

**Universidad Nacional**



**Escuela de Informática**

**Profesor: Johnny Villalobos**

**Curso: Administración de Bases de Datos**

**Tema: Primer Informe de Laboratorio**

**Estudiante:  
Oscar Fabián Chacón Bolaños / 402230694**

**II Ciclo 2014**

## 1. Documentación la base de datos

Sistema gestor de base de datos	Oracle 11g
Sistema Operativo	Windows 7
Nombre de la base de datos	BD1
Archivos de control	<ul style="list-style-type: none"><li>Nombre: control_files</li></ul> Tipo: string Valor: C:\BD1\CONTROLADORES\CONTROL.DBF
Redo log on line	1      ONLINE C:\BD1\BITACORAS\REDO01.LOG NO 2      ONLINE C:\BD1\BITACORAS\REDO02.LOG NO
Tablespace datos usuario	C:\BD1\datos\Contabilidad\CONTABILIDAD.DBF C:\BD1\datos\Produccion\PRODUCCION.DBF C:\BD1\datos\Ventas\VENTAS.DBF
Data files	UNDOTBS1 SYSAUX SYSTEM
Tablespace sistema	C:\BD1\system\SYSTEM.DBF

### **Evidencia de la Documentación de la Base de Datos:**

El Sistema Gestor de Base de Datos es la Oracle 11g. El Sistema Operativo es Windows 7.

Para lograr obtener el nombre de la Base de Datos, colocamos los siguientes parámetros mostrados en la captura de pantalla:

```
Run SQL Command Line
SQL>
SQL>
SQL>
SQL>
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
xe
1 row selected.

SQL>
SQL>
SQL>
SQL> select * from global_name;

GLOBAL_NAME
-----
BD1
1 row selected.

SQL>
```

Como podemos ver, la instancia es xe y el NOMBRE de la Base de Datos es: BD1.

Para lograr obtener los archivos de control de la base de datos se coloca el comando show parameters control file.

```
Run SQL Command Line
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL> show parameters control file;

NAME                                TYPE      VALUE
-----                                -
control_file_record_keep_time      integer    7
control_files                       string     C:\BD1\CONTROLADORES\CONTROL.D
control_management_pack_access      string     BF
control_management_pack_access      string     NONE
SQL>
```

Para lograr obtener los redo log online lo se coloca el comando: select \* from v\$logfile;

```
Run SQL Command Line
SQL>
SQL>
SQL>
SQL>
SQL> select * from v$logfile;
      GROUP# STATUS  TYPE
-----
MEMBER
-----
IS_
-----
          1      ONLINE
C:\BD1\BITACORAS\REDO01.LOG
NO
          2      ONLINE
C:\BD1\BITACORAS\REDO02.LOG
NO

2 rows selected.
SQL>
```

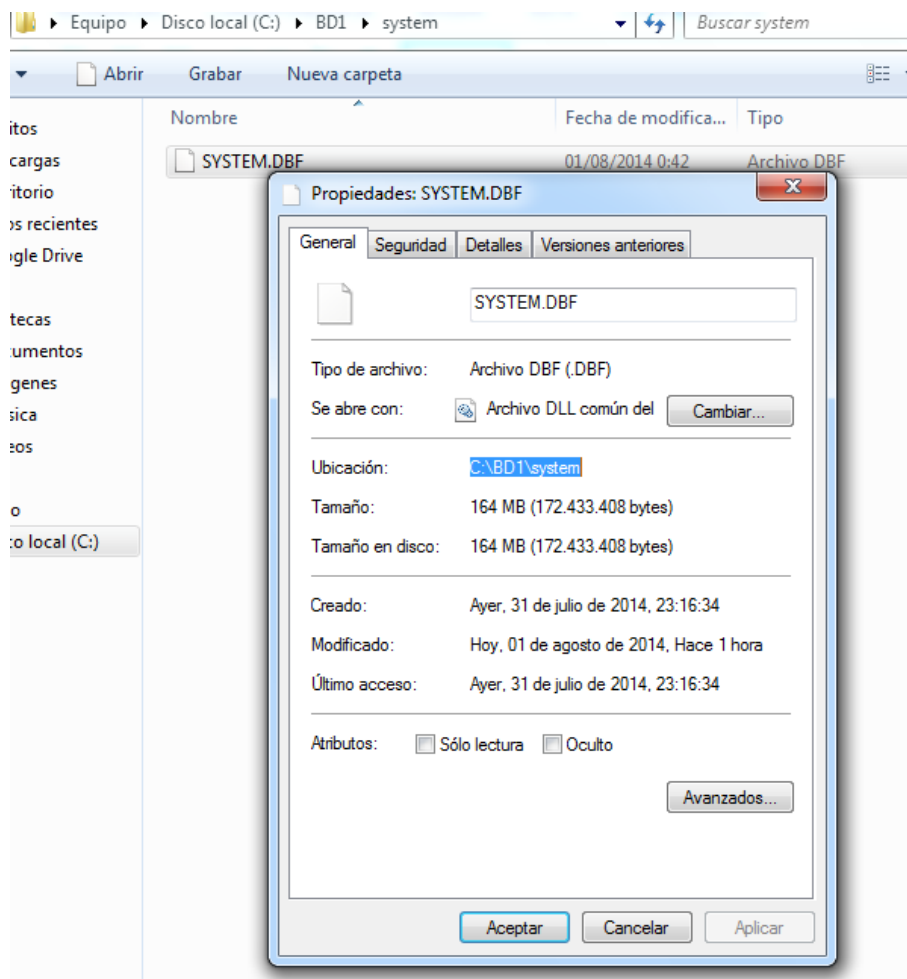
Para obtener los datafiles de la base de datos creada, se coloca el comando: select tablespace\_name from dba\_data\_files;

```
Run SQL Command Line
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL> select tablespace_name from dba_data_files;
TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDO1BS1

3 rows selected.
SQL>
```

