoX Foreign Key (IdCargo) References Cargo(IdCargo),**

**CONSTRAINT FK\_IdSedeX Foreign Key (IdSede) References Sede(IdSede));**

**--Creando el indice EMPLEADO\_NOMBRE\_IDX en la tabla EMPLEADO sobre la columna NombreEmp**

**CREATE INDEX EMPLEADO\_NOMBRE\_IDX ON EMPLEADO (UPPER(NombreEmp));**

**--Creando la secuencia GENERAR\_CARGOIDX\_SEQ para la tabla CARGO**

**CREATE SEQUENCE GENERAR\_CARGOIDX\_SEQ**

**INCREMENT BY 1**

**START WITH 1**

**MINVALUE 1**

**MAXVALUE 5000**

**NOCACHE NOCYCLE;**

**--Creando la secuencia GENERAR\_SEDEIDX\_SEQ para la tabla SEDE**

**CREATE SEQUENCE GENERAR\_SEDEIDX\_SEQ**

**INCREMENT BY 1**

**START WITH 1**

**MINVALUE 1**

**MAXVALUE 5000**

**NOCACHE NOCYCLE;**

**--Probando las Secuencias GENERAR\_CARGOIDX\_SEQ y GENERAR\_SEDEIDX\_SEQ**

**Insert Into Cargo Values(GENERAR\_CARGOIDX\_SEQ.NEXTVAL,'ADMINISTRADOR',4000);**

**Insert Into Cargo Values(GENERAR\_CARGOIDX\_SEQ.NEXTVAL,'GERENTE',3000);**

**Insert Into Cargo Values(GENERAR\_CARGOIDX\_SEQ.NEXTVAL,'SERVICIOS',980);**

**Insert Into Cargo Values(GENERAR\_CARGOIDX\_SEQ.NEXTVAL,'CONTADOR',2500);**

**Insert Into Cargo Values(GENERAR\_CARGOIDX\_SEQ.NEXTVAL,'SECRETARIA',1500);**

**Insert Into Sede Values(GENERAR\_SEDEIDX\_SEQ.NEXTVAL,'MIRAFLORES','AV. LARCO 1234');**

**Insert Into Sede Values(GENERAR\_SEDEIDX\_SEQ.NEXTVAL,'LIMA CENTRO','AV. WILSON 1923');**

**Insert Into Sede Values(GENERAR\_SEDEIDX\_SEQ.NEXTVAL,'OLIVOS','CALLE LOS TULIPANES 223');**

**Select \* From Cargo;**

**Select \* From Sede;**

--Eliminando las Secuencias GENERAR\_CARGOIDX\_SEQ y GENERAR\_SEDEIDX\_SEQ

DROP SEQUENCE GENERAR\_CARGOIDX\_SEQ;

DROP SEQUENCE GENERAR\_SEDEIDX\_SEQ;

--

-- Crear los sinónimos para las siguientes tablas:

-- CARGO por CAR

-- SEDE por SED

-- EMPLEADO por EMP

CREATE SYNONYM CAR FOR CARGO;

CREATE SYNONYM SED FOR SEDE;

CREATE SYNONYM EMP FOR EMPLEADO;

--Probando los sinonimos

Select \* From CAR;

Select \* From SED;

Select \* From EMP;

--Eliminando los sinónimos CAR, SED, EMP

DROP SYNONYM CAR;

DROP SYNONYM SED;

DROP SYNONYM EMP;

--PROBANDO EL DICCIONARIO DE DATOS

--Ver objetos propiedad de usuario.

SELECT \* FROM User\_Objects;

--Ver TableSpace de la base de datos.

SELECT \* FROM Dba\_tablespaces;

--Ver tipos de objeto distintos propiedad del usuario

SELECT table\_name FROM user\_tables;

--Ver tablas, vistas, sinónimos y secuencias propiedad del usuario.

SELECT DISTINCT object\_type FROM user\_objects;

--Vea los nombres de tablas propiedad del usuario.

SELECT \* FROM user\_catalog;

--Se pueden ver los archivos control file

SELECT \* FROM V$CONTROLFILE;

--Se pueden ver los constraints definidos para una tabla específica en la tabla

SELECT \* FROM USER\_CONSTRAINTS;

--Tablespace agrupada por tipo de segmento y ordenada por tablespace.

