

Instituto FOC
Módulo de desarrollo de aplicaciones web

BRUNO MARENCO CERQUEIRA

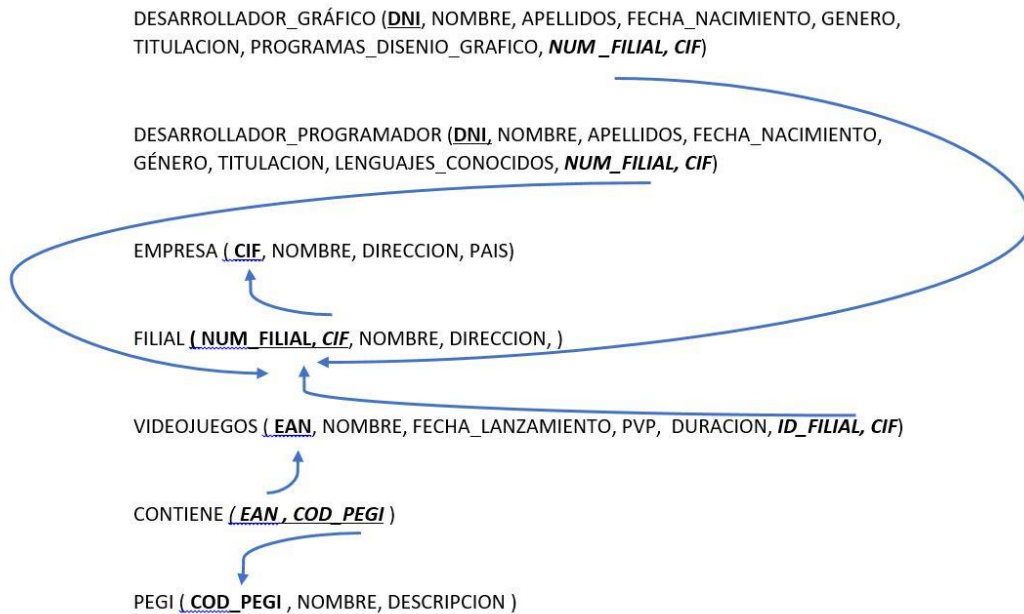
Tarea Individual 3: Creación de bases de datos relacionales

Febrero/2018

Índice

Índice.....	2
Tomando como referencia el modelo relacional de la imagen. Se solicita crear el fichero SQL con sentencias de creación de tablas:.....	3
Crear las tablas especificadas en el grafo relacional, teniendo en cuenta las reglas de nomenclatura en la creación de las mismas.	3
Seleccionar el tipo de dato más adecuado para cada una de las columnas que conforman cada una de las tablas.	3
Tablas Desarrollador_grafico y Desarrollador_programador	4
Tabla Empresa	4
Tabla Filial	4
Tabla Videojuegos	4
Tabla Contiene	5
Tabla Pegi	5
Establecer las restricciones de clave primaria en cada una de las tablas.	5
Tablas Desarrollador_grafico y Desarrollador_programador	5
Tabla Empresa	5
Tabla Filial	6
Tabla Videojuegos	6
Tabla Contiene	6
Tabla Pegi	6
Establecer las restricciones de integridad que se crean convenientes.....	6
Tablas Desarrollador_grafico y Desarrollador_programador	6
Tabla Empresa	7
Tabla Filial	8
Tabla Videojuegos	8
Tabla Contiene	9
Tabla Pegi	9
Código.....	9
Capturas.....	12

Tomando como referencia el modelo relacional de la imagen. Se solicita crear el fichero SQL con sentencias de creación de tablas:



Crear las tablas especificadas en el grafo relacional, teniendo en cuenta las reglas de nomenclatura en la creación de las mismas.

Para la creación de tabla se utiliza el comando siguiente:

```
CREATE TABLE "nombre de la tabla" (
);
```

Siguiendo el modelo relacional de la imagen, creamos 7 tablas: **Desarrollador_grafico**, **Desarrollador_programador**, **Empresa**, **Filial**, **Videojuegos**, **Contiene** y **Pegi**. Cada una con sus respectivas columnas.

Seleccionar el tipo de dato más adecuado para cada una de las columnas que conforman cada una de las tablas.

Para cada columna debemos seleccionar los tipos de datos más adecuados dependiendo de la información a registrar aprovechando al máximo el uso de la memoria y la integridad de datos.

Tablas **Desarrollador_grafico** y **Desarrollador_programador**.