En esta imagen se muestra la evidencia de los comandos de cómo crear una base de datos.

Para obtener el tablespace del sistema copiamos la ruta donde se encuentra la misma:  
C:\BD1\system\SYSTEM.DBF



## Creación de la Base de Datos:

A continuación se muestra la evidencia de los comandos de cómo crear una base de datos.

```

Run SQL Command Line
SQL*Plus: Release 11.2.0.2.0 Production on Jue Jul 31 23:12:49 2014
Copyright (c) 1982, 2010, Oracle. All rights reserved.

SQL> connect sys as sysdba
Enter password:
Connected.
SQL> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
SQL>
SQL> @C:\BD1\BD1.sql
ORACLE instance started.

Total System Global Area 564957184 bytes
Fixed Size 1384956 bytes
Variable Size 348130820 bytes
Database Buffers 209715200 bytes
Redo Buffers 5726208 bytes

Database created.
SQL>

```

Donde los pasos son:

1. Inicialmente creamos una carpeta BD1.

Archivos de programa	31/07/2014 13:08	Carpeta de archivos
BD1	31/07/2014 16:00	Carpeta de archivos
dell	03/04/2014 0:21	Carpeta de archivos
drvrtmp	03/04/2014 0:39	Carpeta de archivos
Hacienda	15/07/2014 19:40	Carpeta de archivos
Intel	03/04/2014 0:23	Carpeta de archivos
oraclexe	13/05/2014 0:47	Carpeta de archivos
PerfLogs	13/07/2009 20:37	Carpeta de archivos
Usuarios	02/04/2014 4:40	Carpeta de archivos
Windows	29/07/2014 18:28	Carpeta de archivos

2. Dentro de esa carpeta crearemos las respectivas carpetas para los archivos de la Base de Datos.

Bitacoras	31/07/2014 23:16	Carpeta de archivos
controladores	31/07/2014 23:16	Carpeta de archivos
datos	30/07/2014 21:58	Carpeta de archivos
parametros	31/07/2014 23:07	Carpeta de archivos
sysaux	31/07/2014 23:16	Carpeta de archivos
system	31/07/2014 23:16	Carpeta de archivos
TEMP	31/07/2014 23:16	Carpeta de archivos
UNDO	31/07/2014 23:16	Carpeta de archivos
BD1	31/07/2014 16:29	Archivo SQL

3. Luego iniciamos el proceso de la creación de la base mediante comandos. Primero nos conectamos a la Base de Datos local.

```
SQL> connect sys as sysdba
Enter password:
Connected.
```

