"**Año** de la universalización de la salud"

Integrantes:

Papuico Miranda Alexander Antony

Cornejo Rojas Abel Arturo

Docente: LEVANO RODRIGUEZ DANIEL

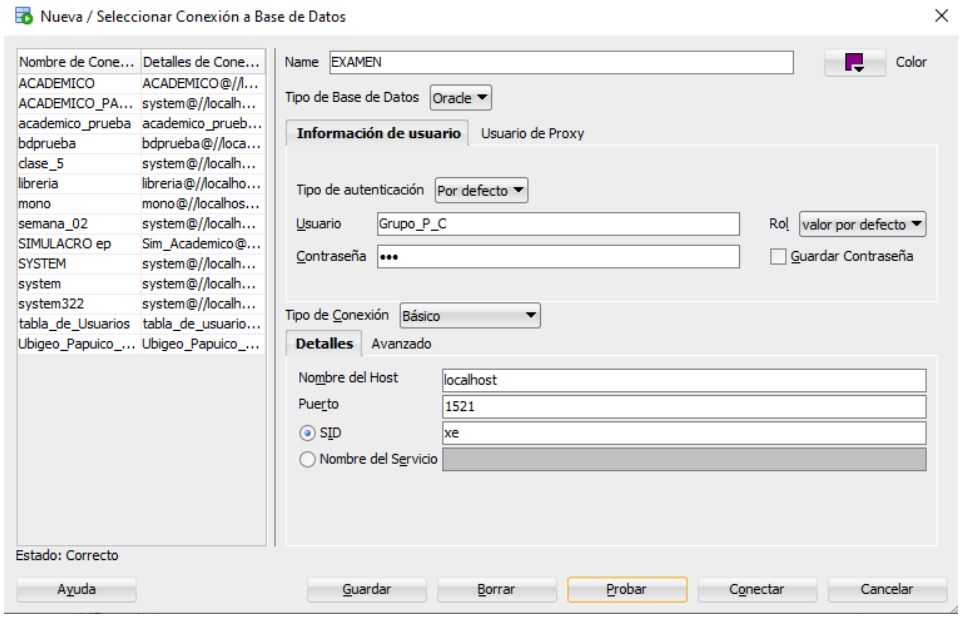
Curso: Diseño de Base de Datos

IS04S1

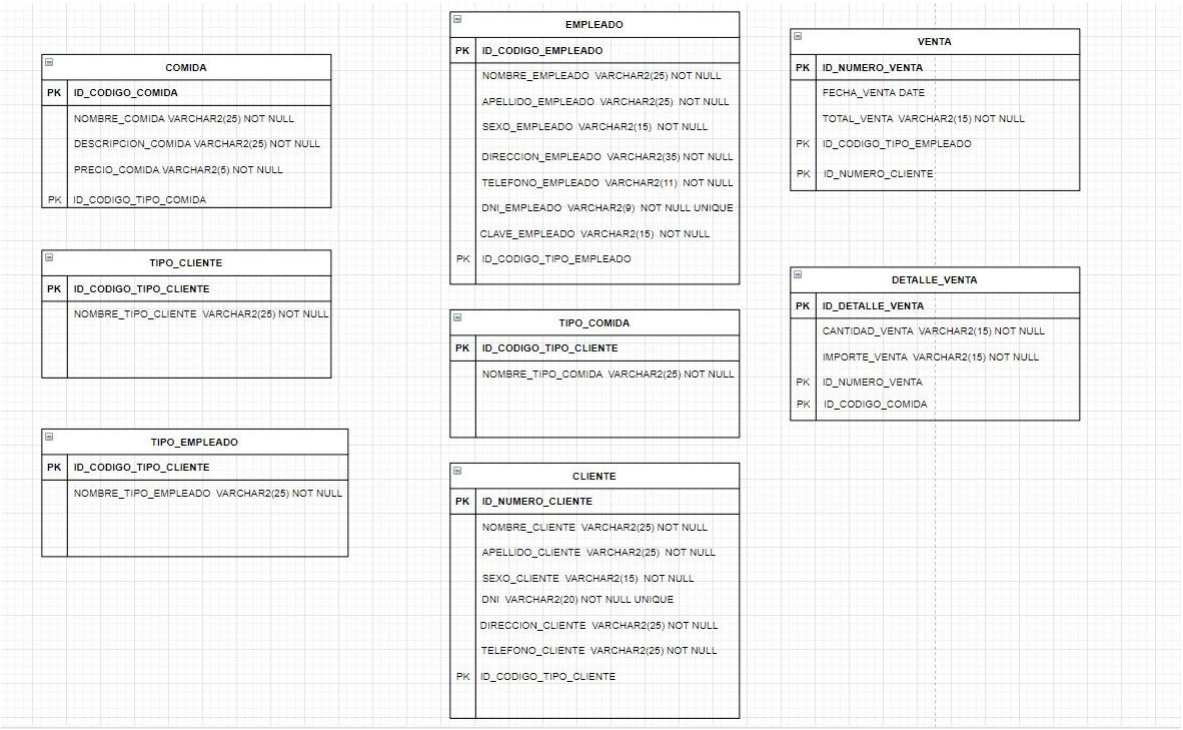
2020

