

Tutoría colectiva DAM1 Bases de Datos



Modalidad: Distancia
21/12/2021 15:45

Contenidos



Aclaraciones UT01

Acceso web Oracle XE 18c

CDB y PDB desde acceso web

CDB y PDB desde SQLPlus

Plug/Unplug PDB

Crear PDB

Conexión a PDB desde SQL Developer

SQL Developer: ver Usuarios, Roles y Permisos

SQL Developer Visor de instancias

Aclaraciones UT02

Generar Diagrama desde SQL Developer

Generar DDL SQL Developer

Script Manual vs Script generado por SQL Developer

Equivalencias sintaxis DDL

Aclaraciones UT03

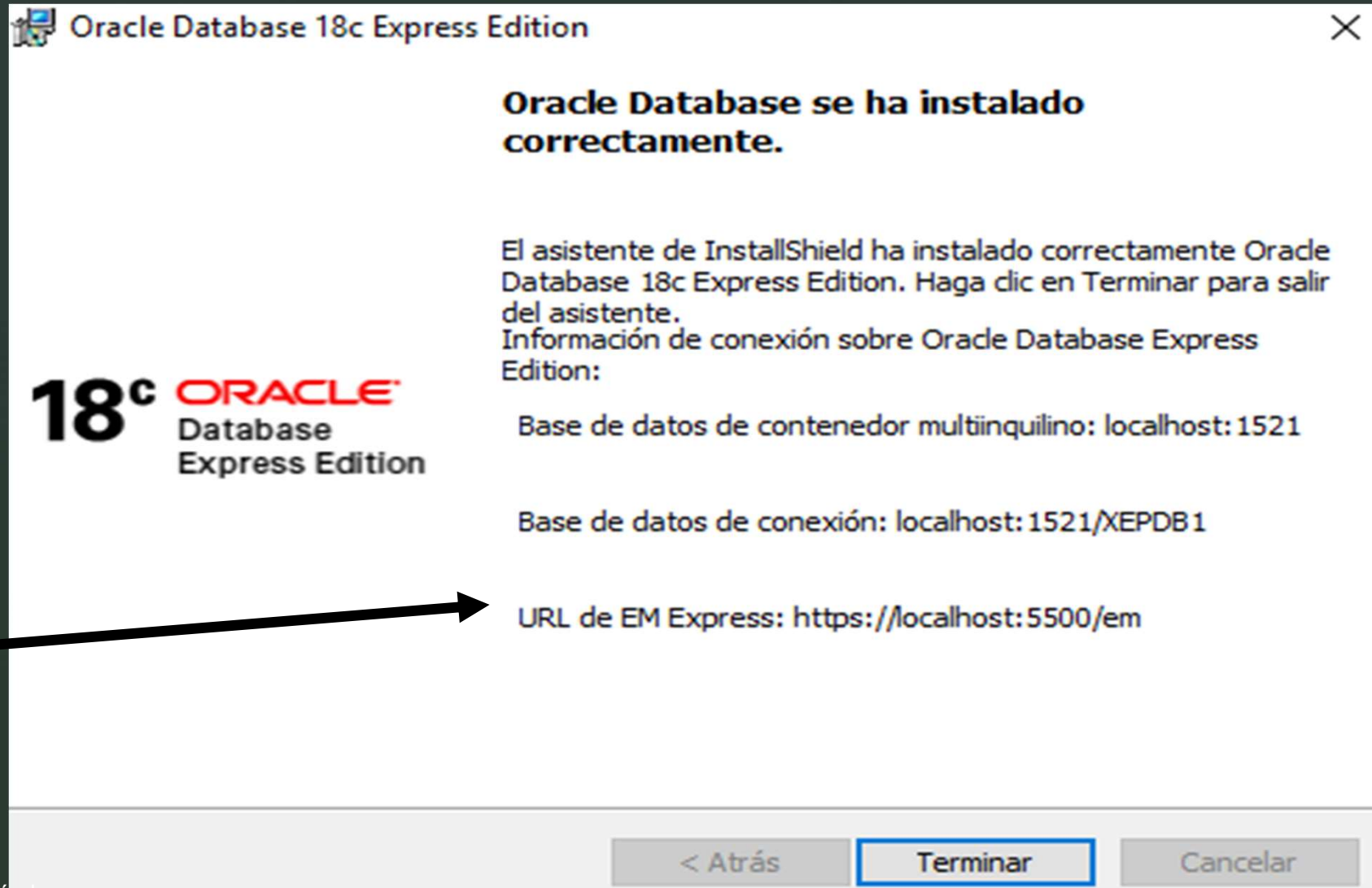
Transformación de relaciones ternarias

Formas normales

Ejemplo normalización.