Las dos tablas tienen datos muy parecidos, el tipo de dato más habitual es VARCHAR2(), este tipo de dato se usa para almacenar caracteres alfanuméricos de una longitud máxima determinada dependiendo de la longitud de cada información. Por ello se utiliza en las columnas de las dos tablas: *dni* (con 9 caracteres), *nombre* (con hasta 15 caracteres), *apellidos* (con hasta 30 caracteres), *genero* (con hasta 6 caracteres, para identificar hombre o mujer), *titulación* (con hasta 30 caracteres), *cif* (con 9 caracteres).

También tenemos el tipo de dato NUMBER(), que almacena números enteros y flotantes, indicando cifras y decimales. Utilizamos este tipo de dato para la columna de las dos tablas: *id_filial* (que es un número indicativo de la filial, que creemos comprende hasta 99 como máximo).

En la tabla **Desarrollador_grafico** tenemos la columna *programas_diseño_grafico* (que almacena el número de programas conocidos, 99 como máximo) y en la tabla **Desarrollador_programador** tenemos *lenguajes_conocidos* (que almacena el número de programas conocidos, 99 como máximo).

Por último, en las dos tablas hay una columna con el tipo de dato DATE, que almacena datos temporales, en este caso se usa para la columna *fecha_nacimiento*.

Tabla **Empresa**.

El tipo de dato VARCHAR2() para las columnas: *cif* (con 9 caracteres), *nombre* (con hasta 15 caracteres), *dirección* (con hasta 50 caracteres), *país* (con hasta 30 caracteres).

Tabla **Filial**.

El tipo de dato VARCHAR2() para las columnas: *cif_empresa* (con 9 caracteres), *nombre* (con hasta 15 caracteres), *dirección* (con hasta 50 caracteres). Y definimos el dato NUMBER() para la columna de *num_filial*, donde establecemos que las filiales se identifican por un número de dos cifras.

Tabla **Videojuegos**.

El tipo de dato VARCHAR2() para las columnas: *cif* (con 9 caracteres), *nombre* (con hasta 15 caracteres) *pvp* (con 3 caracteres).

Y definimos el dato NUMBER() para la columna de *ean*, un código de barras de identificación de videojuegos compuesto por 13 cifras. Para la columna

duración, establecemos un valor de 7 cifras con dos decimales en segundos y la *id_filial* que identifica la filial.

Está el tipo de dato DATE, que almacena datos temporales, en este caso se usa para la columna *fecha_lanzamiento*.

Tabla **Contiene**.

Definimos el dato NUMBER() para la columna de *ean*, un código de barras de identificación de videojuegos compuesto por 13 cifras, y para la columna *cod_pegí*, que identifica la clasificación de contenido y establece las edades permitidas por juego, por lo que está compuesto por 2 cifras.

Tabla **Pegí**.

Definimos el dato NUMBER() para la columna *cod_pegí*, que identifica la clasificación de contenido y establece las edades permitidas por juego, por lo que está compuesto por 2 cifras. El tipo de dato VARCHAR2() para las columnas: *nombre* (con hasta 15 caracteres), *descripcion* (con hasta 100 caracteres).

Establecer las restricciones de clave primaria en cada una de las tablas.

Tablas **Desarrollador_grafico** y **Desarrollador_programador**.

En las tablas **Desarrollador_grafico** y **Desarrollador_programador** tenemos las claves primarias *dni* y las definimos a nivel de tabla.

CONSTRAINT desarrollador_grafico_pk PRIMARY KEY (dni)

CONSTRAINT desarrollador_programador_pk PRIMARY KEY (dni)

Tabla **Empresa**.

En la tabla **Empresa** tenemos la clave primaria *cif* y la definimos a nivel de tabla.

CONSTRAINT empresa_pk PRIMARY KEY (cif)

Tabla **Filial**.

En la tabla **Filial** tenemos la clave primaria compuesta de *num_filial* y *cif*. Así la definimos a nivel de tabla.

CONSTRAINT filial_pk PRIMARY KEY (num_filial, cif_empresa)

Tabla **Videojuegos**.

En la tabla **Videojuegos** está la clave primaria *ean* y la definimos a nivel de tabla.

CONSTRAINT videojuegos_pk PRIMARY KEY (ean)

Tabla **Contiene**.

En la tabla **Contiene** tenemos la clave primaria compuesta de *ean* y *cod_pegí* y la definimos a nivel de tabla.

CONSTRAINT contiene_pk PRIMARY KEY (ean, cod_pegí)

Tabla **Pegí**.