Select segment\_type, tablespace\_name, count(\*)

from dba\_segments

group by segment\_type,tablespace\_name

order by 2 desc;

--Para las tablas del usuario scott

SELECT TABLE\_NAME FROM ALL\_TABLES WHERE OWNER ='SCOTT';

--Usuarios conectados en este momento

SELECT COUNT(\*), USERNAME FROM V$SESSION GROUP BY USERNAME;

PUEDE COPIAR Y PEGAR ESTE CODIGO AL ENTORNO DE PL/SQL.

**--Eliminando las Secuencias GENERAR\_CARGOIDX\_SEQ y GENERAR\_SEDEIDX\_SEQ**

**DROP SEQUENCE GENERAR\_CARGOIDX\_SEQ;**

**DROP SEQUENCE GENERAR\_SEDEIDX\_SEQ;**

**-- Crear los sinónimos para las siguientes tablas:**

**-- CARGO por CAR**

**-- SEDE por SED**

**-- EMPLEADO por EMP**

**CREATE SYNONYM CAR FOR CARGO;**

**CREATE SYNONYM SED FOR SEDE;**

**CREATE SYNONYM EMP FOR EMPLEADO;**

**--Probando los sinonimos**

**Select \* From CAR;**

**Select \* From SED;**

**Select \* From EMP;**

**--Eliminando los sinónimos CAR, SED, EMP**

**DROP SYNONYM CAR;**

**DROP SYNONYM SED;**

**DROP SYNONYM EMP;**

**--PROBANDO EL DICCIONARIO DE DATOS**

**--Ver objetos propiedad de usuario.**

**SELECT \* FROM User\_Objects;**

**--Ver TableSpace de la base de datos.**

**SELECT \* FROM Dba\_tablespaces;**

**--Ver tipos de objeto distintos propiedad del usuario**

**SELECT table\_name FROM user\_tables;**

**--Ver tablas, vistas, sinónimos y secuencias propiedad del usuario.**

**SELECT DISTINCT object\_type FROM user\_objects;**

**--Vea los nombres de tablas propiedad del usuario.**

**SELECT \* FROM user\_catalog;**

**--Se pueden ver los archivos control file**

**SELECT \* FROM V$CONTROLFILE;**

**--Se pueden ver los constraints definidos para una tabla específica en la tabla**

**SELECT \* FROM USER\_CONSTRAINTS;**

**--Tablespace agrupada por tipo de segmento y ordenada por tablespace.**

**Select segment\_type, tablespace\_name, count(\*)**

**from dba\_segments**

**group by segment\_type,tablespace\_name**

**order by 2 desc;**

**--Para las tablas del usuario scott**

**SELECT TABLE\_NAME FROM ALL\_TABLES WHERE OWNER ='SCOTT';**

**--Usuarios conectados en este momento**

**SELECT COUNT(\*), USERNAME FROM V$SESSION GROUP BY USERNAME;**

**--Ver todas las tablas del manejador de todos los TableSpace**

**SELECT \* FROM ALL\_ALL\_TABLES;**

**--Ver todas las tablas de todos los usuarios.**

**SELECT \* FROM USER\_ALL\_TABLES;**

**--Comentarios de las tablas del usuario actual.**

**SELECT \* FROM USER\_TAB\_COMMENTS;**

**--Comentarios de las columnas del usuario actual.**

**SELECT \* FROM USER\_COL\_COMMENTS;**

**--Comentarios de las tablas de todos los usuarios (sólo administradores)**

**SELECT \* FROM ALL\_TAB\_COMMENTS;**

**--Comentarios de las columnas de todos los usuarios (sólo administradores).**

**SELECT \* FROM ALL\_COL\_COMMENTS;**

FIN DE PRUEBA DE DICCIONARIOS.

CREACIÓN Y CONFIGURACIÓN DE CUENTAS DE USUARIO.

<http://epnbdd-oracle.blogspot.com/2012/05/privilegios-y-usuarios-en-oracle.html>

El siguiente es un resumen de algunas consideraciones al momento de crear un usuario o cuenta en Oracle, y los privilegios y roles que le podemos asignar.