4. Luego botamos la actual base de datos.

```
SQL> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
```

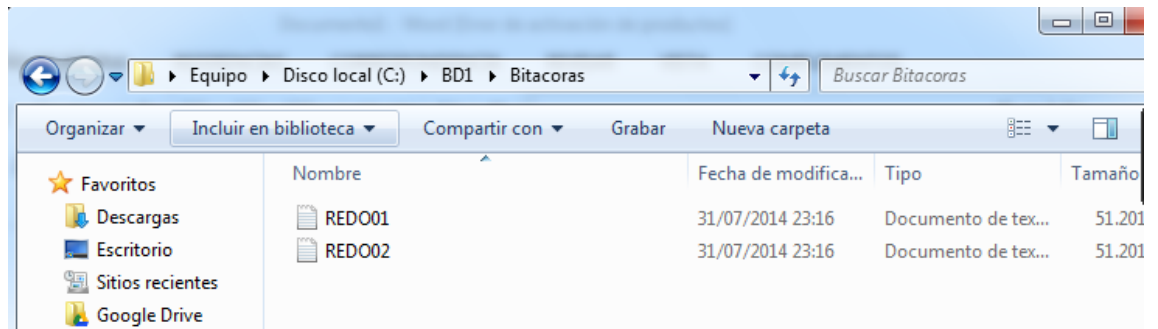
5. Seguidamente ejecutamos el script de la creación de la base de datos. Creándose con éxito de esta manera:

```
SQL> @C:\BD1\BD1.sql
ORACLE instance started.

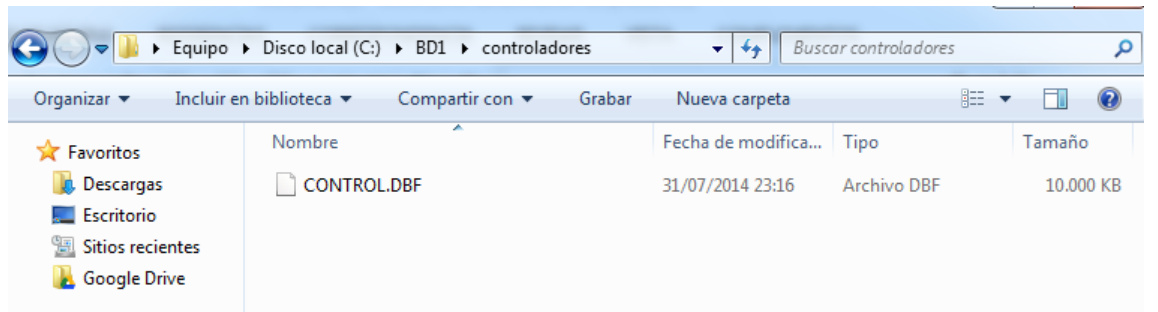
Total System Global Area 564957184 bytes
Fixed Size 1384956 bytes
Variable Size 348130820 bytes
Database Buffers 209715200 bytes
Redo Buffers 5726208 bytes

Database created.
```

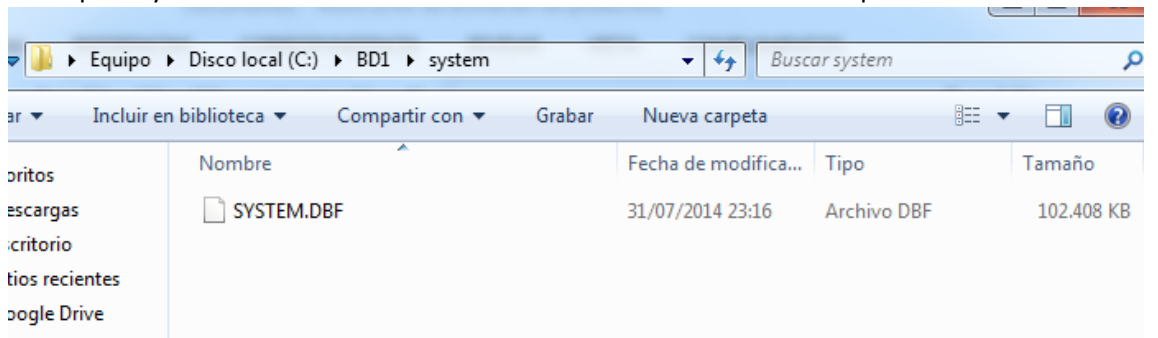
6. La carpeta Bitácoras contendrá los Redo log files, donde observamos que sí se crearon:



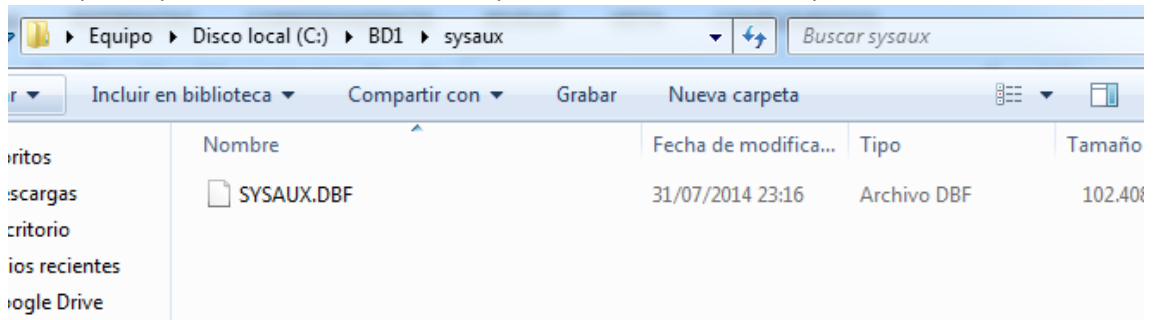
7. La carpeta Controladores contendrá los Archivos de control donde observamos que sí se creó:



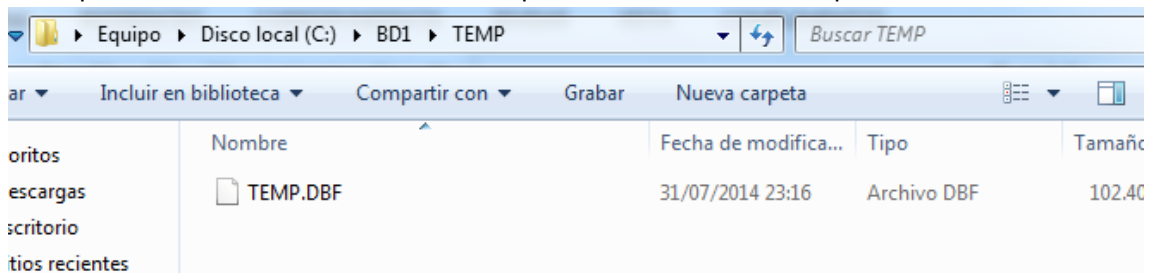
8. La carpeta system contendrá el archivo de sistema donde observamos que sí se creó:



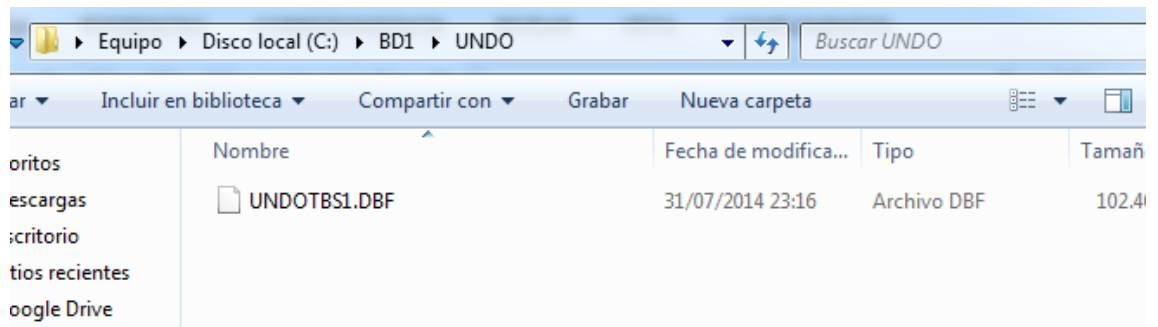
9. La carpeta sysaux contendrá el archivo sysaux donde observamos que sí se creó:



10. La carpeta TEMP contendrá el archivo temp.dbf donde observamos que sí se creó:



11. La carpeta UNDO contendrá el archivo undotbs1 donde observamos que sí se creó



### **Comprobación del Paso del Catalog:**

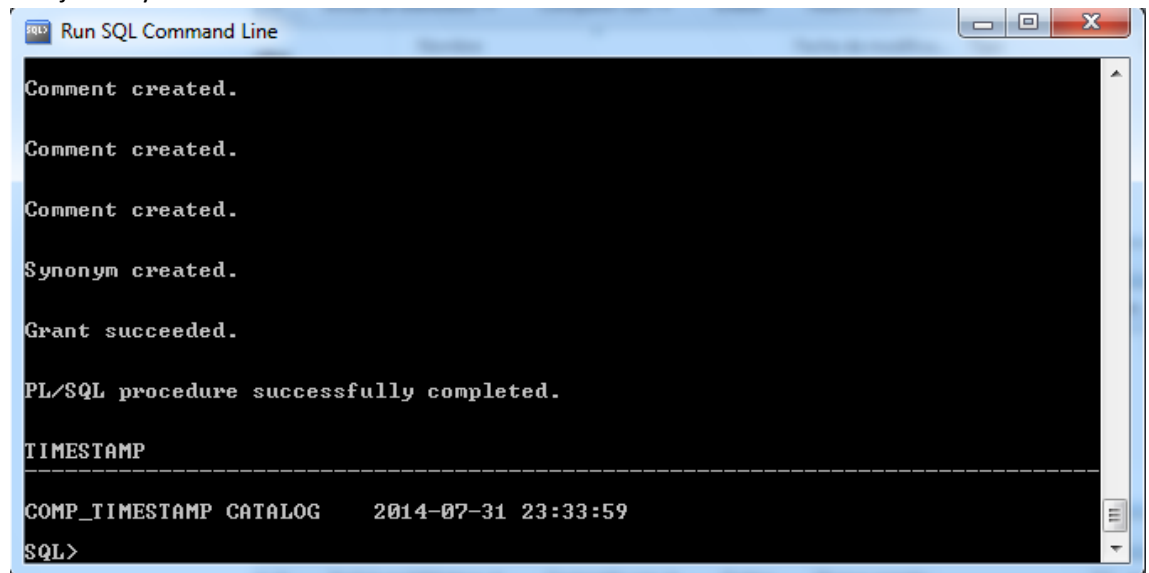
Lo iniciaremos con un comando a continuación:

@C:\oracle\app\oracle\product\11.2.0\server\rdbms\admin\catalog.sql

Luego se evidencia que el proceso de ejecutar el catalog se realizó con éxito.

PD: Al ser tan largo se muestra el inicio y el final del proceso mediante el pantallazo.

Se ejecuta y termina con éxito:



### **Comprobación del Paso del Catproc:**

Lo iniciaremos con un comando a continuación:

@C:\oracle\app\oracle\product\11.2.0\server\rdbms\admin\catproc.sql

Luego se evidencia que el proceso de ejecutar el catalog se realizó con éxito.