1. Crear un usuario llamado **“Grupo_inicialesdesusapellidos”, clave: 123 (1 ptos)**

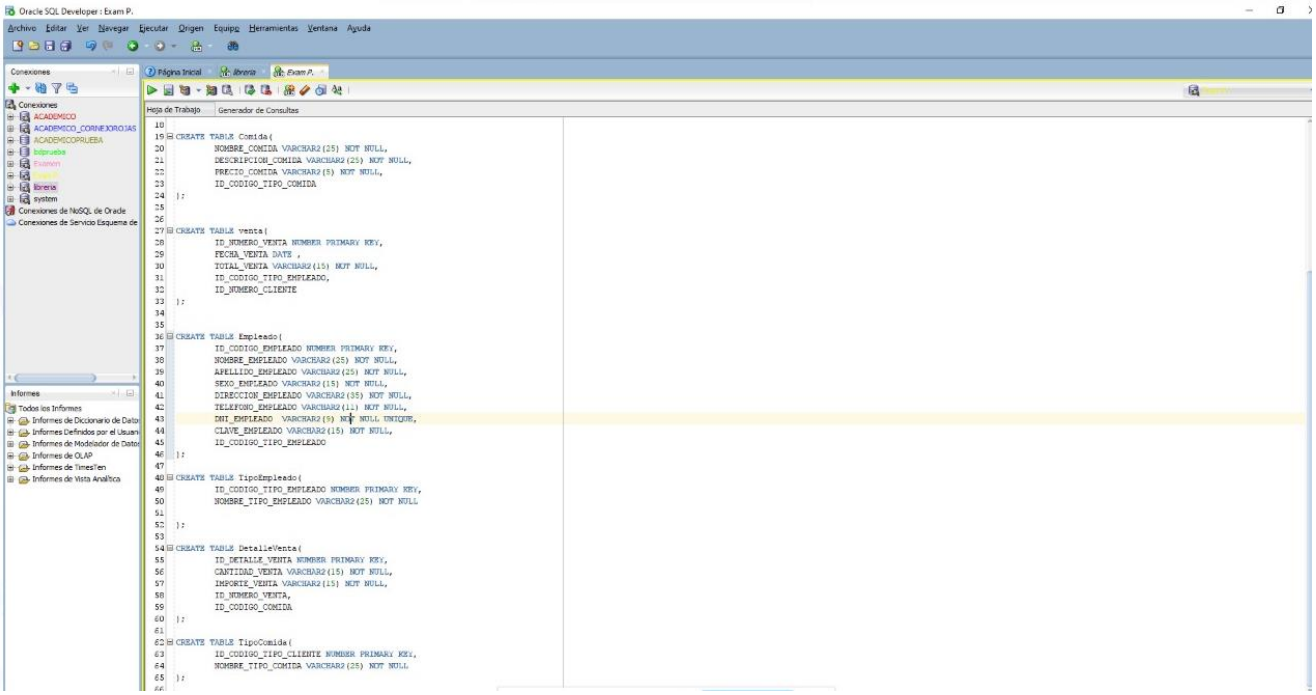
```
CREATE USER Grupo_P_C
IDENTIFIED BY "123"
DEFAULT TABLESPACE "USERS"
TEMPORARY TABLESPACE "TEMP";
GRANT ALL PRIVILEGES TO Grupo_P_C;
```



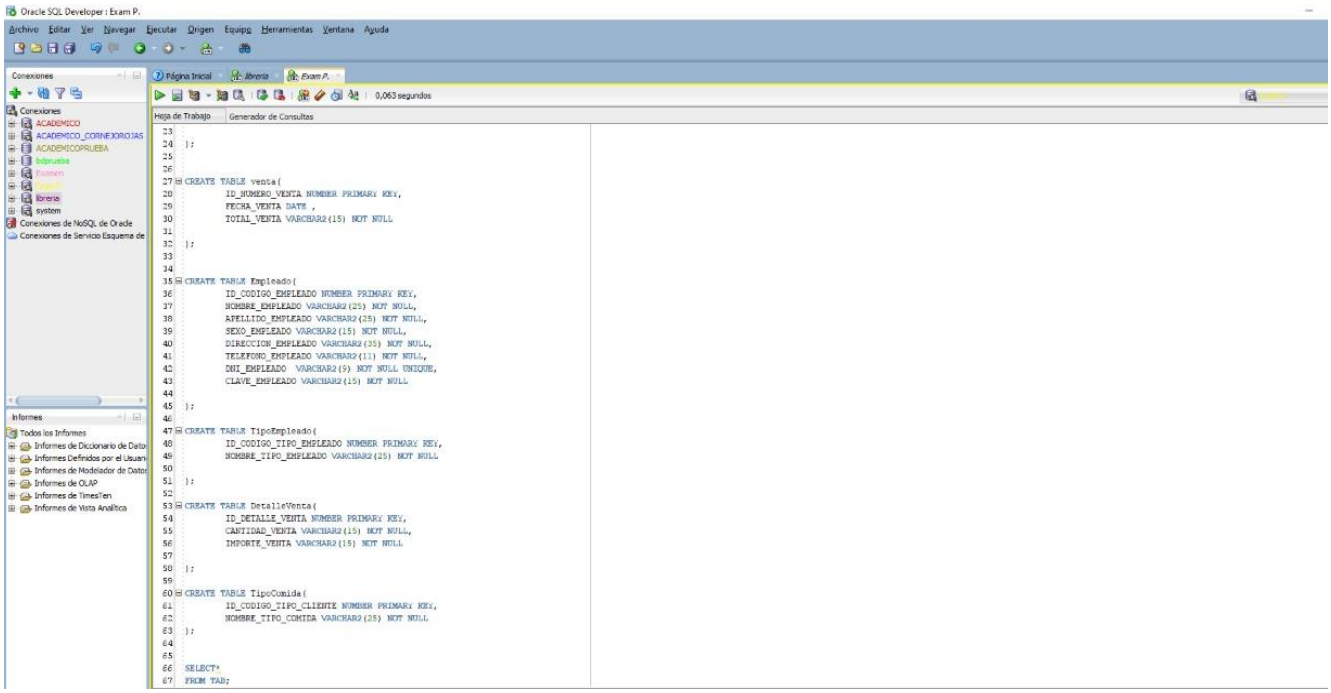
1. Rediseñe las tablas utilizando Draw.io.