* El nombre de usuario no debe superar 30 caracteres, no debe tener caracteres especiales y debe iniciar con una letra.
* Un método de autentificación. El más común es una clave o password, pero Oracle 10g soporta otros métodos (como biometric, certificado y autentificación por medio de token).
* Un Tablespace default, el cual es donde el usuario va a poder crear sus objetos por defecto, sin embargo, esto no significa que pueda crear objetos, o que tenga una cuota de espacio. Estos permisos se asignan de forma separada, salvo si utiliza el privilegio RESOURCE el que asigna una quota unlimited, incluso en el Tablespace SYSTEM! Sin embargo si esto ocurre, Ud. puede posteriormente mover los objetos creados en el SYSTEM a otro Tablespace.
* Un Tablespace temporal, donde el usuario crea sus objetos temporales y hace los sort u ordenamientos.
* Un perfil o profile de usuario, que son las restricciones que puede tener su cuenta (opcional).

HAGALO UD. MISMO.

PASO 1:

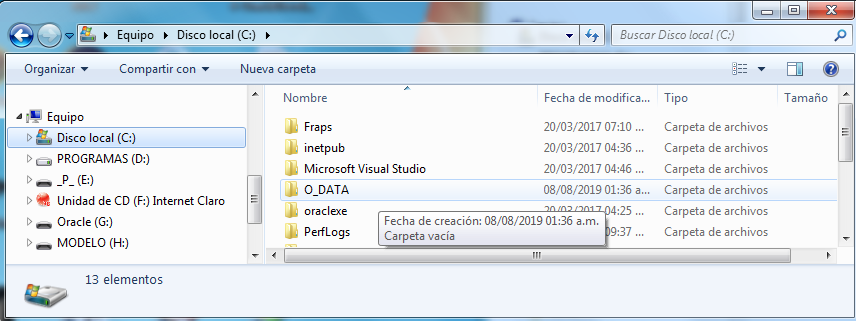
Ingrese a la herramienta PL/SQL e ingrese con el usuario SYSTEM y la contraseña: oracle, ya que este usuario tiene altos privilegios para poder crear la cuenta de usuario.

SUGERENCIA:

En la creación de la cuenta de usuario se recomienda, asignarlo a un TableSpace o Esquema, este debe ser creado primero, para que tenga un mejor propósito la creación de la cuenta de usuario.

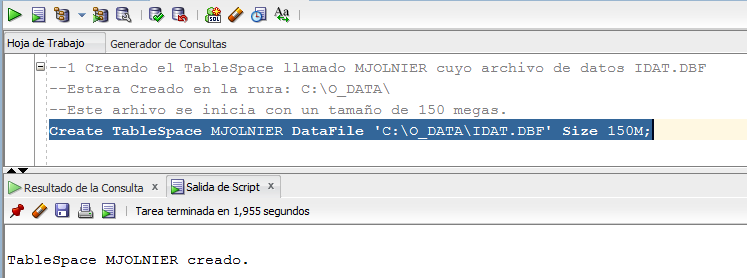
PASO 2:

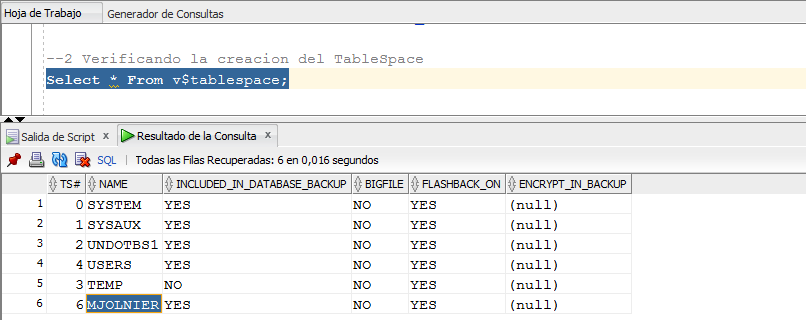
Crear una carpeta en su disco físico (C:\ D:\ o el que disponga), en este caso creara la carpeta O\_DATA, en el disco C:\. Tal como se muestra e la imagen.