PD: Al ser tan largo se muestra el inicio y el final del proceso mediante los dos pantallazos.

Se ejecuta:



```
Run SQL Command Line

Comment created.

Synonym created.

Grant succeeded.

PL/SQL procedure successfully completed.

TIMESTAMP
-----
COMP_TIMESTAMP CATALOG      2014-07-31 23:33:59

SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL> @C:\oracle\app\oracle\product\11.2.0\server\rdbms\admin\catproc.sql
```

Termina con éxito:

```
Run SQL Command Line

3      dbms_registry.schema_list_t(<'SYSTEM', 'OUTLN', 'DBSNMP', 'DIP',
4                                     'ORACLE_OCM', 'APPQOSSYS'>);
5      dbms_registry.loaded('CATPROC');
6      dbms_registry_sys.validate_catproc;
7      dbms_registry_sys.validate_catalog;
8  END;
9  /

PL/SQL procedure successfully completed.

SQL>
SQL> SELECT dbms_registry_sys.time_stamp('CATPROC') AS timestamp FROM DUAL;

TIMESTAMP
-----
COMP_TIMESTAMP CATPROC      2014-08-01 00:42:06

1 row selected.

SQL>
SQL> SET SERVEROUTPUT OFF
SQL>
SQL>
SQL>
```

Se realiza el alter database open. En este caso, ya estaba abierta la base.

```
SQL> alter database open
2 ;
alter database open
*
ERROR at line 1:
ORA-01531: a database already open by the instance

SQL>
```

### Se Adjunta el Archivo Init.ora:

xe.\_\_java\_pool\_size=4194304

xe.\_\_large\_pool\_size=8388608

xe.\_\_oracle\_base='C:\oracle\app\oracle' #ORACLE\_BASE set from environment

```

xe.__pga_aggregate_target=188743680
xe.__sga_target=566231040
xe.__shared_io_pool_size=92274688
xe.__shared_pool_size=335544320
xe.__streams_pool_size=0
*.audit_file_dest='C:\BD1\adump'
*.compatible='11.2.0.0.0'
*.control_files='C:\BD1\controladores\control.dbf'
*.db_name='BD1'
#*.DB_RECOVERY_FILE_DEST_SIZE=10G
#*.DB_RECOVERY_FILE_DEST='C:\BD1'
#*.diagnostic_dest='C:\oracle\app\oracle'
#####
#*.dispatchers='(PROTOCOL=TCP) (SERVICE=XETESTXDB)'
#*.job_queue_processes=4
#*.local_listener='(ADDRESS=(PROTOCOL=TCP)(HOST=AZAR-PC.ace-#ins.com)(PORT=1522))'
*.open_cursors=300
*.os_authent_prefix=''
*.pga_aggregate_target=180M
*.remote_login_passwordfile='EXCLUSIVE'
*.sessions=20
*.sga_target=540M
*.shared_servers=4
*.undo_management='AUTO'
*.undo_tablespace='UNDOTBS1'

```

### **Se Adjunta el Archivo BD1.sql:**

```

Startup nomount pfile=C:\BD1\parametros\initbd1.ora
create database BD1
LOGFILE

```

```
GROUP 1 'C:\BD1\Bitacoras\REDO01.LOG' SIZE 50M BLOCKSIZE 512,  
GROUP 2 'C:\BD1\Bitacoras\REDO02.LOG' SIZE 50M BLOCKSIZE 512  
DATAFILE 'C:\BD1\system\SYSTEM.DBF' size 100m autoextend on  
sysaux datafile 'C:\BD1\sysaux\SYSAUX.DBF' size 100m autoextend on  
undo tablespace undotbs1 datafile 'C:\BD1\undo\UNDOTBS1.DBF' size 100m autoextend on  
default temporary tablespace temp tempfile 'C:\BD1\TEMP\TEMP.dbf' size 100M  
CHARACTER SET AL32UTF8  
;
```