ACLARACIONES UT01

De la instalación de Oracle DB XE 18c

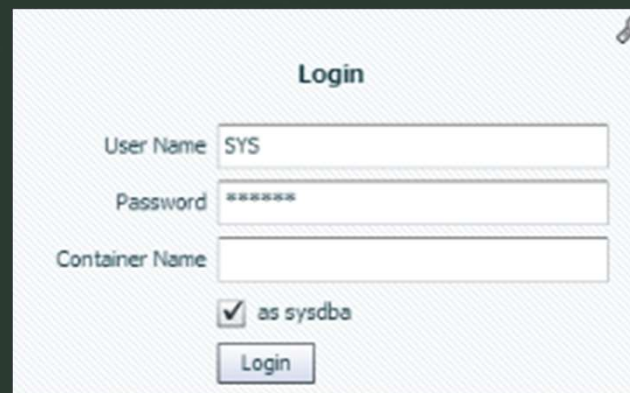
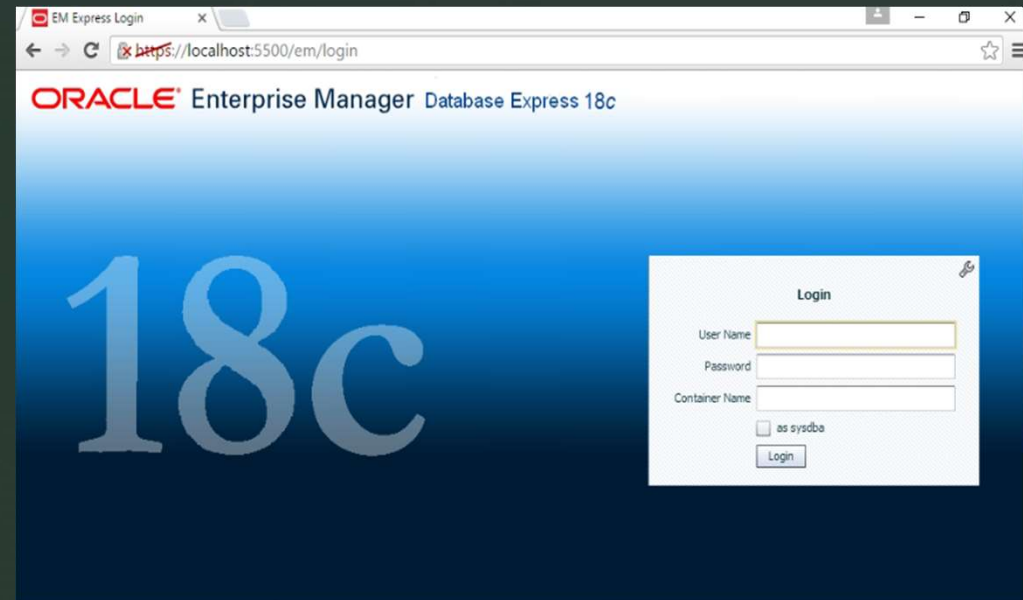
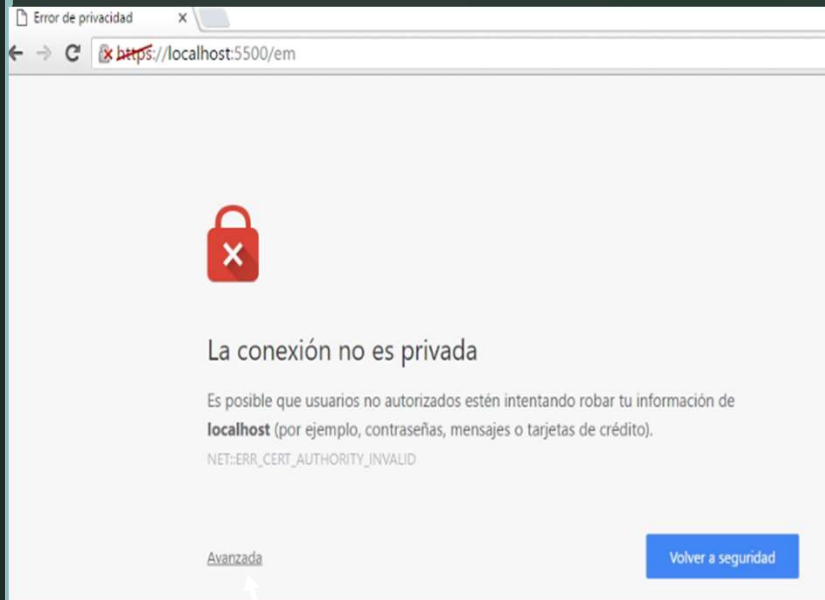


UT01 - Almacenamiento de información



- Oracle Database XE provee un acceso web con url: <https://localhost:5500/em>.
- Desde este acceso se puede ver la información general de la base de datos.
- Para versiones como la 18c de la tarea, este acceso requiere Flash, actualmente no habilitado en los navegadores. Para poder acceder, se puede usar un ChromePortable con Flash activado. [Archivo](#)
- Para ejecutarlo es necesario, descomprimir el archivo y luego ejecutar el archivo GoogleChromePortable.exe. Desde ahí ya se puede cargar el acceso web de Oracle Database XE con la URL. Una vez cargue, deberemos pulsar AVANZADA y posteriormente continuar a localhost.
- Como usuario usaremos SYS que tiene todos los permisos, pondremos su contraseña y hacemos clic en el checkbox AS SYSDBA como se muestra en la imagen siguiente.

Acceso Web Oracle DB XE

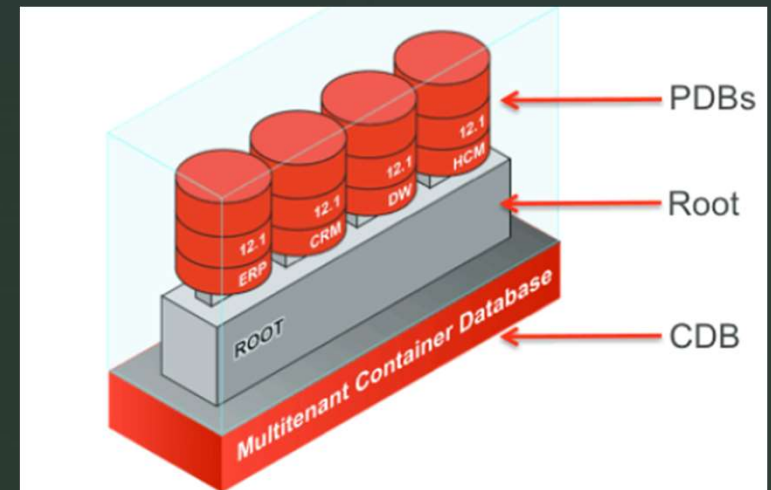


CDB y PDB desde acceso web

Status

Up Time 12 minutes
Type Single instance (xe)
CDB (2 PDBs)
Version 18.4.0.0.0 Express Edition
Database Name XE
Instance Name xe
Platform Name Microsoft Windows x86 64-bit
Host Name DESKTOP-RBRI671
Thread 1
Archiver Stopped

Up Time: tiempo desde último arranque
Type: Instancia única (xe)
CDB (2 PDBs)



<https://medium.com/@m.emmanuel/how-to-navigate-across-cdb-and-pdb-in-oracle-18c-xe-781ac217f5b0>

Oracle Enterprise Manager Database Express 18.4.0.0.0

Containers

Top Containers

Container Name	Type	Open...	Open...	Restr...	Size	Violat...	CPU Resource...	Running Sessions	Active Sessions	Memory Used	IO Requests
MASCO		↓			1GB		100%				
XEPDB1		↑	18 min...		1GB		100%			142MB	< 1/s

CDB y PDB desde SQLPlus

- Un único contenedor (CDB) puede alojar varias PDB (Bases de datos conectables independientes).
- Cuando accedemos con SYS desde SQL Plus, estamos por defecto en el contenedor CDB\$ROOT.
- Podemos ver los contenedores de tipo PDB con una sentencia similar a la de la imagen.

```
SQL> show user con_name  
USER es "SYS"
```

```
CON_NAME  
-----  
CDB$ROOT
```

```
SQL> SELECT NAME, CON_ID, CON_UID FROM V$PDBS ORDER BY CON_ID;
```

```
NAME
```

```
-----  
CON_ID  CON_UID  
-----  
PDB$SEED  
2 1145551651  
XEPDB1  
3 810694091  
MASCO  
4 645174625
```

Para cambiar el contenedor sobre el que trabajamos, podemos ejecutar la sentencia:
`ALTER SESSION SET CONTAINER = XEPDB1;`

```
SQL> ALTER SESSION SET CONTAINER=XEPDB1;  
Sesión modificada.
```

```
SQL> show user con_name  
USER es "SYS"
```

```
CON_NAME  
-----  
XEPDB1  
SQL>
```

Plug/Unplug PDB

Para enchufar/desenchufar una base de datos (PDB) usaremos los botones Plug/Unplug.



