

**Universidad Nacional**



**Escuela de Informática**

**Profesor:**

**Johnny Villalobos**

**Curso:**

**Administración de Bases de Datos**

**Tema:**

**Laboratorio 1**

**Estudiante:**

**José Pablo Céspedes Castro**

**II Ciclo 2014**

## 1. Documentación la base de datos

Sistema gestor de base de datos	Oracle Data Base
Sistema Operativo	Windows 7
Nombre de la base de datos	BD1
Archivos de control	C:\BD1\controladoras\CONTROL.DBF
Redo log on line	C:\BD1\bitacoras\REDO01.LOG
	C:\BD1\bitacoras \REDO02.LOG
Tablespace datos usuario	C:\BD1\datos\contabilidad\contabilidad.dbf
	C:\BD1\datos\ventas\ventas.dbf
	C:\BD1\datos\produccion\produccion.dbf
Data file	SYSTEM
	SYSAUX
	UNDOTBS1
Tablespace sistema	C:\BD1\system\ SYSTEM01.DBF

### 1.1 Pruebas de la documentación de la base de datos.

#### 1.1.1 Control Files

```
SQL> show parameters control file
```

NAME	TYPE	VALUE
control_file_record_keep_time	integer	7
control_files	string	C:\BD1\CONTROLADORAS\CONTROL.DBF
control_management_pack_access	string	DIAGNOSTIC+TUNING

```
SQL>
```

#### 1.1.2 Redo log on line

```
SQL> SELECT * FROM V$LOGFILE;
```

GROUP#	STATUS	TYPE
MEMBER		
IS_	CON_ID	
1	ONLINE	
C:\BD1\BITACORAS\REDO01.LOG		
NO	0	
2	ONLINE	
C:\BD1\BITACORAS\REDO02.LOG		
NO	0	

```
2 filas seleccionadas.
SQL>
```

### 1.1.3 Data Files

```
SQL> select tablespace_name from dba_data_files;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1

3 filas seleccionadas.
SQL>
```

### 1.1.4 TableSpace Sistema

```
SQL> select tablespace_name from dba_data_files where tablespace_name = 'SYSTEM'
;

TABLESPACE_NAME
-----
SYSTEM

SQL>
```

## 2. Creación de la base de datos.

### C:\BD1\parametros\initbld.ora

```
xe.__db_cache_size=117440512
xe.__java_pool_size=4194304
xe.__large_pool_size=8388608
xe.__oracle_base='C:\app\PabloC'    #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\controladoras\control.dbf'
*.db_name='BD1'
#*.DB_RECOVERY_FILE_DEST_SIZE=10G
#*.DB_RECOVERY_FILE_DEST='C:\base'
#*.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'
```

## C:\BD1\bd1.sql

Startup nomount pfile=C:\BD1\parametros\initbdl.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

default temporary tablespace temp tempfile 'C:\BD1\TEMP\temp.dbf' size 100M

undo tablespace undotbs1 datafile 'C:\BD1\UNDO\UNDOTBS1.DBF' size 100m autoextend on

CHARACTER SET AL32UTF8

```
SQL Plus
Introduzca el nombre de usuario: sys as sysdba
Introduzca la contraseña:

Conectado a:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing opt
ions

SQL> shutdown
ORA-01507: base de datos sin montar

Instancia ORACLE cerrada.
SQL> @C:\BD1\bd1.sql
Instancia ORACLE iniciada.

Total System Global Area 563691520 bytes
Fixed Size                2405040 bytes
Variable Size             381685072 bytes
Database Buffers          171966464 bytes
Redo Buffers              7634944 bytes

Base de datos creada.

SQL> _
```

## C:\app\PabloC\product\12.1.0\dbhome\_1\RDBMS\ADMIN\catalog.sql

```
SQL Plus
Comentario creado.

Comentario creado.

Comentario creado.

Sinónimo creado.

Concesión terminada correctamente.

Vista creada.

Comentario creado.

Comentario creado.

Comentario creado.

Comentario creado.

Sinónimo creado.

Concesión terminada correctamente.

Sesión modificada.

Sesión modificada.

Procedimiento PL/SQL terminado correctamente.

TIMESTAMP
-----
COMP_TIMESTAMP CATALOG      2014-07-31 22:42:00

Sesión modificada.

Sesión modificada.

SQL> _
```