### **Comprobación de la Ejecución del Script de Creación del Modelo:**

```
Run SQL Command Line
Table created.
SQL> CREATE TABLE FACTURA_COMPRA(
2     numero NUMBER(4),
3     fecha DATE,
4     total NUMBER(10,2),
5     impuesto NUMBER(10,2),
6     subtotal NUMBER(10,2),
7     proveedor NUMBER(4),
8     saldo NUMBER(10,2),
9     CONSTRAINT PKFACTURA_COMPRA PRIMARY KEY (numero),
10    CONSTRAINT FK1FACTURA_COMPRA FOREIGN KEY (proveedor) REFERENCES PROVEEDOR (codigo)
11 )TABLESPACE Contabilidad;
Table created.
SQL> CREATE TABLE LINEAS_FACTURA_COMPRA(
2     factura NUMBER(4),
3     componente NUMBER(4),
4     cantidad NUMBER(10,2),
5     precio NUMBER(10,2),
6     impuesto NUMBER(10,2),
7     descuento NUMBER(10,2),
8     CONSTRAINT PK1LINEAS_FACTURA_COMPRA PRIMARY KEY (factura, componente),
9     CONSTRAINT FK1LINEAS_FACTURA_COMPRA FOREIGN KEY (componente) REFERENCES COMPONENTE (codigo),
10    CONSTRAINT FK2LINEAS_FACTURA_COMPRA FOREIGN KEY (factura) REFERENCES FACTURA_COMPRA (numero)
11 )TABLESPACE Contabilidad;
Table created.
SQL> CREATE TABLE PAGOS(
2     recibo NUMBER(4),
3     cuenta NUMBER(4),
4     fecha DATE,
5     monto NUMBER(10,2),
6     CONSTRAINT PKPAGOS PRIMARY KEY (recibo),
7     CONSTRAINT FK1PAGOS FOREIGN KEY (cuenta) REFERENCES CUENTAXPAGAR (cuenta)
8 )TABLESPACE Contabilidad;
Table created.
SQL> CREATE TABLE PAGO_FACTURA(
2     recibo NUMBER(4),
3     factura NUMBER(4),
4     monto NUMBER(10,2),
5     CONSTRAINT PK1PAGO_FACTURA PRIMARY KEY (recibo, factura),
6     CONSTRAINT FK1PAGO_FACTURA FOREIGN KEY (factura) REFERENCES FACTURA_COMPRA (numero),
7     CONSTRAINT FK2PAGO_FACTURA FOREIGN KEY (recibo) REFERENCES PAGOS (recibo)
8 )TABLESPACE Contabilidad;
Table created.
SQL>
```

Inserts a al menos 4 tablas:

```
Run SQL Command Line
SQL>
SQL>
SQL> insert into AREADEVENTAS values (4, 'Area04');
1 row created.
SQL> insert into AREADEVENTAS values (2, 'Area02');
1 row created.
SQL> insert into AREADEVENTAS values (1, 'Area01');
1 row created.
SQL> insert into VENDEDOR values (420, 'Maria', 20.6, 70000, 4);
1 row created.
SQL> insert into VENDEDOR values (908, 'Jorge', 19.77, 50000, 2);
1 row created.
SQL> insert into VENDEDOR values (547, 'Oscar', 9.9, 900000, 1);
1 row created.
SQL> insert into ZONA values (001, 'Zona 56');
1 row created.
SQL> insert into ZONA values (002, 'Zona 77');
1 row created.
SQL> insert into ZONA values (003, 'zona 34');
1 row created.
SQL> insert into BODEGA values (56, 001);
1 row created.
SQL> insert into BODEGA values (29, 002);
1 row created.
SQL> insert into BODEGA values (37, 003);
1 row created.
```

Se muestran:

```
SQL> select * from VENDEDOR;
```

CODIGO	NOMBRE	COMISION	VENTA	AREA
420	Maria	20.6	70000	4
908	Jorge	19.77	50000	2
547	Oscar	9.9	900000	1

```
3 rows selected.
```

```
SQL> select * from AREADEVENTAS;
```

CODIGO	NOMBRE
4	Area04
2	Area02
1	Area01

```
3 rows selected.
```

```
SQL> select * from ZONA;
```

CODIGO	DESCRIPCION
1	Zona 56
2	Zona 77
3	zona 34

```
3 rows selected.
```

```
SQL> select * from BODEGA;
```

CODIGO	ZONA
56	1
29	2
37	3

```
3 rows selected.
```

```
SQL>
```

### Se Adjunta el Archivo Lab1.sql (Modelo):

-- Todas las Tablas del Modelo.

```
CREATE TABLESPACE Contabilidad DATAFILE 'C:\BD1\datos\Contabilidad\contabilidad.dbf'
size 3M reuse autoextend on next 5M maxsize 150M;
```

```
CREATE TABLESPACE Produccion DATAFILE 'C:\BD1\datos\Produccion\produccion.dbf' size
3M reuse autoextend on next 5M maxsize 150M;
```

```
CREATE TABLESPACE Ventas DATAFILE 'C:\BD1\datos\Ventas\ventas.dbf' size 3M reuse
autoextend on next 5M maxsize 150M;
```

```
CREATE TABLE AREADEVENTAS(
    codigo NUMBER(4),
    nombre VARCHAR(30),
    CONSTRAINT PKAREADEVENTAS PRIMARY KEY (codigo)
)TABLESPACE Ventas;
```

```
CREATE TABLE VENDEDOR(
    codigo NUMBER(4),
    nombre VARCHAR(30),
```

```

comision NUMBER(10,2),

venta NUMBER(10,2),

area NUMBER(4),

CONSTRAINT PKVENDEDOR PRIMARY KEY (codigo),

CONSTRAINT FKVENDEDOR FOREIGN KEY (area) REFERENCES AREADEVENTAS (codigo)

)TABLESPACE Ventas;

CREATE TABLE CLIENTE(

    codigo NUMBER(4),

    nombre VARCHAR(30),

    telefono VARCHAR(20),

    limiteCredito NUMBER(10,2),

    compra NUMBER(10,2),

    descuento NUMBER(10,2),

    Vendedor NUMBER(4),

    CONSTRAINT PKCLIENTE PRIMARY KEY (codigo),

    CONSTRAINT FKCLIENTE FOREIGN KEY (Vendedor) REFERENCES VENDEDOR (codigo)

)TABLESPACE Ventas;

CREATE TABLE CUENTAXCOBRAR(

    cuenta NUMBER(4),

    Cliente NUMBER(4),

    total NUMBER(10,2),

    saldo NUMBER(10,2),

    estado CHAR,

    CONSTRAINT PKCUENTAXCOBRAR PRIMARY KEY (cuenta),

    CONSTRAINT FKCUENTAXCOBRAR FOREIGN KEY (Cliente) REFERENCES CLIENTE (codigo)

)TABLESPACE Contabilidad;

CREATE TABLE COBROS(

    recibo NUMBER(4),

    cuenta NUMBER(4),

    fecha DATE,

    monto NUMBER(10,2),