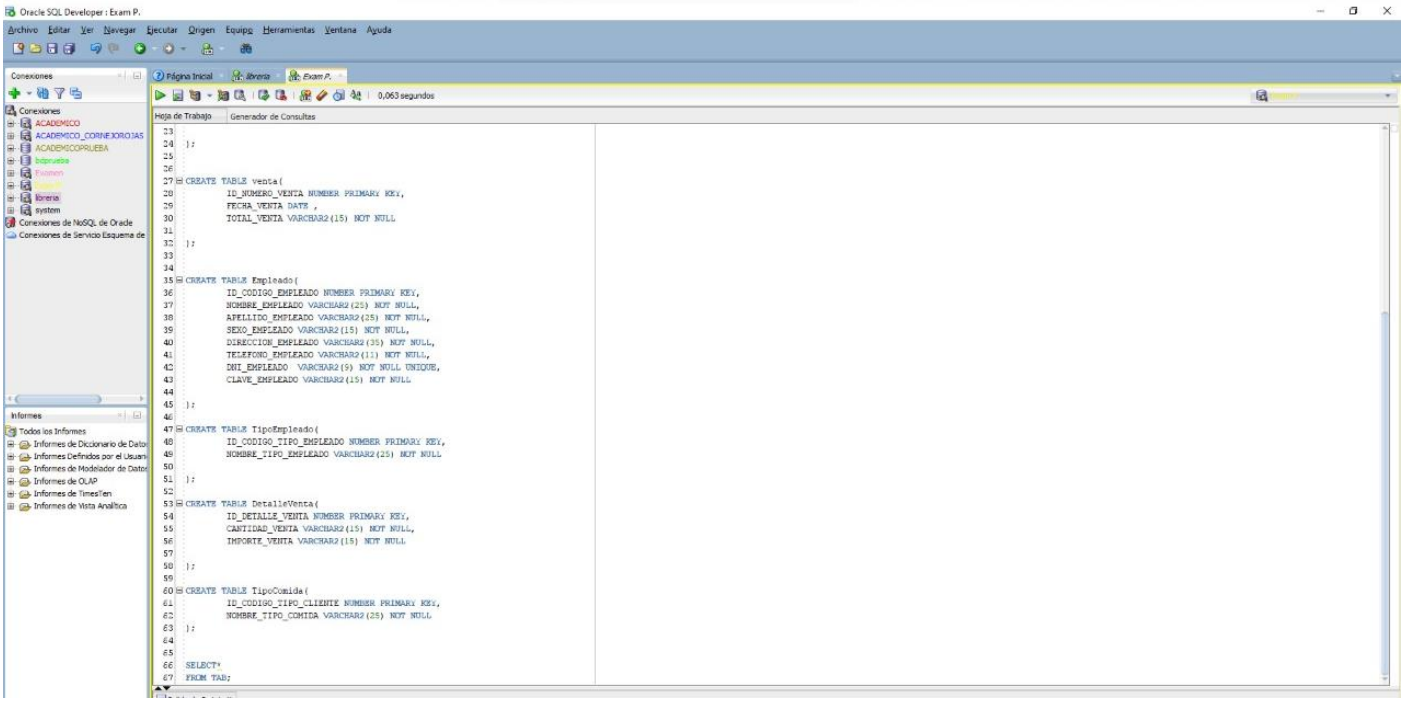
2. Crear las tablas de la Figura 1**. (5 ptos)**

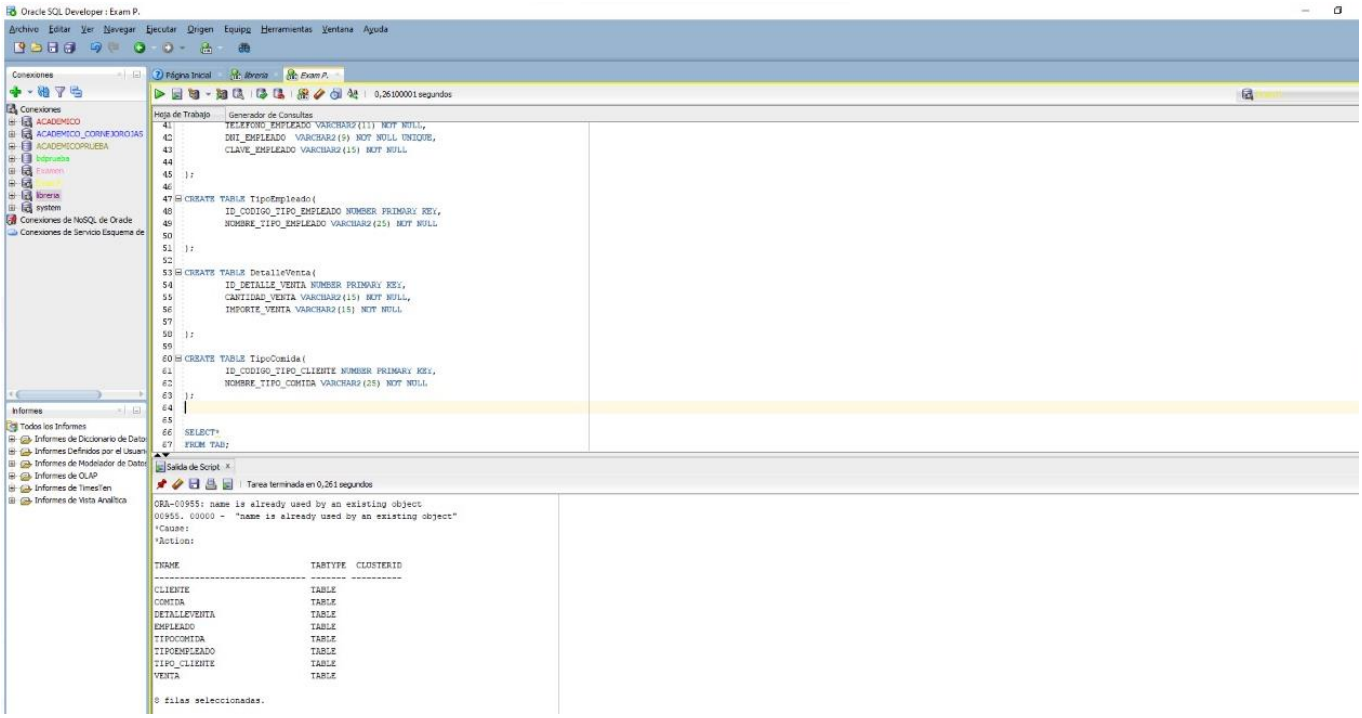


```
CREATE TABLE Comida(
        NOMBRE_COMIDA VARCHAR2(25) NOT NULL,
        DESCRIPCION_COMIDA VARCHAR2(25) NOT NULL,
        PRECIO_COMIDA VARCHAR2(5) NOT NULL,
        ID_CODIGO_TIPO_COMIDA
);

CREATE TABLE venta(
        ID_NUMERO_VENTA NUMBER PRIMARY KEY,
        FECHA_VENTA DATE ,
        TOTAL_VENTA VARCHAR2(15) NOT NULL,
        ID_CODIGO_TIPO_EMPLEADO,
        ID_NUMERO_CLIENTE
);

CREATE TABLE Empleado(
        ID_CODIGO_EMPLEADO NUMBER PRIMARY KEY,
        NOMBRE_EMPLEADO VARCHAR2(25) NOT NULL,
        APELLIDO_EMPLEADO VARCHAR2(25) NOT NULL,
        SEXO_EMPLEADO VARCHAR2(15) NOT NULL,
        DIRECCION_EMPLEADO VARCHAR2(35) NOT NULL,
        TELEFONO_EMPLEADO VARCHAR2(11) NOT NULL,
        DNI_EMPLEADO  VARCHAR2(9) NOT NULL UNIQUE,
        CLAVE_EMPLEADO VARCHAR2(15) NOT NULL,
        ID_CODIGO_TIPO_EMPLEADO
);

CREATE TABLE TipoEmpleado(
        ID_CODIGO_TIPO_EMPLEADO NUMBER PRIMARY KEY,
        NOMBRE_TIPO_EMPLEADO VARCHAR2(25) NOT NULL

);

CREATE TABLE DetalleVenta(
        ID_DETALLE_VENTA NUMBER PRIMARY KEY,
        CANTIDAD_VENTA VARCHAR2(15) NOT NULL,
        IMPORTE_VENTA VARCHAR2(15) NOT NULL,
        ID_NUMERO_VENTA,
        ID_CODIGO_COMIDA
);

CREATE TABLE TipoComida(
        ID_CODIGO_TIPO_CLIENTE NUMBER PRIMARY KEY,
        NOMBRE_TIPO_COMIDA VARCHAR2(25) NOT NULL
);
```