Desde sqlPlus también se puede conectar un PDB, como muestra la imagen.

```
SQL> ALTER SESSION SET CONTAINER=CDB$ROOT;
Sesi n modificada.

SQL> alter PLUGGABLE DATABASE MASCO OPEN;
Base de datos de conexi n modificada.

SQL>
```

Para desconectarlo/desenchufarla la sentencia ser a la misma pero con CLOSE.

Oracle XE 18c est  restringida a un m ximo de 3 PDB's y un total de 12GB de almacenamiento (incluye la suma del espacio ocupado por cada PDB).

Crear PDB

- Desde el botón Crear, tenemos acceso a crear nuevos PDB, especificando usuario, path de almacenamiento de archivos y el límite de recursos asignado a ese PDB.

The image displays three sequential screenshots of the 'Create PDB From Seed' dialog box in Oracle SQL Plus, illustrating the configuration steps for creating a new Pluggable Database (PDB).

- General Tab:** The 'PDB Name' is set to 'EJEMPLO'. Under 'PDB Administrative User', the 'Username' is 'PDBADMIN1', and the 'Password' and 'Confirm Password' fields are masked with asterisks. The 'Grant DBA Role' checkbox is checked.
- Storage Tab:** The 'Use Oracle-Managed Files' checkbox is unchecked. The 'Datafile Location' is specified as 'C:\app\Administrador\product\18.0.0\XE\datafile\EJEMPLO\'. The 'Storage Unlimited' checkbox is checked.
- Resource Limits Tab:** The 'Current Active Resource Plan' is 'DEFAULT_CDB_PLAN'. The 'Shares' are set to 1, the 'CPU Utilization Limit (%)' is 100, and the 'Parallel Server Limit (%)' is 100.

Each screenshot includes a 'Show SQL' button and 'OK'/'Cancel' buttons at the bottom.

- Si queremos crear el PDB desde SQLPlus las sentencias, para las opciones mostradas serían, tal cual nos genera el acceso web de Oracle 18c XE desde el botón Show SQL:

```
create pluggable database "EJEMPLO" admin user
"PDBADMIN1" identified by *****
file_name_convert =
('C:\APP\ADMINISTRADOR\PRODUCT\18.0.0\OR
ADATA\XE\PDBSEED\',
'C:\app\Administrador\product\18.0.0\XE\datafile\EJ
EMPLO\');

alter pluggable database "EJEMPLO" open read
write;

alter session set container = "EJEMPLO";
grant dba to "PDBADMIN1";

-- declare bind variables
var b1 varchar2(4000);
var b2 varchar2(4000);
var b3 number;
var b4 number;
var b5 number;
var b6 number;
var b7 number;

-- init bind values
```

Mª Luisa Pelayo Fernández

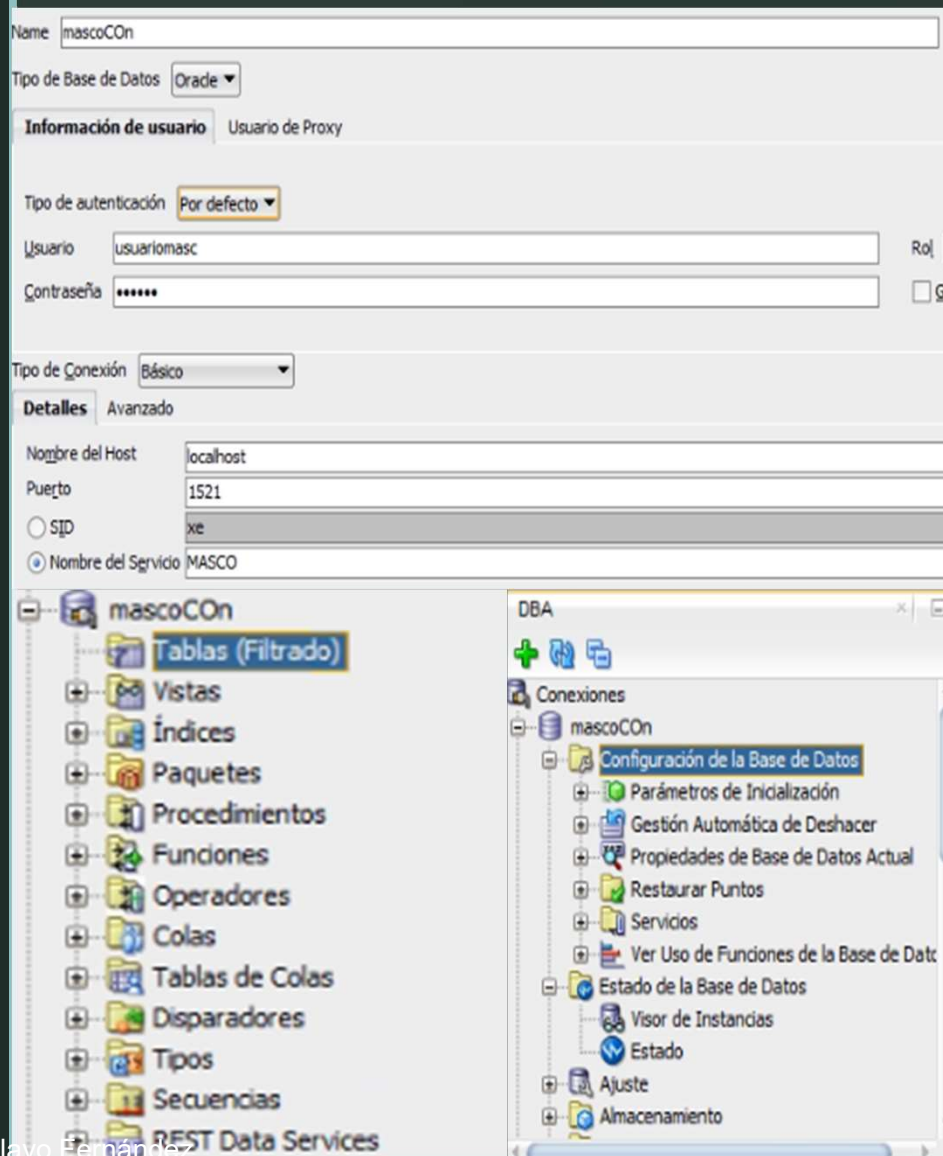
```
exec :b1 := 'DEFAULT_CDB_PLAN';
exec :b2 := 'EJEMPLO';
exec :b3 := 1;
exec :b4 := 100;
exec :b5 := 100;
exec :b6 := ;
exec :b7 := ;

begin  sys.dbms_resource_manager.clear_pending_area();
sys.dbms_resource_manager.create_pending_area();
sys.dbms_resource_manager.create_cdb_plan_directive(
    plan                => :b1,
    pluggable_database  => :b2,
    shares              => :b3,
    utilization_limit   => :b4,
    parallel_server_limit => :b5,
    memory_min          => :b6,
    memory_limit        => :b7);

sys.dbms_resource_manager.validate_pending_area();
sys.dbms_resource_manager.submit_pending_area();
end;

/;
```