En la tabla **Pegí** definimos la clave *cod_pegí* como clave primaria y la definimos así.

CONSTRAINT pegí_pk PRIMARY KEY (cod_pegí)

Establecer las restricciones de integridad que se crean convenientes.

Tablas **Desarrollador_grafico** y **Desarrollador_programador**.

En las tablas **Desarrollador_grafico** y **Desarrollador_programador** tenemos las claves externas *id_filial* y *cif*, referentes a la tabla **Filial** y su clave primaria compuesta *num_filial* y *cif_empresa*, y las definimos a nivel de tabla para cada tabla.

Establecemos la opción **ON DELETE SET NULL** para que si la fila de la tabla principal se elimina, los valores de clave externa sean nulos, ya que puede cerrarse la filial pero mantener a los trabajadores y cambiarlos de filial.

**CONSTRAINT desarrollador_grafico_fk FOREIGN KEY (id_filial, cif)
REFERENCES Filial (num_filial, cif_empresa) ON DELETE SET NULL**

**CONSTRAINT desarrollador_programador_fk FOREIGN KEY (id_filial, cif)
REFERENCES Filial (num_filial, cif_empresa) ON DELETE SET NULL**

Establecemos la restricción **NOT NULL** a nivel de columna a varios datos en la tabla de Desarrollador_grafico colocando grafico en los nombres de los Constraint y en la tabla de Desarrollador_programador colocando pgr en los nombres de los Constraint.

**nombre VARCHAR2(15) CONSTRAINT nombre_grafico_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_grafico_nnull NOT
NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_grafico_nnull
NOT NULL,**

**nombre VARCHAR2(15) CONSTRAINT nombre_pgr_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_pgr_nnull NOT NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_pgr_nnull NOT
NULL,**

Establecemos la restricción **CHECK** para añadir condiciones a la columna *genero* para que el dato a insertar sea o Mujer o Hombre. Diferenciando en cada tabla con pgr y grafico.

**genero VARCHAR2(6) CONSTRAINT genero_grafico_ck CHECK
(genero='Hombre' OR genero='Mujer')**

**genero VARCHAR2(6) CONSTRAINT genero_pgr_ck CHECK
(genero='Hombre' OR genero='Mujer')**

Tabla Empresa.

Establecemos la restricción **NOT NULL** a nivel de columna a varios datos en la tabla.

**nombre VARCHAR2(15)CONSTRAINT nombre_nnull NOT NULL,
direccion VARCHAR2(50)CONSTRAINT direccion_nnull NOT NULL,
pais VARCHAR2(30)CONSTRAINT pais_nnull NOT NULL,**

Tabla **Filial**.

En la tabla **Filial** tenemos la clave externa *cif* referentes a la tabla empresa y la definimos a nivel de tablas.

Establecemos la opción **ON DELETE CASCADE** para que si la fila de la tabla principal se elimina, los valores de clave externa se supriman también, ya que al cerrarse la empresa, se cierran las filiales.

```
CONSTRAINT filial_fk FOREIGN KEY (cif_empresa) REFERENCES  
Empresa(cif) ON DELETE CASCADE
```

Establecemos la restricción **NOT NULL** a nivel de columna a varios datos en la tabla.

```
nombre VARCHAR2(15)CONSTRAINT nombre_filial_nnnull NOT NULL,  
direccion VARCHAR2(50)CONSTRAINT direccion_filial _nnnull NOT NULL,
```

Tabla **Videojuegos**.

En la tabla **Videojuegos** tenemos las claves externas *id_filial* y *cif*, referentes a la tabla filial y su clave primaria compuesta *num_filial* y *cif_empresa*, y las definimos a nivel de tabla.

Establecemos la opción **ON DELETE SET NULL** para que si la fila de la tabla principal se elimina, los valores de clave externa sean nulos, ya que puede cerrarse la filial pero mantener los videojuegos y cambiarlos de filial.

```
CONSTRAINT videojuegos_fk FOREIGN KEY (id_filial, cif) REFERENCES  
Filial (num_filial, cif_empresa) ON DELETE SET NULL
```

Establecemos la restricción **NOT NULL** a nivel de columna en *nombre* en la tabla.

```
nombre VARCHAR2(15)CONSTRAINT nombre_vdj_nnnull NOT NULL,
```


Establecemos la restricción **CHECK** a nivel de columna a *pvp* en la tabla para añadir condiciones a la columna *pvp* para que el dato a insertar sea o YES o NOT.

```
pvp VARCHAR2(3) CONSTRAINT pvp_ck CHECK (pvp='YES' OR
pvp='NOT'),
```

Tabla **Contiene**.