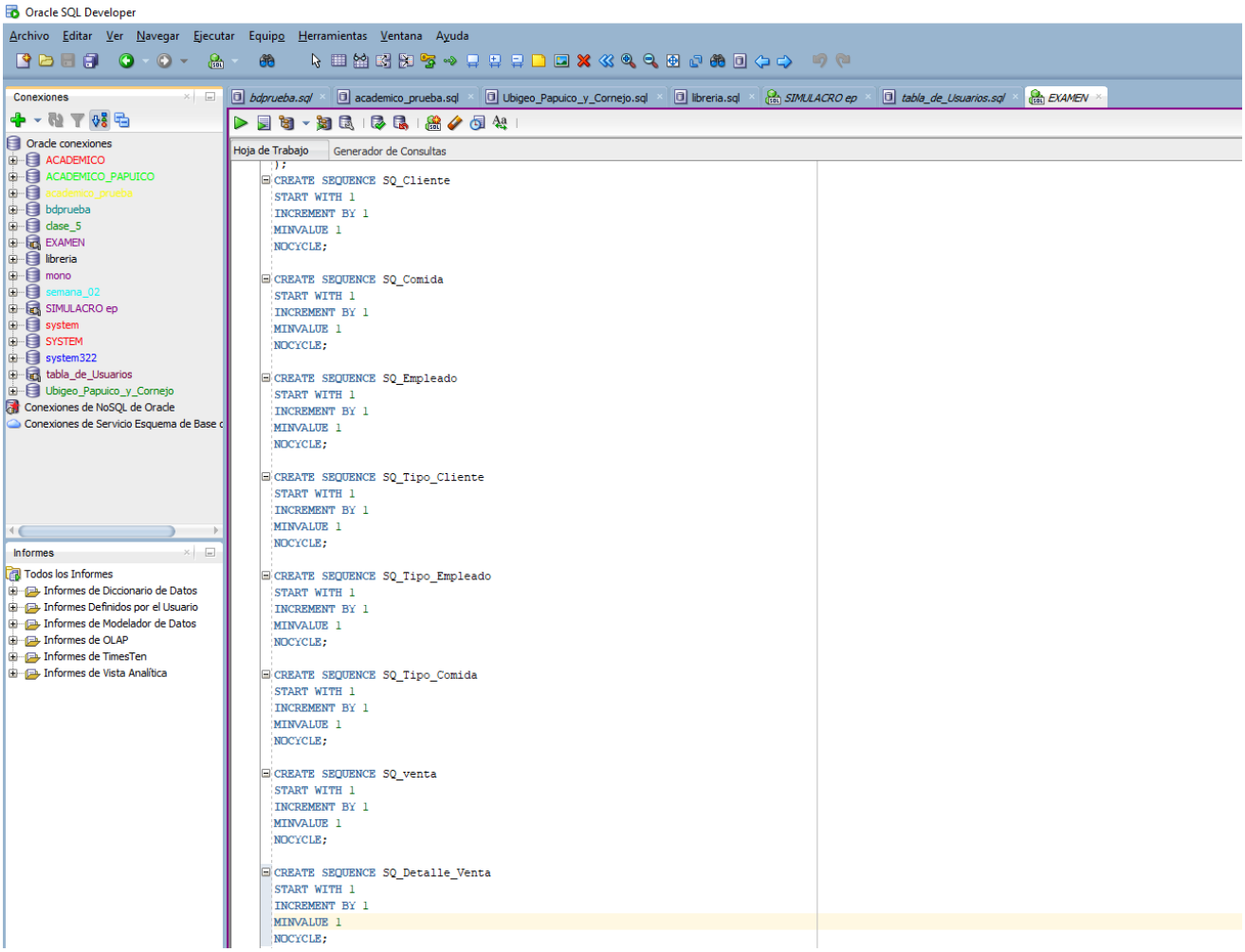
```
);

CREATE TABLE venta(
        ID_NUMERO_VENTA NUMBER PRIMARY KEY,
        FECHA_VENTA DATE ,
        TOTAL_VENTA VARCHAR2(15) NOT NULL

);

CREATE TABLE Empleado(
        ID_CODIGO_EMPLEADO NUMBER PRIMARY KEY,
        NOMBRE_EMPLEADO VARCHAR2(25) NOT NULL,
        APELLIDO_EMPLEADO VARCHAR2(25) NOT NULL,
        SEXO_EMPLEADO VARCHAR2(15) NOT NULL,
        DIRECCION_EMPLEADO VARCHAR2(35) NOT NULL,
        TELEFONO_EMPLEADO VARCHAR2(11) NOT NULL,
        DNI_EMPLEADO  VARCHAR2(9) NOT NULL UNIQUE,
        CLAVE_EMPLEADO VARCHAR2(15) NOT NULL

);

CREATE TABLE TipoEmpleado(
        ID_CODIGO_TIPO_EMPLEADO NUMBER PRIMARY KEY,
        NOMBRE_TIPO_EMPLEADO VARCHAR2(25) NOT NULL

);

CREATE TABLE DetalleVenta(
        ID_DETALLE_VENTA NUMBER PRIMARY KEY,
        CANTIDAD_VENTA VARCHAR2(15) NOT NULL,
        IMPORTE_VENTA VARCHAR2(15) NOT NULL

);

CREATE TABLE TipoComida(
        ID_CODIGO_TIPO_CLIENTE NUMBER PRIMARY KEY,
        NOMBRE_TIPO_COMIDA VARCHAR2(25) NOT NULL
);


SELECT*
FROM TAB;
```