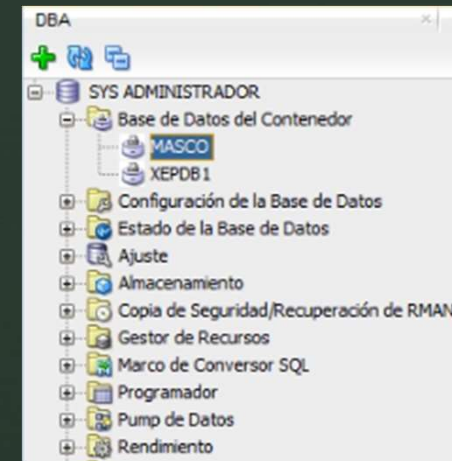
Conexión desde Oracle SQL Developer a una PDB



Usuariomasc es un usuario creado en la PDB MASCO con permisos connect, DBA y resource. Una vez conectados todas las operaciones que realicemos de tipo DDL sólo se ejecutarán en la PDB MASCO.

Si desde el menú Ver activamos DBA y agregamos con el + la conexión que acabamos de crear, podemos ver y manejar el PDB.

Si desde DBA agregamos cualquier conexión que hayamos realizado con SYS y valores por defecto (XE) podremos ver los PDB que existen.



SQL Developer: Usuarios, Roles y Permisos

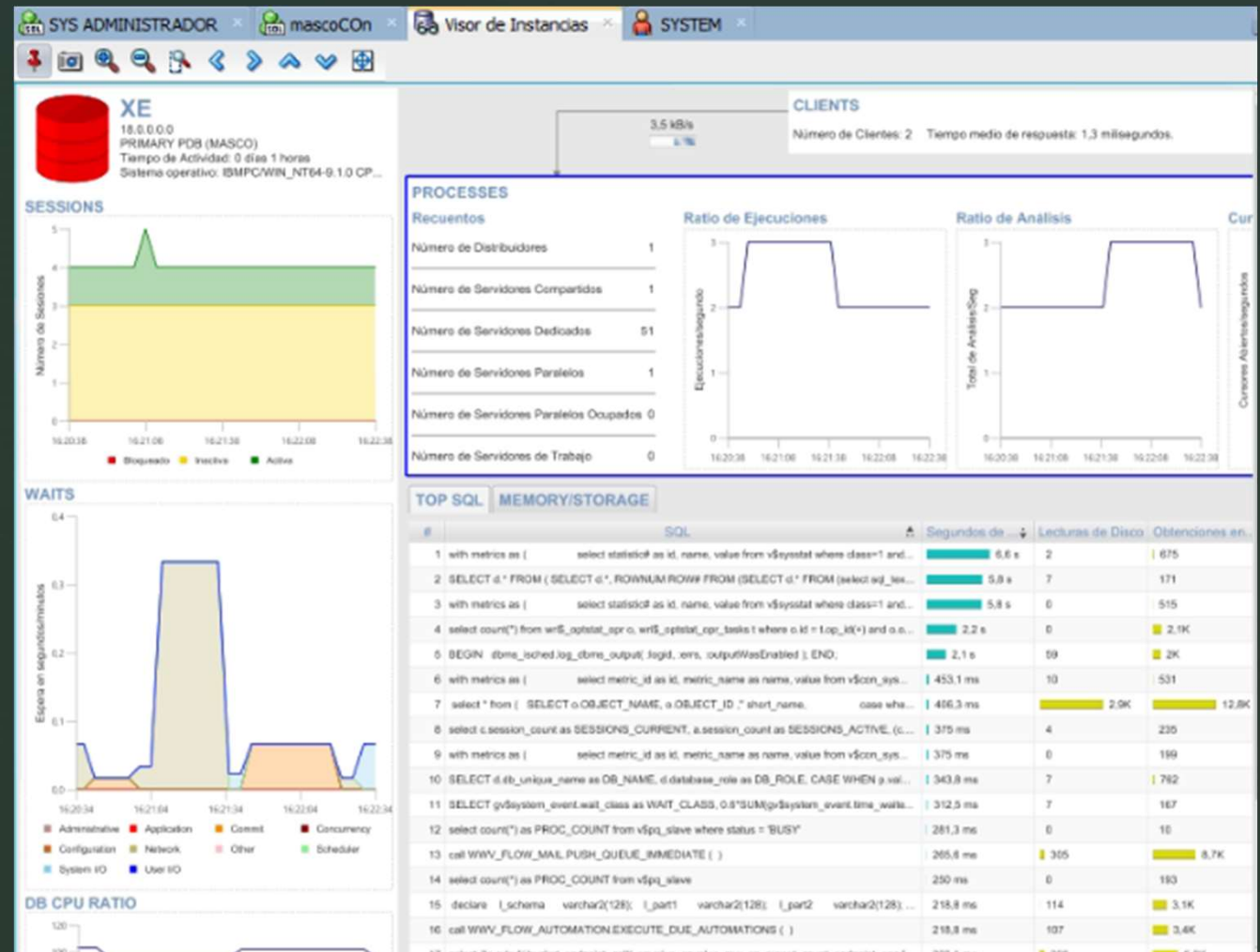
- El usuario administrador en OracleDB es SYS. SYSTEM tiene permisos similares pero por ejemplo, no tiene permisos de replicación de bases de datos.
- Cuando conectamos con SYS, puesto que puede ejercer distintos roles, debemos indicarle el rol con el que accedemos (DBA). Desde DBA situados en una conexión podemos ver usuarios, roles que tiene y los permisos de cada rol.

The screenshot displays the SQL Developer interface with the DBA connection selected. The left pane shows the 'Roles' tree under the 'Seguridad' (Security) folder. The right pane shows the 'Roles Privileges' window, which lists the roles and their associated privileges.