En la tabla **Contiene** tenemos dos claves externas de dos tablas, en primer lugar, la clave externa *ean* referente a la tabla **Videojuegos** y en segundo lugar, la clave externa *cod_pegí* referente a la tabla **Pegí**.

Establecemos la opción **ON DELETE SET NULL**, ya que puede eliminarse el código pegí pero se puede mantener el videojuego sin pegí o cambiarlo.

Establecemos la opción **ON DELETE CASCADE**, ya que al eliminarse el videojuego, se elimina la relación con el código pegí.

```
CONSTRAINT contiene_fk1 FOREIGN KEY (ean) REFERENCES
Videojuegos (ean) ON DELETE CASCADE
CONSTRAINT contiene_fk2 FOREIGN KEY (cod_pegí) REFERENCES Pegí
(cod_pegí) ON DELETE SET NULL
```

Tabla **Pegí**.

Establecemos la restricción **NOT NULL** a nivel de columna a varios datos en la tabla.

```
nombre VARCHAR2(15) CONSTRAINT nombre_pegí_nnull NOT NULL,
descripcion VARCHAR2(50) CONSTRAINT descripcion_nnull NOT NULL,
```

Código

```
CREATE TABLE Empresa (
cif VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
```

```

direccion VARCHAR2(50) CONSTRAINT direccion_nnull NOT NULL,
pais VARCHAR2(30) CONSTRAINT pais_nnull NOT NULL,
CONSTRAINT empresa_pk PRIMARY KEY (cif)
);

```

```

CREATE TABLE Filial (
num_filial NUMBER(2),
cif_empresa VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_filial_nnull NOT NULL,
direccion VARCHAR2(50) CONSTRAINT direccion_filial_nnull NOT NULL,
CONSTRAINT filial_pk PRIMARY KEY (num_filial, cif_empresa),
CONSTRAINT filial_fk FOREIGN KEY (cif_empresa) REFERENCES Empresa(cif) ON
DELETE CASCADE
);

```

```

CREATE TABLE Desarrollador_grafico (
dni VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_grafico_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_grafico_nnull NOT NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_grafico_nnull NOT NULL,
genero VARCHAR2(6) CONSTRAINT genero_grafico_ck CHECK (genero='Hombre'
OR genero='Mujer'),
titulacion VARCHAR2(30),
programas_diseño_grafico NUMBER(2),
id_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT desarrollador_grafico_pk PRIMARY KEY (dni),
CONSTRAINT desarrollador_grafico_fk FOREIGN KEY (id_filial, cif) REFERENCES
Filial (num_filial, cif_empresa) ON DELETE SET NULL
);

```

```

CREATE TABLE Desarrollador_programador (
dni VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_pgr_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_pgr_nnull NOT NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_pgr_nnull NOT NULL,
genero VARCHAR2(6) CONSTRAINT genero_pgr_ck CHECK (genero='Hombre' OR
genero='Mujer'),

```

```

titulacion VARCHAR2(30),
lenguajes_conocidos NUMBER(2),
id_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT desarrollador_programador_pk PRIMARY KEY (dni),
CONSTRAINT desarrollador_programador_fk FOREIGN KEY (id_filial, cif)
REFERENCES Filial (num_filial, cif_empresa) ON DELETE SET NULL
);
CREATE TABLE Videojuegos (
ean NUMBER(13),
nombre VARCHAR2(15) CONSTRAINT nombre_vdj_nnull NOT NULL,
fecha_lanzamiento DATE,
pvp VARCHAR2(3) CONSTRAINT pvp_ck CHECK (pvp='YES' OR pvp='NOT'),
duracion NUMBER(7,2),
id_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT videojuegos_pk PRIMARY KEY (ean),
CONSTRAINT videojuegos_fk FOREIGN KEY (id_filial, cif) REFERENCES Filial
(num_filial, cif_empresa) ON DELETE SET NULL
);

CREATE TABLE Pegi (
cod_pegi VARCHAR2(2),
nombre VARCHAR2(15) CONSTRAINT nombre_pegi_nnull NOT NULL,
descripcion VARCHAR2(100) CONSTRAINT descripcion_nnull NOT NULL,
CONSTRAINT pegi_pk PRIMARY KEY (cod_pegi)
);

CREATE TABLE Contiene (
ean NUMBER(13),
cod_pegi VARCHAR2(2),
CONSTRAINT contiene_pk PRIMARY KEY (ean, cod_pegi),
CONSTRAINT contiene_fk1 FOREIGN KEY (ean) REFERENCES Videojuegos (ean)
ON DELETE CASCADE,
CONSTRAINT contiene_fk2 FOREIGN KEY (cod_pegi) REFERENCES Pegi
(cod_pegi) ON DELETE SET NULL
);

```

Capturas

SQL y tabla de Desarrollador_grafico.