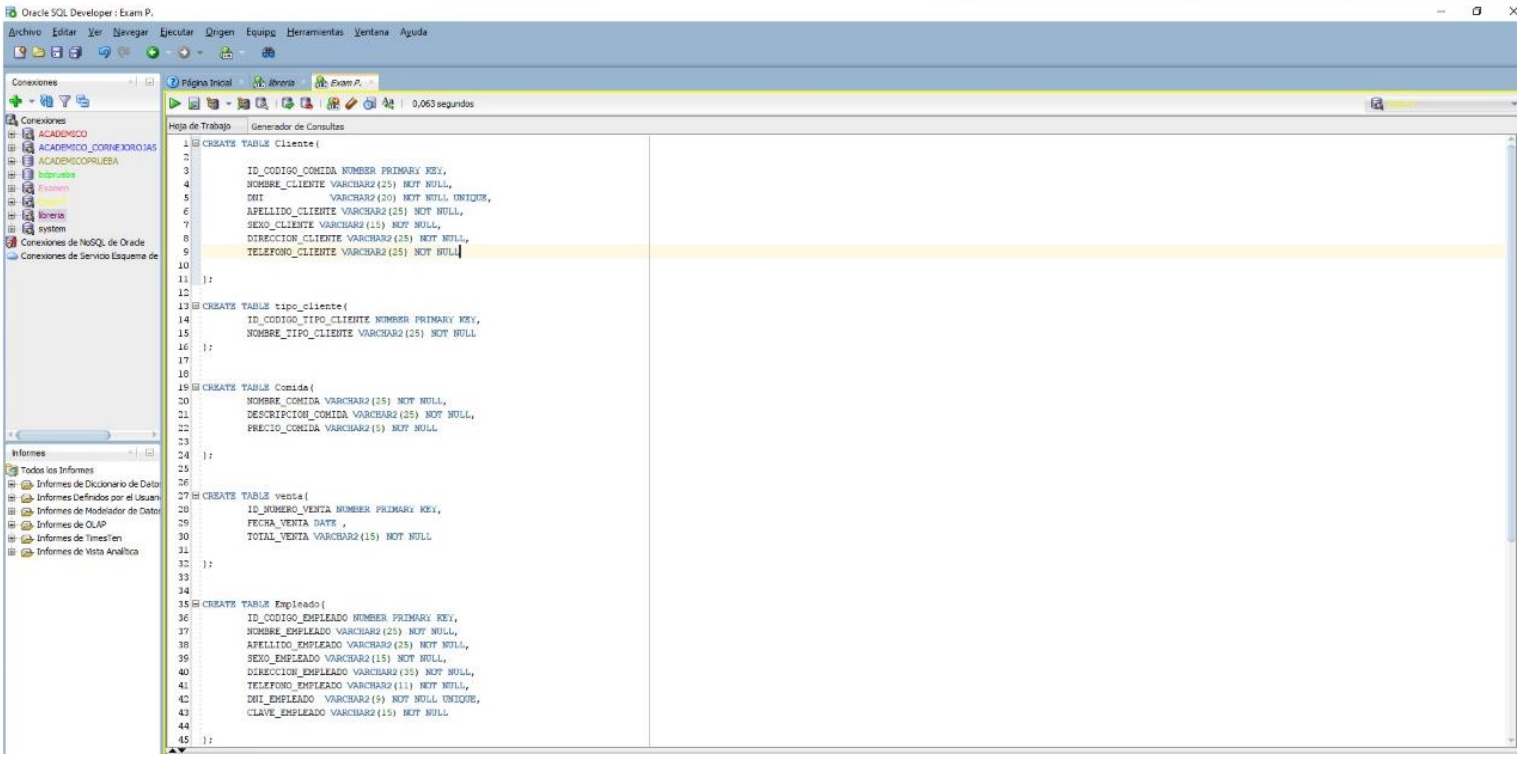

Oracle SQL Developer : Exam P.
Archivo Editar Ver Navegar Ejecutar Origen Equipo Herramientas Ventana Ayuda
Página Inicial
librería
Exam P.
0,063 segundos
Hoja de Trabajo
Generador de Consultas
);
CREATE TABLE Venta(
ID_NUMERO_VENTA NUMBER PRIMARY KEY,
FECHA_VENTA DATE ,
TOTAL_VENTA VARCHAR2(15) NOT NULL
);
CREATE TABLE Empleado(
ID_CODIGO_EMPLEADO NUMBER PRIMARY KEY,
NOMBRE_EMPLEADO VARCHAR2(25) NOT NULL,
APELLIDO_EMPLEADO VARCHAR2(25) NOT NULL,
SEXO_EMPLEADO VARCHAR2(15) NOT NULL,
DIRECCION_EMPLEADO VARCHAR2(35) NOT NULL,
TELEFONO_EMPLEADO VARCHAR2(11) NOT NULL,
DNI_EMPLEADO VARCHAR2(9) NOT NULL UNIQUE,
CLAVE_EMPLEADO VARCHAR2(15) NOT NULL
);
CREATE TABLE TipoEmpleado(
ID_CODIGO_TIPO_EMPLEADO NUMBER PRIMARY KEY,
NOMBRE_TIPO_EMPLEADO VARCHAR2(25) NOT NULL
);
CREATE TABLE DetalleVenta(
ID_DETALLE_VENTA NUMBER PRIMARY KEY,
CANTIDAD_VENTA VARCHAR2(15) NOT NULL,
IMPORTE_VENTA VARCHAR2(15) NOT NULL
);
CREATE TABLE TipoComida(
ID_CODIGO_TIPO_CLIENTE NUMBER PRIMARY KEY,
NOMBRE_TIPO_COMIDA VARCHAR2(25) NOT NULL
);
SELECT*
FROM TAB;