Role	Status	Expiration Date	System	Temp	Default
ORDSYS	EXPIRED & LOCKED	04/02/19	SYSAUX	TEMP	DEFAU
OUTLN	EXPIRED & LOCKED	04/02/19	SYSTEM	TEMP	DEFAU
REMOTE_SCHEDULER_AGENT	EXPIRED & LOCKED	04/02/19	USERS	TEMP	DEFAU
SI_INFORMTN_SCHEMA	EXPIRED & LOCKED	04/02/19	SYSAUX	TEMP	DEFAU
SYSBACKUP	EXPIRED & LOCKED	04/02/19	USERS	TEMP	DEFAU
SYSDG	EXPIRED & LOCKED	04/02/19	USERS	TEMP	DEFAU
SYSKM	EXPIRED & LOCKED	04/02/19	USERS	TEMP	DEFAU
SYS	OPEN	03/08/19	SYSTEM	TEMP	DEFAU
SYSRAC	EXPIRED & LOCKED	04/02/19	USERS	TEMP	DEFAU
SYSTEM	OPEN	03/04/22	SYSTEM	TEMP	DEFAU

The 'Roles Privileges' window also shows the 'Roles Otorgados' (Granted Roles) and 'Privilegios Otorgados del Sistema' (System Granted Privileges) for the selected role.

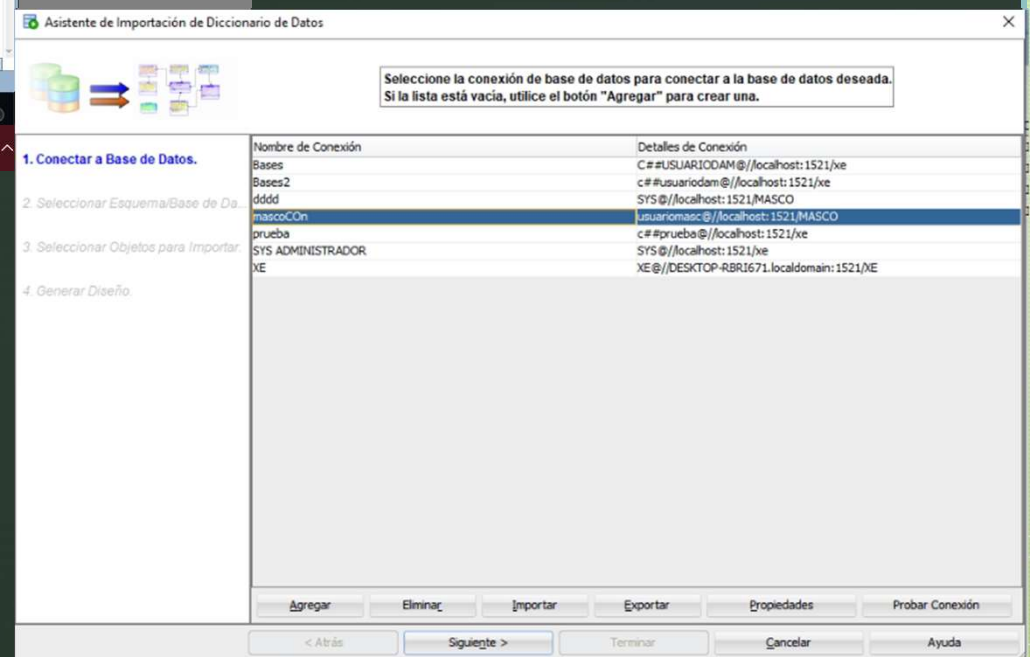
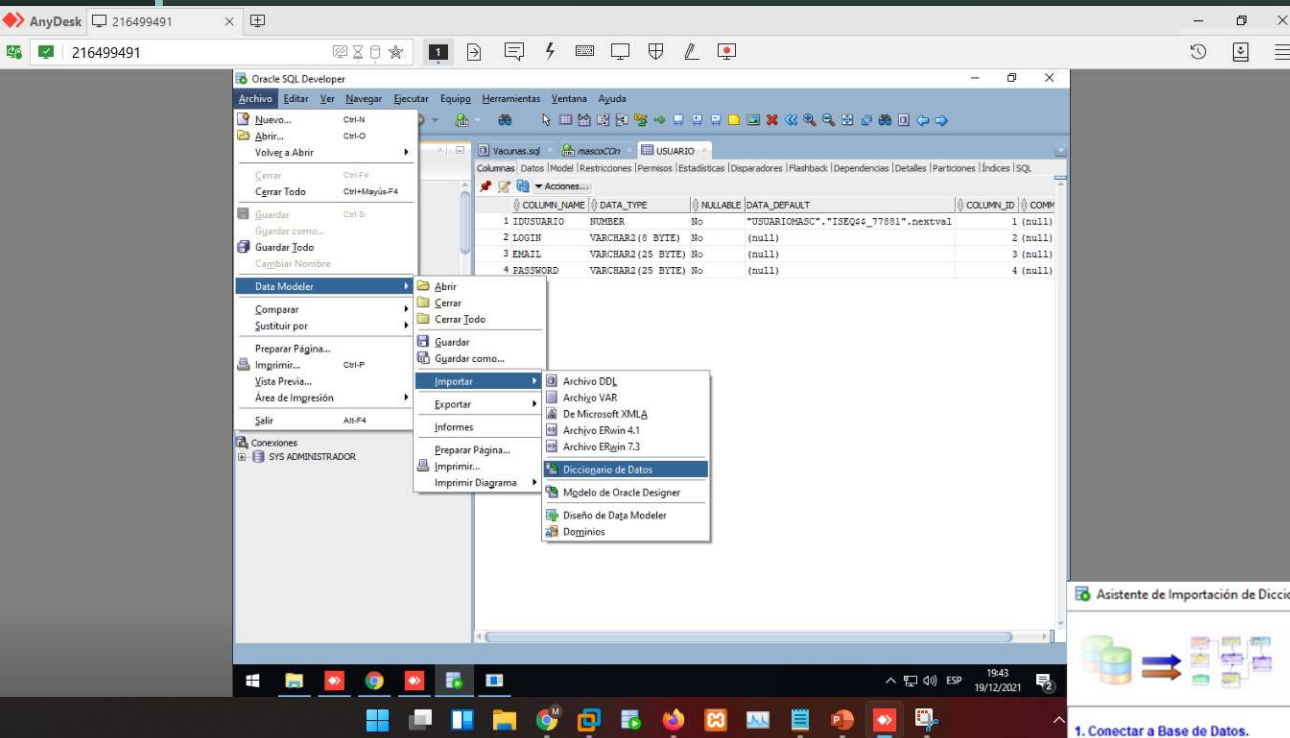
SQL Developer Visor de instancias