```

```

        CONSTRAINT PKCOBROS PRIMARY KEY (Recibo),
        CONSTRAINT FKCOBROS FOREIGN KEY (cuenta) REFERENCES CUENTAXCOBRAR (cuenta)
)TABLESPACE Contabilidad;

CREATE TABLE FACTURA_VENTA(
    numero NUMBER(4),
    fecha DATE,
    total NUMBER(10,2),
    impuesto NUMBER(10,2),
    subtotal NUMBER(10,2),
    Cliente NUMBER(4),
    saldo NUMBER(10,2),
    CONSTRAINT PKFACTURA_VENTA PRIMARY KEY (numero),
    CONSTRAINT FKFACTURA_VENTA FOREIGN KEY (Cliente) REFERENCES CLIENTE (codigo)
)TABLESPACE Contabilidad;

CREATE TABLE COBRO_FACTURA(
    recibo NUMBER(4),
    factura NUMBER(4),
    monto NUMBER(10,2),
    CONSTRAINT PKCOBRO_FACTURA PRIMARY KEY (recibo, factura),
    CONSTRAINT FK1COBRO_FACTURA FOREIGN KEY (recibo) REFERENCES COBROS (recibo),
    CONSTRAINT FK2COBRO_FACTURA FOREIGN KEY (factura) REFERENCES FACTURA_VENTA
(numero)
)TABLESPACE Contabilidad;

CREATE TABLE ZONA(
    codigo NUMBER(4),
    descripcion VARCHAR(30),
    CONSTRAINT PKZONA PRIMARY KEY (codigo)
)TABLESPACE Produccion;

CREATE TABLE BODEGA(
    codigo NUMBER(4),
    zona NUMBER(4),

```



```

        CONSTRAINT PKBODEGA PRIMARY KEY (codigo),

        CONSTRAINT FKBODEGA FOREIGN KEY (zona) REFERENCES ZONA (codigo)

)TABLESPACE Produccion;

CREATE TABLE PRODUCTOS(

    codigo NUMBER(4),

    descripcion VARCHAR(30),

    costo NUMBER(10,2),

    utilidad NUMBER(10,2),

    grabado CHAR,

    CONSTRAINT PKPRODUCTOS PRIMARY KEY (codigo)

)TABLESPACE Produccion;

CREATE TABLE LINEAS_FACTURA(

    factura NUMBER(4),

    articulo NUMBER(4),

    cantidad NUMBER(10,2),

    precio NUMBER(10,2),

    impuesto NUMBER(10,2),

    descuento NUMBER(10,2),

    CONSTRAINT PKLINEAS_FACTURA PRIMARY KEY (factura, articulo),

    CONSTRAINT FK1LINEAS_FACTURA FOREIGN KEY (factura) REFERENCES FACTURA_VENTA
(numero),

    CONSTRAINT FK2LINEAS_FACTURA FOREIGN KEY (articulo) REFERENCES PRODUCTOS
(codigo)

)TABLESPACE Contabilidad;

CREATE TABLE PROVEEDOR(

    codigo NUMBER(4),

    nombre VARCHAR(30),

    telefono VARCHAR(20),

    CONSTRAINT PKPROVEEDOR PRIMARY KEY (codigo)

)TABLESPACE Produccion;

CREATE TABLE CUENTAXPAGAR(

    cuenta NUMBER(4),