Oracle SQL Developer : Exam P.
Archivo Editar Ver Navegar Ejecutar Origen Equipo Herramientas Ventana Ayuda
Página Inicial
librería
Exam P.
0,26100001 segundos
Hoja de Trabajo
Generador de Consultas
TELEFONO_EMPLEADO VARCHAR2(11) NOT NULL,
DNI_EMPLEADO VARCHAR2(9) NOT NULL UNIQUE,
CLAVE_EMPLEADO VARCHAR2(15) NOT NULL
);
CREATE TABLE TipoEmpleado(
ID_CODIGO_TIPO_EMPLEADO NUMBER PRIMARY KEY,
NOMBRE_TIPO_EMPLEADO VARCHAR2(25) NOT NULL
);
CREATE TABLE DetalleVenta(
ID_DETALLE_VENTA NUMBER PRIMARY KEY,
CANTIDAD_VENTA VARCHAR2(15) NOT NULL,
IMPORTE_VENTA VARCHAR2(15) NOT NULL
);
CREATE TABLE TipoComida(
ID_CODIGO_TIPO_CLIENTE NUMBER PRIMARY KEY,
NOMBRE_TIPO_COMIDA VARCHAR2(25) NOT NULL
);
SELECT*
FROM TAB;
Salida de Script
Tarea terminada en 0,261 segundos
ORA-00955: name is already used by an existing object
00955. 00000 - "name is already used by an existing object"
*Cause:
*Action:
TNAME TABTYPE CLUSTERID
CLIENTE TABLE
COMIDA TABLE
DETALLEVENTA TABLE
EMPLEADO TABLE
TIPOCOMIDA TABLE
TIPOEMPLEADO TABLE
TIPO_CLIENTE TABLE
VENTA TABLE
8 filas seleccionadas.

4 . Insertar registros de tal manera que se puedan generar los reportes. Considerar lo siguiente:

SECUENCIA: Generación de ID Automático (Adjuntar ese código) **(2 ptos)**

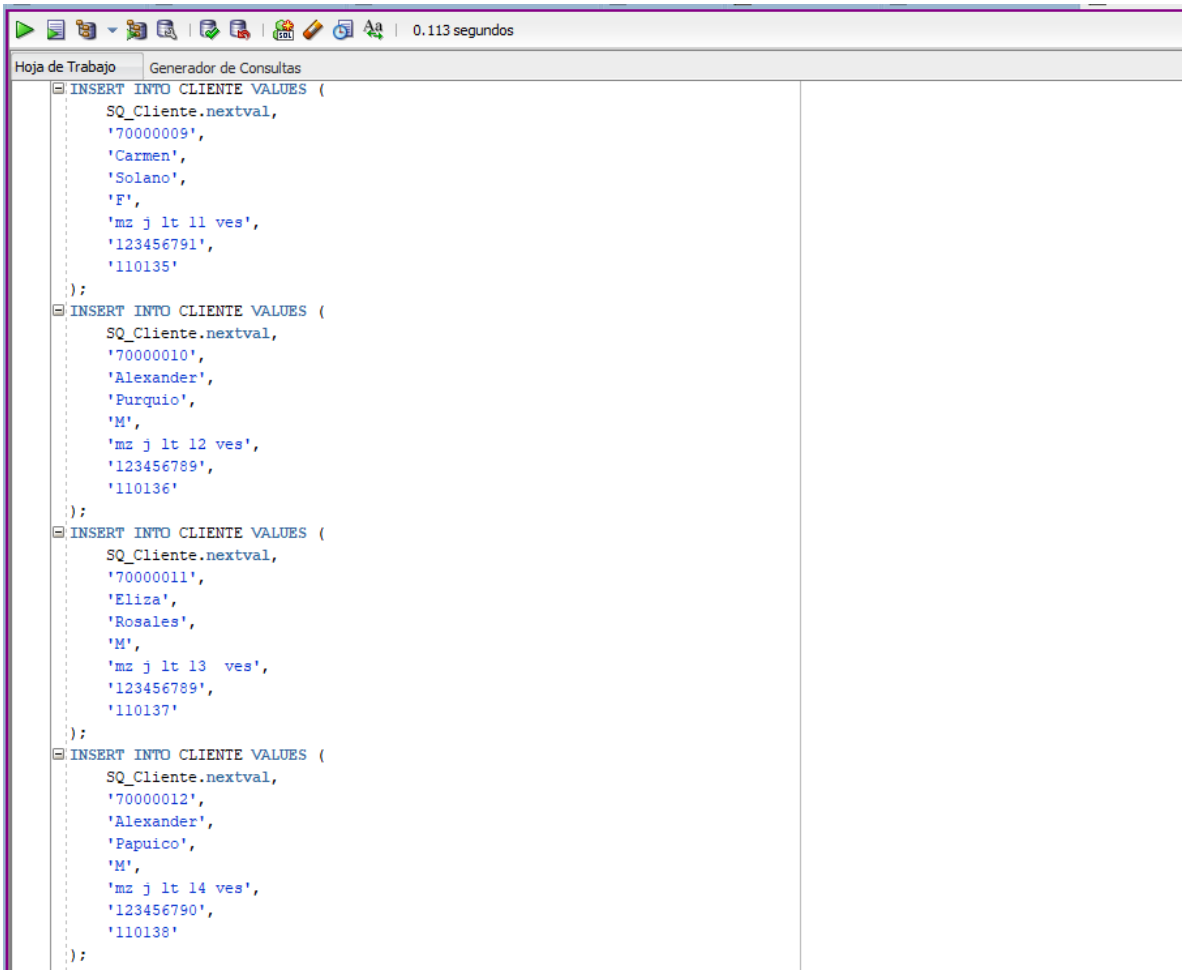


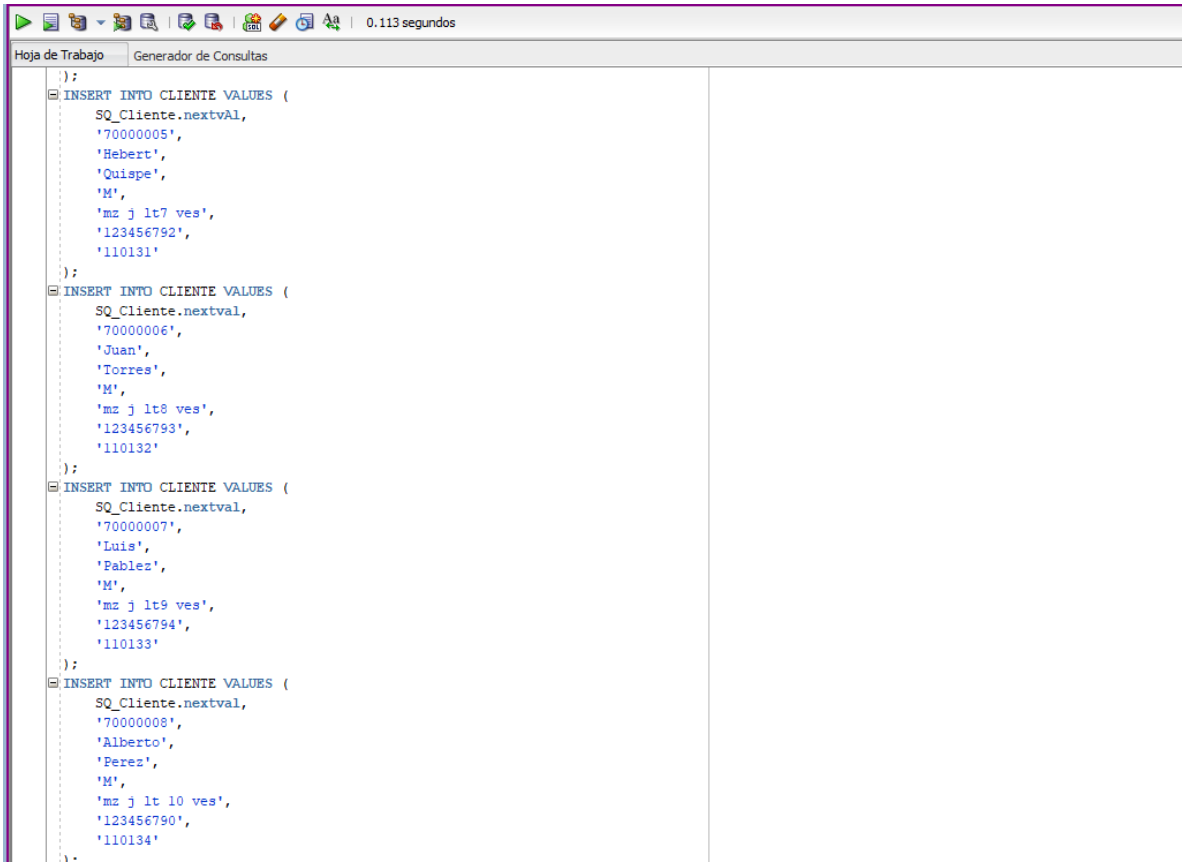



```
CREATE TABLE Cliente(
        ID_CODIGO_COMIDA NUMBER PRIMARY KEY,
        NOMBRE_CLIENTE VARCHAR2(25) NOT NULL,
        DNI            VARCHAR2(20) NOT NULL UNIQUE,
        APELLIDO_CLIENTE VARCHAR2(25) NOT NULL,
        SEXO_CLIENTE VARCHAR2(15) NOT NULL,
        DIRECCION_CLIENTE VARCHAR2(25) NOT NULL,
        TELEFONO_CLIENTE VARCHAR2(25) NOT NULL

);

CREATE TABLE tipo_cliente(
        ID_CODIGO_TIPO_CLIENTE NUMBER PRIMARY KEY,
        NOMBRE_TIPO_CLIENTE VARCHAR2(25) NOT NULL
);


CREATE TABLE Comida(
        NOMBRE_COMIDA VARCHAR2(25) NOT NULL,
        DESCRIPCION_COMIDA VARCHAR2(25) NOT NULL,
        PRECIO_COMIDA VARCHAR2(5) NOT NULL

);


CREATE TABLE venta(
        ID_NUMERO_VENTA NUMBER PRIMARY KEY,
        FECHA_VENTA DATE ,
        TOTAL_VENTA VARCHAR2(15) NOT NULL

);


CREATE TABLE Empleado(
        ID_CODIGO_EMPLEADO NUMBER PRIMARY KEY,
        NOMBRE_EMPLEADO VARCHAR2(25) NOT NULL,
        APELLIDO_EMPLEADO VARCHAR2(25) NOT NULL,
        SEXO_EMPLEADO VARCHAR2(15) NOT NULL,
        DIRECCION_EMPLEADO VARCHAR2(35) NOT NULL,
        TELEFONO_EMPLEADO VARCHAR2(11) NOT NULL,
        DNI_EMPLEADO  VARCHAR2(9) NOT NULL UNIQUE,
        CLAVE_EMPLEADO VARCHAR2(15) NOT NULL

);
```