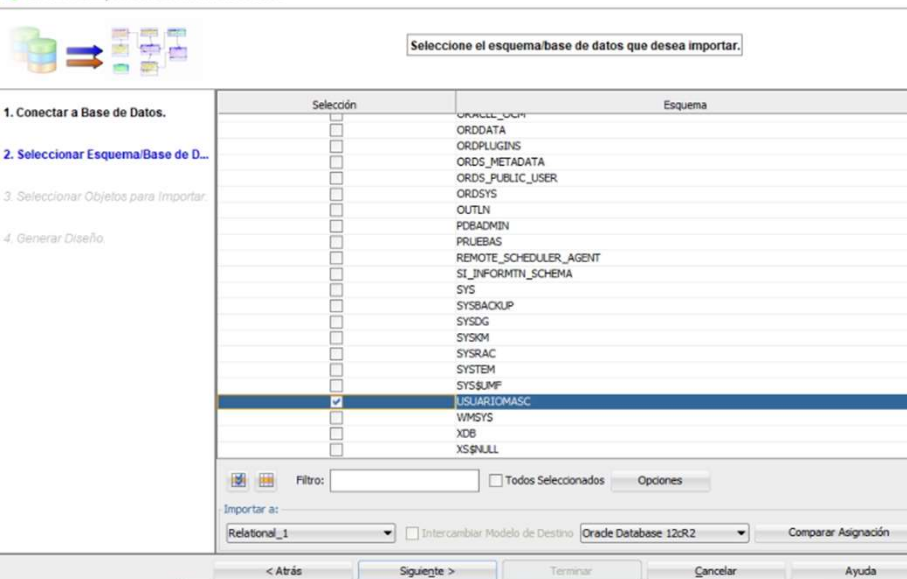


Nos permite monitorizar en tiempo real el uso de recursos de la PDB.

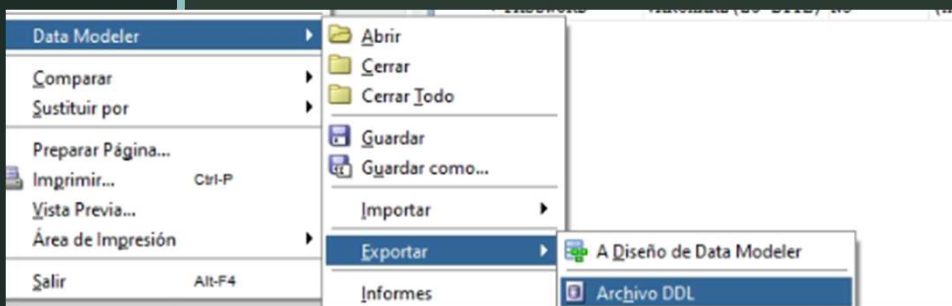
ACLARACIONES UT02

Generar Diagrama desde SQL Developer

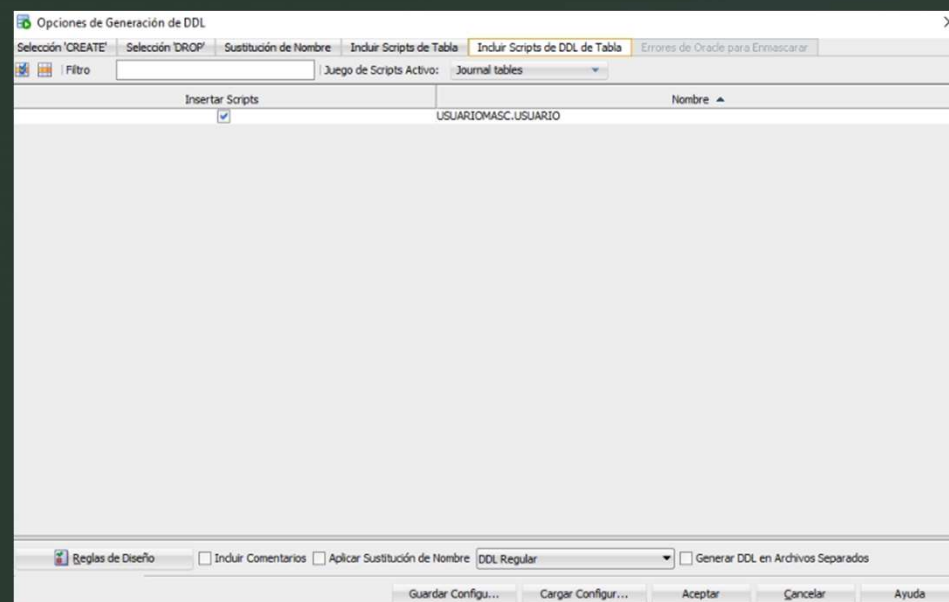




Generar DDL SQL Developer



Selección	Esquema	Nombre de Objeto
<input checked="" type="checkbox"/>	USUARIOMASC	USUARIO



Script SQL manual

```
CREATE TABLE usuario(
```

```
IdUsuario number generated always as identity,
```

```
Login varchar2(8) NOT NULL,
```

```
Email varchar2(25) NOT NULL,
```

```
Password varchar(25) NOT NULL
```

```
);
```

```
ALTER TABLE usuario ADD CONSTRAINT usuario_pk PRIMARY KEY(IdUsuario);
```

```
ALTER TABLE usuario ADD CONSTRAINT usuario_uk UNIQUE (Login);
```

```
ALTER TABLE usuario ADD CONSTRAINT usuario_uk2 UNIQUE (Email);
```

DDL GENERADO POR DESDE SQL DEVELOPER

```
-- Generado por Oracle SQL Developer Data Modeler 21.2.0.165.1515
CREATE USER usuariomasc IDENTIFIED BY ACCOUNT UNLOCK ;
-- predefined type, no DDL - MDSYS.SDO_GEOMETRY
-- predefined type, no DDL - XMLTYPE
CREATE TABLE usuariomasc.usuario (
  idusuario NUMBER GENERATED ALWAYS AS IDENTITY ( START WITH 1 CACHE 20 ) NOT NULL,
  login   VARCHAR2(8 BYTE) NOT NULL,
  email   VARCHAR2(25 BYTE) NOT NULL,
  password VARCHAR2(25 BYTE) NOT NULL
)
PCTFREE 10 PCTUSED 40 TABLESPACE users LOGGING
  STORAGE ( PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST GROUPS
1 BUFFER_POOL DEFAULT )
NO INMEMORY;
CREATE UNIQUE INDEX usuariomasc.usuario_pk ON usuariomasc.usuario (idusuario  ASC )
  TABLESPACE users PCTFREE 10
  STORAGE ( PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST
GROUPS 1 BUFFER_POOL DEFAULT )
LOGGING;
```