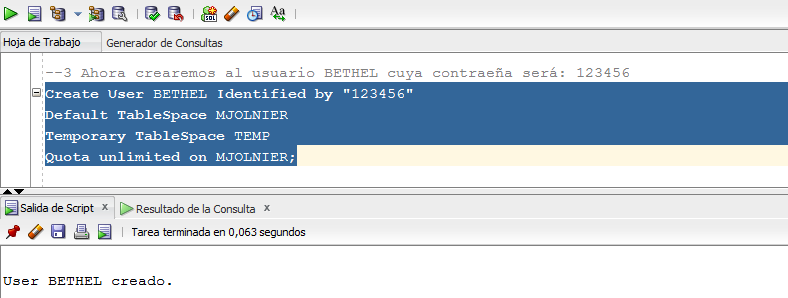


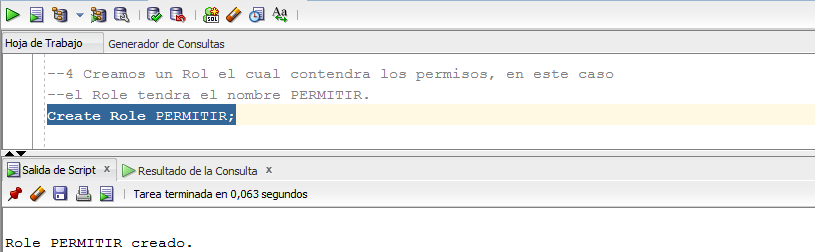
PASO 3:

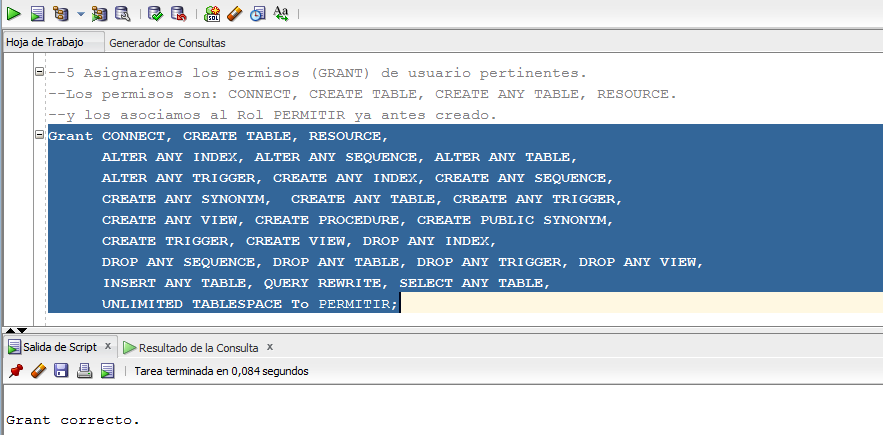
Estando en la herramienta PL/SQL, redacte y ejecute cada bloque seleccionado, de los siguientes códigos para crear la cuenta de usuario.

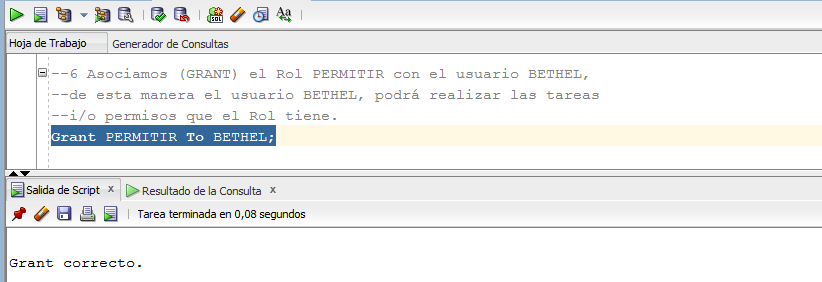












CODIGO COMPLETO DE LA ACTIVIDAD.

--1 Creando el TableSpace llamado MJOLNIER cuyo archivo de datos IDAT.DBF

--Estará Creado en la ruta: C:\O\_DATA\

--Este archivo se inicia con un tamaño de 150 megas.

**Create TableSpace MJOLNIER DataFile 'C:\O\_DATA\IDAT.DBF' Size 150M;**

--2 Verificando la creación del TableSpace

**Select \* From v$tablespace;**

--3 Ahora crearemos al usuario BETHEL cuya contraseña será: 123456

**Create User BETHEL Identified by "123456"**

**Default TableSpace MJOLNIER**

**Temporary TableSpace TEMP**

**Quota unlimited on MJOLNIER;**

--4 Creamos un Rol el cual contendrá los permisos, en este caso

--el Role tendrá el nombre PERMITIR.

**Create Role PERMITIR;**

--5 Asignaremos los permisos (GRANT) de usuario pertinentes.

--Los permisos son: CONNECT, CREATE TABLE, CREATE ANY TABLE, RESOURCE.

--y otros, los asociamos al Rol PERMITIR ya antes creado.