C:\app\PabloC\product\12.1.0\dbhome\_1\RDBMS\ADMIN\catproc  
.sql

```
SQL Plus
SQL> Rem sqlsessend.sql
SQL> Rem
SQL> Rem Copyright (c) 2012, Oracle and/or its affiliates. All rights reserved.
SQL> Rem
SQL> Rem      NAME
SQL> Rem      sqlsessend.sql - SQL session end
SQL> Rem
SQL> Rem      DESCRIPTION
SQL> Rem      Any commands which should be run at the end of all oracle
SQL> Rem      supplied scripts.
SQL> Rem
SQL> Rem      NOTES
SQL> Rem      See sqlsessstart.sql for the corresponding start script.
SQL> Rem
SQL> Rem      MODIFIED   <MM/DD/YY>
SQL> Rem      surman      11/14/12 - 12876907: Common start and end scripts
SQL> Rem      surman      11/14/12 - Created
SQL> Rem
SQL> alter session set "_ORACLE_SCRIPT" = false;
Sesión modificada.

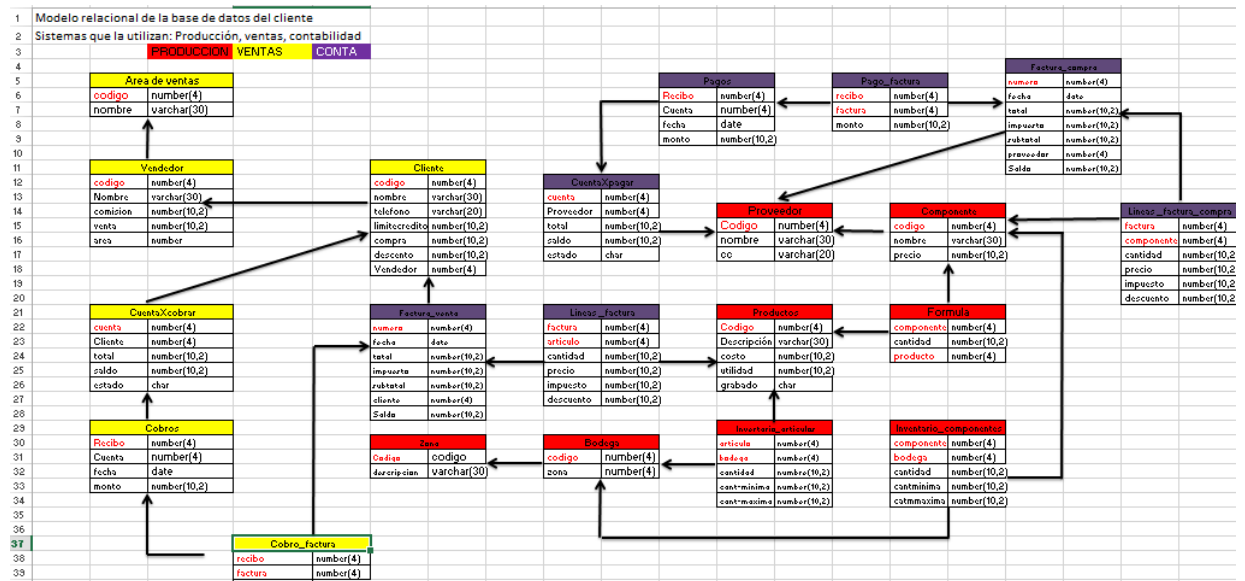
SQL>
SQL> @?/rdbms/admin/sqlsessend.sql
SQL> Rem $Header: rdbms/admin/sqlsessend.sql /st_rdbms_12.1.0.1/1 2013/01/03 10:
22:45 surman Exp $
SQL> Rem
SQL> Rem sqlsessend.sql
SQL> Rem
SQL> Rem Copyright (c) 2012, Oracle and/or its affiliates. All rights reserved.
SQL> Rem
SQL> Rem      NAME
SQL> Rem      sqlsessend.sql - SQL session end
SQL> Rem
SQL> Rem      DESCRIPTION
SQL> Rem      Any commands which should be run at the end of all oracle
SQL> Rem      supplied scripts.
SQL> Rem
SQL> Rem      NOTES
SQL> Rem      See sqlsessstart.sql for the corresponding start script.
SQL> Rem
SQL> Rem      MODIFIED   <MM/DD/YY>
SQL> Rem      surman      11/14/12 - 12876907: Common start and end scripts
SQL> Rem      surman      11/14/12 - Created
SQL> Rem
SQL> alter session set "_ORACLE_SCRIPT" = false;
Sesión modificada.

SQL>
SQL> Rem *****
SQL> Rem END catproc.sql
SQL> Rem *****
SQL> _
```