DDL GENERADO POR SQL DEVELOPER



```
CREATE UNIQUE INDEX usuariomasc.usuario_uk ON usuariomasc.usuario (login ASC )
    TABLESPACE users PCTFREE 10 STORAGE ( PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED
FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT )
    LOGGING;

CREATE UNIQUE INDEX usuariomasc.usuario_uk2 ON
    usuariomasc.usuario (email ASC ) TABLESPACE users PCTFREE 10
    STORAGE ( PCTINCREASE 0 MINEXTENTS 1 MAXEXTENTS UNLIMITED FREELISTS 1 FREELIST
GROUPS 1 BUFFER_POOL DEFAULT )
    LOGGING;

ALTER TABLE usuariomasc.usuario ADD CONSTRAINT usuario_pk PRIMARY KEY ( idusuario )
    USING INDEX usuariomasc.usuario_pk;

ALTER TABLE usuariomasc.usuario ADD CONSTRAINT usuario_uk UNIQUE ( login )
    USING INDEX usuariomasc.usuario_uk;

ALTER TABLE usuariomasc.usuario ADD CONSTRAINT usuario_uk2 UNIQUE ( email )
    USING INDEX usuariomasc.usuario_uk2;

CREATE TABLE USUARIOMASC.USUARIO_JN (JN_OPERATION CHAR(3) NOT NULL ,JN_ORACLE_USER
VARCHAR2(30) NOT NULL ,JN_DATETIME DATE NOT NULL ,JN_NOTES VARCHAR2(240) ,JN_APPLN
VARCHAR2(35) ,JN_SESSION NUMBER(38) ,IDUSUARIO NUMBER NOT NULL ,LOGIN VARCHAR2 (8 BYTE) NOT
NULL ,EMAIL VARCHAR2 (25 BYTE) NOT NULL,PASSWORD VARCHAR2 (25 BYTE) NOT NULL);
```

DDL GENERADO POR SQL DEVELOPER

```
CREATE OR REPLACE TRIGGER USUARIOMASC.USUARIO_JNtrg
AFTER
INSERT OR
UPDATE OR
DELETE ON USUARIOMASC.USUARIO for each row
Declare
rec USUARIOMASC.USUARIO_JN%ROWTYPE;
blank USUARIOMASC.USUARIO_JN%ROWTYPE;
BEGIN
  rec := blank;
  IF INSERTING OR UPDATING THEN
    rec.IDUSUARIO := :NEW.IDUSUARIO;
    rec.LOGIN := :NEW.LOGIN;
    rec.EMAIL := :NEW.EMAIL;
    rec.PASSWORD := :NEW.PASSWORD;
    rec.JN_DATETIME := SYSDATE;
    rec.JN_ORACLE_USER := SYS_CONTEXT ('USERENV', 'SESSION_USER');
    rec.JN_APPLN := SYS_CONTEXT ('USERENV', 'MODULE');
    rec.JN_SESSION := SYS_CONTEXT ('USERENV', 'SESSIONID');
    IF INSERTING THEN
      rec.JN_OPERATION := 'INS';
    ELSIF UPDATING THEN
      rec.JN_OPERATION := 'UPD';
    END IF;
  ELSIF DELETING THEN
    rec.IDUSUARIO := :OLD.IDUSUARIO;
    rec.LOGIN := :OLD.LOGIN;
    rec.EMAIL := :OLD.EMAIL;
    rec.PASSWORD := :OLD.PASSWORD;
    rec.JN_DATETIME := SYSDATE;
    rec.JN_ORACLE_USER := SYS_CONTEXT ('USERENV', 'SESSION_USER');
    rec.JN_APPLN := SYS_CONTEXT ('USERENV', 'MODULE');
    rec.JN_SESSION := SYS_CONTEXT ('USERENV', 'SESSIONID');
    rec.JN_OPERATION := 'DEL';
  END IF;
  INSERT into USUARIOMASC.USUARIO_JN VALUES rec;
END;
```


Equivalencias Sintaxis DDL



```
CREATE TABLE usuario(  
  IdUsuario number generated always as identity,  
  Login varchar2(8) NOT NULL,  
  Email varchar2(25) NOT NULL,  
  Contraseña varchar(25)  
);
```

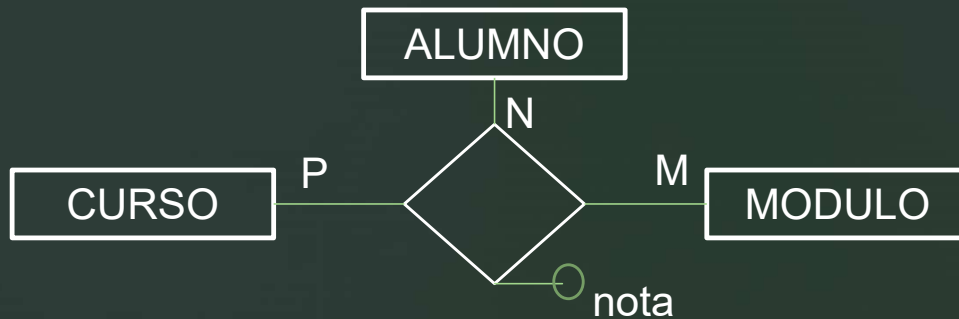
```
ALTER TABLE usuario ADD CONSTRAINT usuario_pk PRIMARY KEY(IdUsuario);  
ALTER TABLE usuario ADD CONSTRAINT usuario_uk UNIQUE (Login);  
ALTER TABLE usuario ADD CONSTRAINT usuario_uk2 UNIQUE (Email);  
ALTER TABLE usuario ADD CONSTRAINT usuario_ck CHECK(length(rtrim(ltrim(Login)))>4);  
ALTER TABLE usuario MODIFY Contraseña varchar2(25) DEFAULT '123456';  
ALTER TABLE usuario ADD FechaAlta DATE DEFAULT SYSDATE;
```

```
CREATE TABLE usuario(  
  IdUsuario number generated always as identity PRIMARY KEY,  
  Login varchar2(8) NOT NULL UNIQUE CHECK(length(rtrim(ltrim(Login)))>4),  
  Email varchar2(25) NOT NULL UNIQUE,  
  Contraseña varchar2(25) DEFAULT ON NULL '123456',  
  FechaAlta date DEFAULT SYSDATE  
);
```

```
CREATE TABLE usuario(  
  IdUsuario number generated always as identity,  
  Login varchar2(8) NOT NULL,  
  Email varchar2(25) NOT NULL,  
  Contraseña varchar2(25) DEFAULT ON NULL '123456',  
  FechaAlta date DEFAULT SYSDATE,  
  constraint pk_usuario PRIMARY KEY(IdUsuario),  
  constraint uk_usuario_login UNIQUE(login),  
  constraint uk_usuario_email UNIQUE(email),  
  constraint ck_usuario_login CHECK(length(rtrim(ltrim(Login)))>4)  
);
```

ACLARACIONES UT03

Transformación de relaciones ternarias



Un curso tiene varios módulos.
 Un curso puede realizarse por varios alumnos.
 Un módulo puede impartirse en varios cursos.
 Un módulo puede ser cursado por varios alumnos.