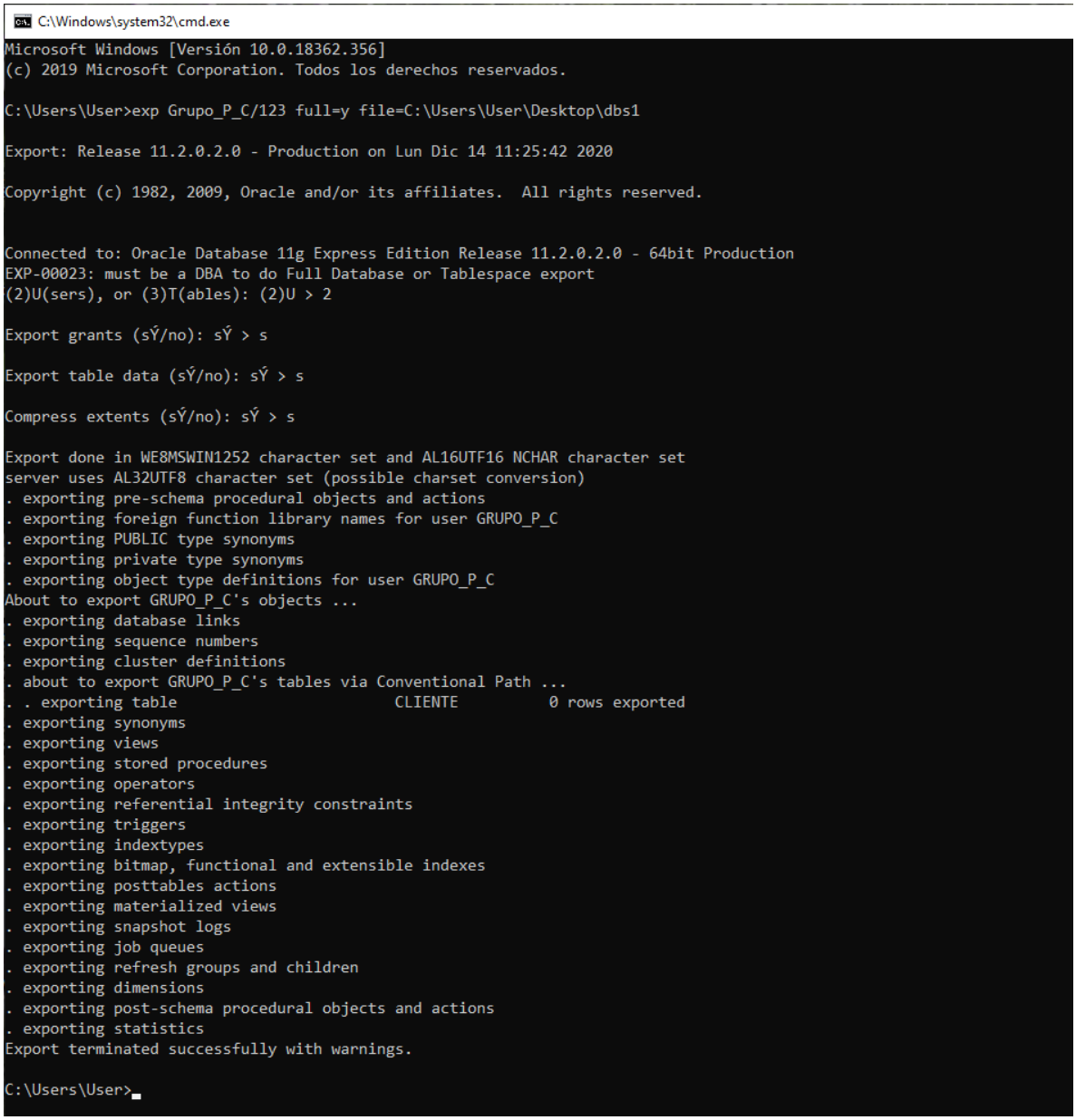
- INSERTAR 15 registros por cada integrante de equipo distribuido estratégicamente en las tablas. **<u>Considere para cada venta por lo menos 3 detalles.</u> (4 ptos)**

```
INSERT INTO CLIENTE VALUES (
    SQ_cliente.nextval,
    '70000001',
    'Alexander',
    'Papuico',
    'M',
    'mz j lt2 ves',
    '123456789',
    '110129'
);
INSERT INTO CLIENTE VALUES (
    SQ_Cliente.nextval,
    '70000002',
    'Alexanderio',
    'Fernandez',
    'M',
    'mz j lt3 ves',
    '123456781',
    '110129'
);
INSERT INTO CLIENTE VALUES (
    SQ_Cliente.nextval,
    '70000003',
    'Alexis',
    'Pañipilco',
    'M',
    'mz j lt5 ves',
    '123456790',
    '110129'
);
INSERT INTO CLIENTE VALUES (
    SQ_Cliente.nextval,
    '70000004',
    'Erick',
    'Miranda',
    'M',
    'mz j lt6 ves',
    '123456791',
    '110130'
);
```


0.113 segundos
Hoja de Trabajo
Generador de Consultas
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000009',
'Carmen',
'Solano',
'F',
'mz j lt 11 ves',
'123456791',
'110135'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000010',
'Alexander',
'Purquio',
'M',
'mz j lt 12 ves',
'123456789',
'110136'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000011',
'Eliza',
'Rosales',
'M',
'mz j lt 13 ves',
'123456789',
'110137'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000012',
'Alexander',
'Papuico',
'M',
'mz j lt 14 ves',
'123456790',
'110138'
);


0.113 segundos
Hoja de Trabajo
Generador de Consultas
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextvAl,
'70000005',
'Hebert',
'Quispe',
'M',
'mz j lt7 ves',
'123456792',
'110131'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000006',
'Juan',
'Torres',
'M',
'mz j lt8 ves',
'123456793',
'110132'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000007',
'Luis',
'Pablez',
'M',
'mz j lt9 ves',
'123456794',
'110133'
);
INSERT INTO CLIENTE VALUES (
SQ_Cliente.nextval,
'70000008',
'Alberto',
'Perez',
'M',
'mz j lt 10 ves',
'123456790',
'110134'
);

7. Exporte su base de datos **(2 ptos)**



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.18362.356]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\User>exp Grupo_P_C/123 full=y file=C:\Users\User\Desktop\dbs1

Export: Release 11.2.0.2.0 - Production on Lun Dic 14 11:25:42 2020

Copyright (c) 1982, 2009, Oracle and/or its affiliates.  All rights reserved.


Connected to: Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
EXP-00023: must be a DBA to do Full Database or Tablespace export
(2)U(sers), or (3)T(ables): (2)U > 2

Export grants (sÝ/no): sÝ > s

Export table data (sÝ/no): sÝ > s

Compress extents (sÝ/no): sÝ > s

Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set
server uses AL32UTF8 character set (possible charset conversion)
. exporting pre-schema procedural objects and actions
. exporting foreign function library names for user GRUPO_P_C
. exporting PUBLIC type synonyms
. exporting private type synonyms
. exporting object type definitions for user GRUPO_P_C
About to export GRUPO_P_C's objects ...
. exporting database links
. exporting sequence numbers
. exporting cluster definitions
. about to export GRUPO_P_C's tables via Conventional Path ...
. . exporting table                        CLIENTE          0 rows exported
. exporting synonyms
. exporting views
. exporting stored procedures
. exporting operators
. exporting referential integrity constraints
. exporting triggers
. exporting indextypes
. exporting bitmap, functional and extensible indexes
. exporting posttables actions
. exporting materialized views
. exporting snapshot logs
. exporting job queues
. exporting refresh groups and children
. exporting dimensions
. exporting post-schema procedural objects and actions
. exporting statistics
Export terminated successfully with warnings.

C:\Users\User>
```

LINK PARA VER EL GITHUB

https://github.com/alexander451996/examen.parcial_dbs