## Alter database open (Ya estaba la instancia de la Base de Datos).

```
SQL> alter database open;
alter database open
**
ERROR en línea 1:
ORA-01531: ya hay una base de datos abierta por la instancia
SQL>
```

## Modelo



## 3. Implementar el modelo de la base de datos

### Implementar las TableSpaces

```
SQL> create tablespace ventas datafile 'C:\BD1\datos\ventas\ventas.dbf' size 3M
reuse autoextend on next 5M maxsize 150M;

Tablespace creado.

SQL> create tablespace contabilidad datafile 'C:\BD1\datos\contabilidad\contabil
idad.dbf' size 3M reuse autoextend on next 5M maxsize 150M;

Tablespace creado.

SQL> _

SQL> create tablespace produccion datafile 'C:\BD1\datos\produccion\produccion.d
bf' size 3M reuse autoextend on next 5M maxsize 150M;

Tablespace creado.
```

## Implementar Tablas

```
CREATE TABLE AreaVentas(  
    codigo number(4),  
    nombre varchar(30),  
    CONSTRAINT PKAREAVENTAS 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 AreaVentas(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 ventas;
```

```
CREATE TABLE Cobros(
```

```
    recibo number(4),
```

```
    cuenta number(4),
```

```
    fecha date,
```

```
    montop number(10,2),
```

```
    CONSTRAINT PKCOBROS PRIMARY KEY(recibo),
```

```
    CONSTRAINT FKCOBROS FOREIGN KEY(cuenta) REFERENCES CuentaXCobrar(cuenta)
```

```
)TABLESPACE ventas;
```

```
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 PPKFACTURA_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 FKCOBRO_FACTURA FOREIGN KEY(recibo) REFERENCES Cobros(recibo),  
  
    CONSTRAINT FK1COBRO_FACTURA FOREIGN KEY(factura) REFERENCES Factura_Venta(numero)  
  
)TABLESPACE ventas;
```

```
CREATE TABLE Zona(  
  
    codigo int,  
  
    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 FKLINEAS_FACTURA FOREIGN KEY(factura) REFERENCES Factura_Venta(numero),  
  
CONSTRAINT FK1LINEAS_FACTURA FOREIGN KEY(articulo) REFERENCES Productos(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 PKINVENTARIO_ARTICULOS PRIMARY KEY(articulo, bodega),  
  
CONSTRAINT FKINVENTARIO_ARTICULOS FOREIGN KEY(articulo) REFERENCES Productos(codigo),  
  
CONSTRAINT FK1INVENTARIO_ARTICULOS FOREIGN KEY(bodega) REFERENCES Bodega(codigo)  
  
)TABLESPACE produccion;
```

```
CREATE TABLE Proveedor(  
  
    codigo number(4),  
  
    nombre varchar(30),  
  
    telefono varchar(20),  
  
    CONSTRAINT PKPROOVEDOR PRIMARY KEY(codigo)  
  
)TABLESPACE produccion;
```

```
CREATE TABLE Componente(  
  
    codigo number(4),  
  
    nombre varchar(30),  
  
    precio number(10,2),  
  
    CONSTRAINT PKCOMPONENTE PRIMARY KEY(codigo),  
  
    CONSTRAINT FKCOMPONENTE FOREIGN KEY(codigo) REFERENCES Proveedor(codigo)  
  
)TABLESPACE produccion;
```

```
CREATE TABLE Formula(  
  
    componente number(4),  
  
    cantidad number(10,2),  
  
    producto number(4),  
  
    CONSTRAINT PKFORMULA PRIMARY KEY(componente, producto),  
  
    CONSTRAINT FKFORMULA FOREIGN KEY(componente) REFERENCES Componente(codigo),  
  
    CONSTRAINT FK1FORMULA FOREIGN KEY(producto) REFERENCES Productos(codigo)  
  
)TABLESPACE produccion;
```

```
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 FKINVENTARIO_COMPONENTES FOREIGN KEY(bodega) REFERENCES Bodega(codigo),

CONSTRAINT FK1INVENTARIO_COMPONENTES FOREIGN KEY(componente) REFERENCES Componente(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 Pagos(

    recibo number(4),

    cuenta number(4),

    fecha date,

    monto number(10,2),

    CONSTRAINT PKPAGOS PRIMARY KEY(recibo),

    CONSTRAINT FKPAGOS FOREIGN KEY(cuenta) REFERENCES CuentaXPagar(cuenta)

)TABLESPACE contabilidad;