127.0.0.1:8080/apex/f?p=4500:1003:3938327116761581::NO::

ORACLE Application Express

Home Application Builder SQL Workshop Team Development Administration

Home > SQL Workshop > SQL Commands

☒ Autocommit Rows 10 Save Run

```
CREATE TABLE Desarrollador_grafico (
dni VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_nnull NOT NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_nnull NOT NULL,
genero VARCHAR2(6) CONSTRAINT genero_ck CHECK (genero='Hombre' OR genero='Mujer'),
titulacion VARCHAR2(30),
programas_diseño_grafico NUMBER(2),
num_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT desarrollador_grafico_pk PRIMARY KEY (dni),
CONSTRAINT desarrollador_grafico_fk FOREIGN KEY (num_filial, cif) REFERENCES Filial (num_filial, cif) ON DELETE SET NULL
);
```

DESARROLLADOR_GRAFICO

TableDataIndexesModelConstraintsGrantsStatisticsUI DefaultsTriggersDependenciesSQL

Add ColumnModify ColumnRename ColumnDrop ColumnRenameCopyDropTruncateCreate Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
DNI	VARCHAR2(9)	No	-	1
NOMBRE	VARCHAR2(15)	No	-	-
APELLIDOS	VARCHAR2(30)	No	-	-
FECHA_NACIMIENTO	DATE	No	-	-
GENERO	VARCHAR2(6)	Yes	-	-
TITULACION	VARCHAR2(30)	Yes	-	-
PROGRAMAS_DISEÑO_GRAFICO	NUMBER(2,0)	Yes	-	-
ID_FILIAL	NUMBER(2,0)	Yes	-	-
CIF	VARCHAR2(9)	Yes	-	-
				1 - 9

Download

SQL y tabla de Desarrollador_programador.

```
CREATE TABLE Desarrollador_programador (
dni VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
apellidos VARCHAR2(30) CONSTRAINT apellidos_nnull NOT NULL,
fecha_nacimiento DATE CONSTRAINT fecha_nacimiento_nnull NOT NULL,
genero VARCHAR2(6) CONSTRAINT genero_ck CHECK (genero='Hombre' OR genero='Mujer'),
titulacion VARCHAR2(30),
lenguajes_conocidos NUMBER(2),
num_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT desarrollador_programador_pk PRIMARY KEY (dni),
CONSTRAINT desarrollador_programador_fk FOREIGN KEY (num_filial, cif) REFERENCES Filial (num_filial, cif) ON DELETE SET NULL
);
```

DESARROLLADOR_PROGRAMADOR

Table

Data

Indexes

Model

Constraints

Grants

Statistics

UI Defaults

Triggers

Dependencies

SQL

Add Column

Modify Column

Rename Column

Drop Column

Rename

Copy

Drop

Truncate

Create Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
DNI	VARCHAR2(9)	No	-	1
NOMBRE	VARCHAR2(15)	No	-	-
APELLIDOS	VARCHAR2(30)	No	-	-
FECHA_NACIMIENTO	DATE	No	-	-
GENERO	VARCHAR2(6)	Yes	-	-
TITULACION	VARCHAR2(30)	Yes	-	-
LENGUAJES_CONOCIDOS	NUMBER(2,0)	Yes	-	-
ID_FILIAL	NUMBER(2,0)	Yes	-	-
CIF	VARCHAR2(9)	Yes	-	-

1 - 9

Download

SQL y tabla de Empresa.

```
CREATE TABLE Empresa (
cif VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
direccion VARCHAR2(50) CONSTRAINT direccion_nnull NOT NULL,
pais VARCHAR2(30) CONSTRAINT pais_nnull NOT NULL,
CONSTRAINT empresa_pk PRIMARY KEY (cif)
);
```

EMPRESA

TableDataIndexesModelConstraintsGrantsStatisticsUI DefaultsTriggersDependenciesSQL