```

```

    Proveedor NUMBER(4),

    total NUMBER(10,2),

    saldo NUMBER(10,2),

    estado CHAR,

    CONSTRAINT PKCUENTAXPAGAR PRIMARY KEY (cuenta),

    CONSTRAINT FKCUENTAXPAGAR FOREIGN KEY (Proveedor) REFERENCES PROVEEDOR
(codigo)
)TABLESPACE Contabilidad;

CREATE TABLE INVENTARIO_ARTICULOS(

    articulo NUMBER(4),

    bodega NUMBER(4),

    cantidad NUMBER(10,2),

    cantMinima NUMBER(10,2),

    cantMaxima NUMBER(10,2),

    CONSTRAINT PK1INVENTARIO_ARTICULOS PRIMARY KEY (articulo, bodega),

    CONSTRAINT FK1INVENTARIO_ARTICULOS FOREIGN KEY (articulo) REFERENCES
PRODUCTOS (codigo),

    CONSTRAINT FK2INVENTARIO_ARTICULOS FOREIGN KEY (bodega) REFERENCES BODEGA
(codigo)
)TABLESPACE Produccion;

CREATE TABLE COMPONENTE(

    codigo NUMBER(4),

    nombre VARCHAR(30),

    precio NUMBER(10,2),

    CONSTRAINT PKCOMPONENTE PRIMARY KEY (codigo)
)TABLESPACE Produccion;

CREATE TABLE FORMULA(

    componente NUMBER(4),

    cantidad NUMBER(10,2),

    producto NUMBER(4),

    CONSTRAINT PKFORMULA PRIMARY KEY (componente, producto),

    CONSTRAINT FK1FORMULA FOREIGN KEY (producto) REFERENCES PRODUCTOS (codigo),

```

```

        CONSTRAINT FK2FORMULA FOREIGN KEY (componente) REFERENCES COMPONENTE
(codigo)
)TABLESPACE Contabilidad;

CREATE TABLE INVENTARIO_COMPONENTES(
    componente NUMBER(4),
    bodega NUMBER(4),
    cantidad NUMBER(10,2),
    cantMinima NUMBER(10,2),
    cantMaxima NUMBER(10,2),
    CONSTRAINT PKINVENTARIO_COMPONENTES PRIMARY KEY (componente, bodega),
    CONSTRAINT FK1INVENTARIO_COMPONENTES FOREIGN KEY (bodega) REFERENCES
BODEGA (codigo),
    CONSTRAINT FK2INVENTARIO_COMPONENTES FOREIGN KEY (componente) REFERENCES
COMPONENTE (codigo)
)TABLESPACE Produccion;

CREATE TABLE FACTURA_COMPRA(
    numero NUMBER(4),
    fecha DATE,
    total NUMBER(10,2),
    impuesto NUMBER(10,2),
    subtotal NUMBER(10,2),
    proveedor NUMBER(4),
    saldo NUMBER(10,2),
    CONSTRAINT PKFACTURA_COMPRA PRIMARY KEY (numero),
    CONSTRAINT FK1FACTURA_COMPRA FOREIGN KEY (proveedor) REFERENCES PROVEEDOR
(codigo)
)TABLESPACE Contabilidad;

CREATE TABLE LINEAS_FACTURA_COMPRA(
    factura NUMBER(4),
    componente NUMBER(4),
    cantidad NUMBER(10,2),
    precio NUMBER(10,2),

```

```

    impuesto NUMBER(10,2),

    descuento NUMBER(10,2),

    CONSTRAINT PK1LINEAS_FACTURA_COMPRA PRIMARY KEY (factura, componente),

    CONSTRAINT FK1LINEAS_FACTURA_COMPRA FOREIGN KEY (componente) REFERENCES
COMPONENTE (codigo),

    CONSTRAINT FK2LINEAS_FACTURA_COMPRA FOREIGN KEY (factura) REFERENCES
FACTURA_COMPRA (numero)

)TABLESPACE Contabilidad;

CREATE TABLE PAGOS(

    recibo NUMBER(4),

    cuenta NUMBER(4),

    fecha DATE,

    monto NUMBER(10,2),

    CONSTRAINT PKPAGOS PRIMARY KEY (recibo),

    CONSTRAINT FK1PAGOS FOREIGN KEY (cuenta) REFERENCES CUENTAXPAGAR (cuenta)

)TABLESPACE Contabilidad;

CREATE TABLE PAGO_FACTURA(

    recibo NUMBER(4),

    factura NUMBER(4),

    monto NUMBER(10,2),

    CONSTRAINT PK1PAGO_FACTURA PRIMARY KEY (recibo, factura),

    CONSTRAINT FK1PAGO_FACTURA FOREIGN KEY (factura) REFERENCES FACTURA_COMPRA
(numero),

    CONSTRAINT FK2PAGO_FACTURA FOREIGN KEY (recibo) REFERENCES PAGOS (recibo)

)TABLESPACE Contabilidad;

insert into AREADEVENTAS values (4, 'Area04');

insert into AREADEVENTAS values (2, 'Area02');

insert into AREADEVENTAS values (1, 'Area01');

insert into VENDEDOR values (420, 'Maria', 20.6, 70000, 4);

insert into VENDEDOR values (908, 'Jorge', 19.77, 50000, 2);

insert into VENDEDOR values (547, 'Oscar', 9.9, 900000, 1);

insert into ZONA values (001, 'Zona 56');

```

insert into ZONA values (002, 'Zona 77');

insert into ZONA values (003, 'zona 34');

insert into BODEGA values (56, 001);

insert into BODEGA values (29, 002);

insert into BODEGA values (37, 003);