```

```

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 FKFACTURA_COMPRA FOREIGN KEY(proveedor) REFERENCES Proveedor(codigo)

)TABLESPACE contabilidad;

```

```

CREATE TABLE Pago_Factura(

recibo number(4),

factura number(4),

monto number(10,2),

CONSTRAINT PKPAGO_FACTURA PRIMARY KEY(recibo, factura),

CONSTRAINT FKPAGO_FACTURA FOREIGN KEY(recibo) REFERENCES Pagos(recibo),

CONSTRAINT FK1PAGO_FACTURA FOREIGN KEY(factura) REFERENCES Factura_Compra(numero)

)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 PKLINEAS_FACTURA_COMPRA PRIMARY KEY(factura,componente),

CONSTRAINT FK1LINEAS_FACTURA_COMPRA FOREIGN KEY(factura) REFERENCES Factura_Compra(numero),

```

CONSTRAINT FKLNEAS\_FACTURA\_COMPRA FOREIGN KEY(componente) REFERENCES Componente(codigo)

)TABLESPACE contabilidad;

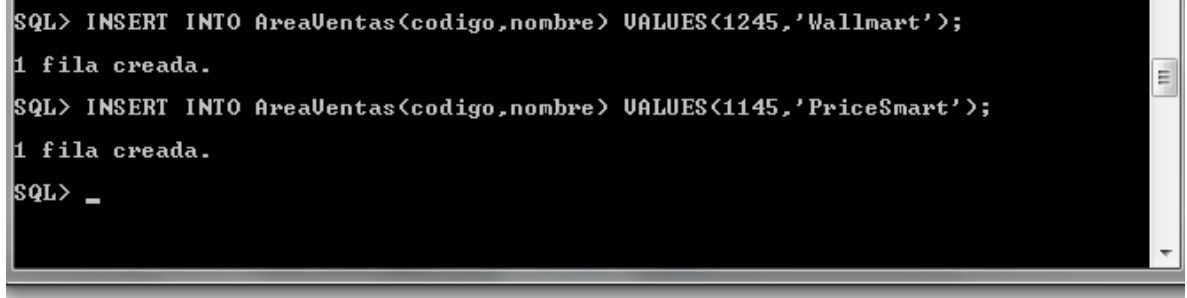


A screenshot of a SQL Plus window titled "SQL Plus". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The main area is a black terminal with white text. The text shows the execution of a script file: "SQL> @C:\BD1\BasesII.sql". Below this, there are eight lines of output, each stating "Tabla creada." (Table created.).

```
SQL> @C:\BD1\BasesII.sql
Tabla creada.
Tabla creada.
Tabla creada.
Tabla creada.
Tabla creada.
Tabla creada.
Tabla creada.
Tabla creada.
```