**Grant CONNECT, CREATE TABLE, RESOURCE,**

**ALTER ANY INDEX, ALTER ANY SEQUENCE, ALTER ANY TABLE,**

**ALTER ANY TRIGGER, CREATE ANY INDEX, CREATE ANY SEQUENCE,**

**CREATE ANY SYNONYM, CREATE ANY TABLE, CREATE ANY TRIGGER,**

**CREATE ANY VIEW, CREATE PROCEDURE, CREATE PUBLIC SYNONYM,**

**CREATE TRIGGER, CREATE VIEW, DROP ANY INDEX,**

**DROP ANY SEQUENCE, DROP ANY TABLE, DROP ANY TRIGGER, DROP ANY VIEW,**

**INSERT ANY TABLE, QUERY REWRITE, SELECT ANY TABLE,**

**UNLIMITED TABLESPACE To PERMITIR;**

--6 Asociamos (GRANT) el Rol PERMITIR con el usuario BETHEL,

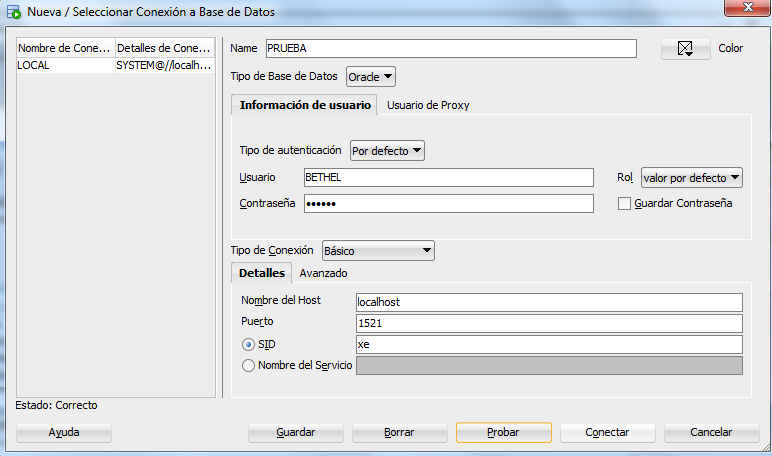
--de esta manera el usuario BETHEL, podrá realizar las tareas

--i/o permisos que el Rol tiene.

**Grant PERMITIR To BETHEL;**

PASO 4:

En la ventana de PL/SQL favor de crear el perfil de conexión con el nombre PRUEBA, donde usara al usuario BETHEL y la contraseña 123456. Tal como se muestra en la siguiente imagen.



1. Ingrese el Nombre del Perfil, en este caso PRUEBA.
2. Ingrese nombre de usuario: BETHEL y contraseña: 123456
3. Probar la Conexión.
4. Observar el estado, si es Correcto, proceda a realizar el paso 5.
5. Conectar.

CONCLUSIONES Y RECOMENDACIONES DE LA EXPERIENCIA

* El objetivo de este capítulo es comprender las potencialidades de oracle al registrar cada acción que el usuario realiza, es decir desde que se crean tablas, constructores (validaciones), índices, secuencias, sinónimos, entre otros y llevar un control de los componentes recientemente creados por el usuario de manera exitosa.
* El estudiante maneja la información de la base de datos Oracle y configura componentes que quedaran automáticamente registrados en un diccionario de la base de datos, mediante las herramientas SQL \*PLUS y PL/SQL, gestionando una instancia de base de datos.

ACTIVIDAD VIRTUAL

* Ingresa a la plataforma virtual, revisa y analizar el material presentado en esta sesión (Semana 2). Luego responde las preguntas propuestas:

1. ¿Qué es un Modelo Relacional?
2. ¿Qué es una Restricción?
   * Defina: CONSTRAINT, UNIQUE, CHECK, IN, DEFAULT
3. ¿Qué entiende Ud. Por Diccionario de datos?
4. ¿Es importante un Diccionario de datos?
5. Mencione Ud. Algunas tablas del diccionario de datos, las más importantes
6. Mediante el comando SELECT Muestre los Índices de una determinada tabla.
7. Mediante el comando SELECT Muestre las Secuencias de una determinada tabla.
8. Mediante el comando SELECT Muestre los comentarios a nivel de columna de una determinada tabla.
9. Mediante el comando SELECT Muestre los TableSpace de la base de datos.
10. Mediante el comando SELECT ver tablas, vistas, sinónimos y secuencias propiedad del usuario.