Add ColumnModify ColumnRename ColumnDrop ColumnRenameCopyDropTruncateCreate Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
CIF	VARCHAR2(9)	No	-	1
NOMBRE	VARCHAR2(15)	No	-	-
DIRECCION	VARCHAR2(50)	No	-	-
PAIS	VARCHAR2(30)	No	-	-
				1 - 4

Download

SQL y tablas de Filial y Videojuegos.

```

CREATE TABLE Filial (
num_filial NUMBER(2),
cif VARCHAR2(9),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
direccion VARCHAR2(50) CONSTRAINT direccion_nnull NOT NULL,
CONSTRAINT filial_pk PRIMARY KEY (num_filial, cif),
CONSTRAINT filial_fk FOREIGN KEY (cif) REFERENCES Empresa (cif) ON DELETE CASCADE
);

CREATE TABLE Videojuegos(
ean NUMBER(13),
nombre VARCHAR2(15) CONSTRAINT nombre_nnull NOT NULL,
fecha_lanzamiento DATE,
pvp BIT CONSTRAINT pvp_nnull NOT NULL,
duracion NUMBER(7,2),
id_filial NUMBER(2),
cif VARCHAR2(9),
CONSTRAINT videojuegos_pk PRIMARY KEY (ean),
CONSTRAINT videojuegos_fk FOREIGN KEY (id_filial, cif) REFERENCES Filial (num_filial, cif) ON DELETE SET NULL
);

```

FILIAL

Table

Data

Indexes

Model

Constraints

Grants

Statistics

UI Defaults

Triggers

Dependencies

SQL

Add Column

Modify Column

Rename Column

Drop Column

Rename

Copy

Drop

Truncate

Create Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
NUM_FILIAL	NUMBER(2,0)	No	-	1
CIF_EMPRESA	VARCHAR2(9)	No	-	2
NOMBRE	VARCHAR2(15)	No	-	-
DIRECCION	VARCHAR2(50)	No	-	-
				1 - 4

Download

aA



VIDEOJUEGOS

TableDataIndexesModelConstraintsGrantsStatisticsUI DefaultsTriggersDependenciesSQL

Add ColumnModify ColumnRename ColumnDrop ColumnRenameCopyDropTruncateCreate Lookup Table

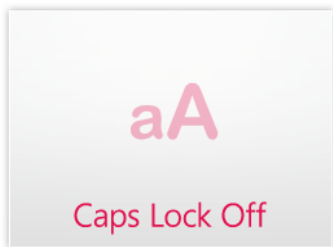
Column Name	Data Type	Nullable	Default	Primary Key
EAN	NUMBER(13,0)	No	-	1
NOMBRE	VARCHAR2(15)	No	-	-
FECHA_LANZAMIENTO	DATE	Yes	-	-
PVP	VARCHAR2(3)	Yes	-	-
DURACION	NUMBER(7,2)	Yes	-	-
ID_FILIAL	NUMBER(2,0)	Yes	-	-
CIF	VARCHAR2(9)	Yes	-	-

1 - 7

Download

aA

Caps Lock Off



SQL y tablas de Contiene y Pegi.

```

CREATE TABLE Contiene (
ean NUMBER(13),
cod_pegí VARCHAR2(7),
CONSTRAINT contiene_pk PRIMARY KEY (ean, cod_pegí)
CONSTRAINT contiene_fk1 FOREIGN KEY (ean) REFERENCES Videojuegos (ean) ON DELETE CASCADE,
CONSTRAINT contiene_fk2 FOREIGN KEY (cod_pegí) REFERENCES Pegí (cod_pegí) ON DELETE SET NULL
);

CREATE TABLE Pegí (
cod_pegí VARCHAR2(7),
nombre VARCHAR2(15) CONSTRAINT nombre_nnóll NOT NULL,
descripción VARCHAR2(100) CONSTRAINT descripción_nnóll NOT NULL,
CONSTRAINT pegí_pk PRIMARY KEY (cod_pegí)
);|

```

Browser

CONTIENE				
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL				
Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table				
Column Name	Data Type	Nullable	Default	Primary Key
EAN	NUMBER(13,0)	No	-	1
COD_PEGI	VARCHAR2(2)	No	-	2
				1 - 2
Download				



PEGI				
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL				
Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table				
Column Name	Data Type	Nullable	Default	Primary Key
COD_PEGI	VARCHAR2(2)	No	-	1
NOMBRE	VARCHAR2(15)	No	-	-
DESCRIPCION	VARCHAR2(100)	No	-	-
				1 - 3
Download				