#### 4. Pruebas

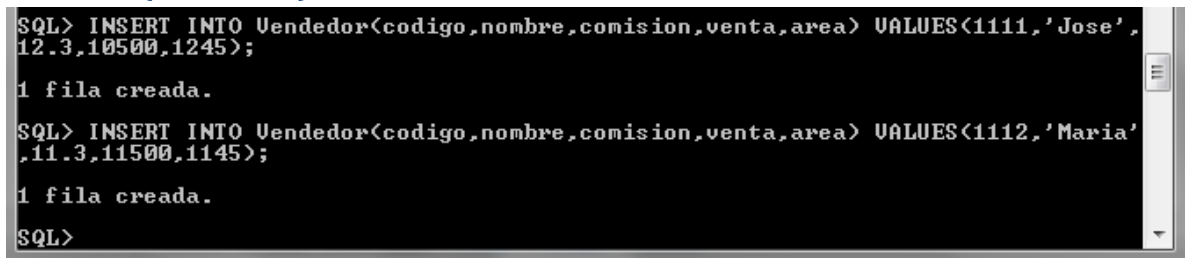
##### Prueba1 (Área Ventas)



A screenshot of a SQL Plus window showing two successful INSERT statements into the 'AreaVentas' table. The first statement inserts a record with code 1245 and name 'Wallmart'. The second statement inserts a record with code 1145 and name 'PriceSmart'. Each insertion is followed by the output "1 fila creada." (1 row created.). The prompt "SQL> \_" is visible at the bottom.

```
SQL> INSERT INTO AreaVentas(codigo,nombre) VALUES(1245,'Wallmart');
1 fila creada.
SQL> INSERT INTO AreaVentas(codigo,nombre) VALUES(1145,'PriceSmart');
1 fila creada.
SQL> _
```

##### Prueba2 (Vendedor)



A screenshot of a SQL Plus window showing two successful INSERT statements into the 'Vendedor' table. The first statement inserts a record with code 1111, name 'Jose', commission 12.3, sales 10500, and area 1245. The second statement inserts a record with code 1112, name 'Maria', commission 11.3, sales 11500, and area 1145. Each insertion is followed by the output "1 fila creada." (1 row created.). The prompt "SQL>" is visible at the bottom.

```
SQL> INSERT INTO Vendedor(codigo,nombre,comision,venta,area) VALUES(1111,'Jose',
12.3,10500,1245);
1 fila creada.
SQL> INSERT INTO Vendedor(codigo,nombre,comision,venta,area) VALUES(1112,'Maria'
,11.3,11500,1145);
1 fila creada.
SQL>
```

### Prueba3 (Zona)

```
SQL>
SQL> INSERT INTO Zona(codigo,descripcion) VALUES(1235,'Vieja');
1 fila creada.
SQL> INSERT INTO Zona(codigo,descripcion) VALUES(5431,'Grande');
1 fila creada.
SQL> _
```

### Prueba4 (Bodega)

```
SQL>
SQL> INSERT INTO Bodega(codigo,zona) VALUES(5555,5431);
1 fila creada.
SQL> INSERT INTO Bodega(codigo,zona) VALUES(4444,1235);
1 fila creada.
SQL> _
```

### Prueba5(Proveedor)

```
SQL>
SQL> INSERT INTO Proveedor(codigo,nombre,telefono) VALUES(1111,'Marchisio',84751
245);
1 fila creada.
SQL> INSERT INTO Proveedor(codigo,nombre,telefono) VALUES(7777,'Cerci',86445875)
;
1 fila creada.
SQL>
```

## 5. Muestras

### Muestra 1

```
SQL> INSERT INTO Proveedor(codigo,nombre,telefono) VALUES(7777,'Cerci',86445875)
;
1 fila creada.
SQL> select * from proveedor;

```

CODIGO	NOMBRE	TELEFONO
1111	Marchisio	84751245
7777	Cerci	86445875

```
SQL> _
```

## Muestra2

```
SQL> select * from zona;
```

CODIGO	DESCRIPCION
--------	-------------

1235	Uieja
5431	Grande

```
SQL>
```

## Muestra3

```
SQL> select * from vendedor;
```

CODIGO	NOMBRE	COMISION	VENTA	AREA
--------	--------	----------	-------	------

1111	Jose	12,3	10500	1245
1112	Maria	11,3	11500	1145

```
SQL>
```