Alumno (idAlumno,...);
 Curso (idCurso,....);
 Modulo (idModulo,....);
 Calificacion (idAlumno,idCurso,idModulo,nota);



Formas normales

Basadas en el análisis de los atributos y sus dependencias:

- Funcional: $A \rightarrow B$ B depende funcionalmente de A. A un elemento de A le corresponde un elemento de B. dni \rightarrow nombre
- Funcional completa: $A_1, A_2 \dots A_k \rightarrow B$ dni.empresa \rightarrow sueldo
2FN dni.modulo.evaluacion \rightarrow nota
- Transitiva: $A \rightarrow B \rightarrow C$, C depende transitivamente de A.
num_matricula \rightarrow grupo_asignado \rightarrow aula_grupo. Aula_grupo depende transitivamente de num_matricula. (3FN)

1FN: todos los atributos son atómicos. La tabla tiene una clave primaria única. La clave no tiene atributos nulos. Todos los atributos no clave dependen funcionalmente de la clave. Debe existir independencia en el orden de filas y columnas. No hay grupos repetidos.

2FN: todos los atributos que no sean clave tienen dependencia funcional completa de la clave.

Ejemplo, una tabla trabajadores en la que recogemos dni, empresa y dirección del trabajo. La clave sería dni y empresa, pero la dirección del trabajo no depende completamente de la clave sino de una parte de ella.

3FN: cada atributo que no sea clave primaria no depende transitivamente de la clave.

Ejemplo, tabla con el mejor jugador de cada equipo año. Tendremos (nombre equipo, año, nombre jugador, nacionalidad), nombre equipo y año serán clave primaria. Nacionalidad depende de nombre jugador que a su vez depende de la clave primaria.

Formas normales II



FNBC: los atributos determinantes son claves candidatas. Por ejemplo, tabla con idTrabajador, idDepartamento, idResponsable, donde el idResponsable es la persona responsable del trabajador. La clave sería, si cada trabajador puede trabajar en varios departamentos y tener distintos responsables (idTrabajador, idDepartamento, idResponsable). Pero si cada responsable lo es de un único departamento, entonces idResponsable dependería de idDepartamento, lo que convierte a idResponsable en determinante, pero no es clave candidata.

Este problema se soluciona creando otra tabla (idDepartamento, idResponsable) y eliminando idResponsable de la entidad anterior.

Ejemplo – Tabla Pedidos

idPedido	Fecha	idCliente	nombreCli	Ciudad	idArticulo	nombreArticulo	cantidad	precio
1	23/12/2020	3	Pepe Pérez	Santander	2	Donuts	4	1.25
1	23/12/2020	3	Pepe Pérez	Santander	8	Gominola	50	2.25
1	23/12/2020	3	Pepe Pérez	Santander	5	Cacao	1	3.23
2	14/02/2021	5	María Ro	Madrid	2	Donuts	8	1.25
2	14/02/2021	5	María Ro	Madrid	10	Café	2	3.50
3	16/09/2021	10	Juan Lanza	Barcelona	3	Croissants	10	0.50

1FN: no está en 1FN, lo primero debido a que existen grupos repetidos. IdArticulo,nombreArticulo,cantidad y precio es un grupo repetido para cada idPedido,fecha,idcliente, nombrecli y ciudad. Debemos sacarlos a otra tabla con la pk de la tabla base:

Pedido(idPedido, Fecha,idCliente, nombreCli, Ciudad)

ArticuloPedido(IdPedido,IdArticulo,nombreArticulo,cantidad,precio)

idPedido	Fecha	idCliente	nombreCli	Ciudad
1	23/12/2020	3	Pepe Pérez	Santander
2	14/02/2021	5	María Ro	Madrid
3	16/09/2021	10	Juan Lanza	Barcelona

idPedido	idArticulo	nombreArticulo	cantidad	precio
1	2	Donuts	4	1.25
1	8	Gominola	50	2.25
1	5	Cacao	1	3.23
2	2	Donuts	8	1.25
2	10	Café	2	3.50
3	3	Croissants	10	0.50

Ejemplo Pedidos

- 2FN: eliminar cualquier columna no clave que no dependa de la clave.

Pedido, ya está en 2FN dado que un valor de la clave determina un único valor en los atributos no clave. Todas las columnas son dependientes de la clave primaria idPedido.

ArticuloPedido no está en 2FN dado que nombreArticulo y Precio no dependen del idPedido sino del IdArticulo → creamos la tabla articulo(idArticulo, nombreArticulo, Precio) y eliminamos esas columnas de la tabla ArticuloPedido

idArticulo	nombreArticulo	precio
2	Donuts	1.25
8	Gominola	2.25
5	Cacao	3.23
10	Café	3.50
3	Croissants	0.50

idPedido	idArticulo	cantidad
1	2	4
1	8	50
1	5	1
2	2	8
2	10	2
3	3	10

Ejemplo Pedidos

3FN: Eliminar cualquier columna no clave que sea dependiente de otra no clave. Artículo y ArtículoPedido están en 3FN, no así Pedido, dado que nombreCli y ciudad dependen de IdCliente → creamos nueva tabla cliente con esos campos y el atributo que usaremos como clave del que dependen. Cliente(idCliente, nombreCli, ciudad)

idCliente	nombreCli	Ciudad
3	Pepe Pérez	Santander
5	María Ro	Madrid
10	Juan Lanza	Barcelona

idPedido	Fecha	idCliente
1	23/12/2020	3
2	14/02/2021	5
3	16/09/2021	10

pedido(idPedido, fecha, idCliente)
 cliente(idCliente, nombreCli, ciudad)
 articulo(idArticulo, nombreArticulo, Precio)
 articuloPedido(idPedido, IdArticulo, cantidad)

FNBC: para cada dependencia funcional entre A y B, A debería ser una superclave, Un atributo o columna de una tabla identifica de forma única otro atributo (s) o columna (s) de la misma tabla. Una sola clave o un grupo de varias claves que podrían identificar de forma única una sola fila en una tabla se puede denominar Super Clave. Estas tablas cumplen este criterio.

Gracias por su atención y ánimo!!!