

Práctica 3

Bases de Datos



Creación y Carga Inicial de la Base de Datos 1-SPOOL

Crear tabla departamento

```
SQL> CREATE TABLE DEPARTAMENTO
```

```
2 (CD NUMBER(2) PRIMARY KEY,
```

```
3 D VARCHAR2(60));
```

Table created.

Crear tabla área

```
SQL> CREATE TABLE AREA
```

```
2 (CAR NUMBER(3) PRIMARY KEY,
```

```
3 AR VARCHAR(60),
```

```
4 CD NUMBER(2) REFERENCES DEPARTAMENTO ON DELETE CASCADE);
```

Table created.

Crear tabla profesor

```
SQL> CREATE TABLE PROFESOR
```

```
2 (DNI NUMBER(8) PRIMARY KEY,
```

```
3 P VARCHAR(60),
```

```
4 CAR NUMBER(3),
```

```
5 CAT VARCHAR2(5),
```

```
6 FOREIGN KEY (CAR) REFERENCES AREA ON DELETE SET NULL);
```

Table created.

Crear tabla asignatura

```
SQL> CREATE TABLE ASIGNATURA
```

```
2 (CAS NUMBER(3) PRIMARY KEY,
```

```
3 A VARCHAR2(50) NOT NULL,
```

```
4 T CHAR(4) NOT NULL,
```

```
5 CUR NUMBER(1) CHECK (CUR BETWEEN 1 AND 5),
```

```
6 CAR NUMBER(3) REFERENCES AREA ON DELETE SET NULL,
```

```
7 CT NUMBER(3,1) DEFAULT 0.0,
```

```
8 CP NUMBER(3,1) DEFAULT 0.0,
```

```
9 CL NUMBER(3,1) DEFAULT 0.0);
```

Table created.

Crear tabla plan docente

```
SQL> CREATE TABLE PLAN_DOCENTE
```

```
2 (DNI NUMBER(8),
```

```
3 CAS NUMBER(3),
```

```
4 CTA NUMBER(3,1) DEFAULT 0.0,
```

```
5 CPA NUMBER(3,1) DEFAULT 0.0,
```

```
6 CLA NUMBER(3,1) DEFAULT 0.0,
```

```
7 FI DATE DEFAULT SYSDATE,
```

```
8 FF DATE,
```

```
9 PRIMARY KEY(DNI, CAS, FI),
```

```
10 FOREIGN KEY (CAS) REFERENCES ASIGNATURA ON DELETE CASCADE,
```

```
11 FOREIGN KEY (DNI) REFERENCES PROFESOR ON DELETE CASCADE,
```

```
12 CHECK (FF >= FI));
```

Table created.

Razonar el motivo de las definiciones hechas. Poner especial atención en las claves primarias, las claves ajenas, las acciones compensatorias de integridad, las restricciones de dominios.

```
CREATE TABLE DEPARTAMENTO
(CD NUMBER(2) PRIMARY KEY,
D VARCHAR2(60));
```

-En la tabla **DEPARTAMENTO**, cuya clave primaria es atributo **CD**, consiguiendo que cada departamento(**D**) tenga un código de departamento único. Distinguiendo entre departamentos con un código de identificación único, si no fuera así, no se podría distinguir con un CD a qué departamento se hace referencia.

```
CREATE TABLE AREA
(CAR NUMBER(3) PRIMARY KEY,
AR VARCHAR2(60),
CD NUMBER(2) REFERENCES DEPARTAMENTO ON DELETE CASCADE);
```

-Tabla **AREA**, observamos que el atributo **CAR** es la clave primaria, relacionando cada área con un número. Además, el atributo **CD** está definido con una condición de integridad de borrado en cascada, si se elimina algún departamento también se borrarán las áreas que pertenecieran al mismo. Generaría inconsistencia la posibilidad de un área que no pertenezca a ningún departamento.

```
CREATE TABLE PROFESOR
(DNI NUMBER(8) PRIMARY KEY,
P VARCHAR2(60),
CAR NUMBER(3),
CAT VARCHAR2(5),
FOREIGN KEY (CAR) REFERENCES AREA ON DELETE SET NULL);
```

-Tabla **PROFESOR** la clave primaria es el atributo **DNI** ya que hay un por persona en el mundo. El atributo **CAR** si queremos insertar un valor en ese campo debe existir en la tabla **AREA** en caso de que no exista no dejará introducir el valor ya que se trata de una clave ajena además tiene una condición de integridad SET NULL, en caso de que se eliminara una área los profesores adscritos a la misma no se eliminarían, en cambio se pondría su valor como null, ya que puede darse el caso de que se le traslade a otra área.

```
CREATE TABLE ASIGNATURA
(CAS NUMBER(3) PRIMARY KEY,
A VARCHAR2(50) NOT NULL,
T CHAR(4) NOT NULL,
CUR NUMBER(1) CHECK (CUR BETWEEN 1 AND 5),
CAR NUMBER(3) REFERENCES AREA ON DELETE SET NULL,
CT NUMBER(3,1) DEFAULT 0.0,
CP NUMBER(3,1) DEFAULT 0.0,
CL NUMBER(3,1) DEFAULT 0.0);
```

-Tabla **ASIGNATURA**, la clave primaria es **CAS**, cada asignatura tendrá un código de área único, Valores no nulos sobre los atributos **A y T**, En caso contrario, podría haber códigos de asignatura que hicieran referencia a asignaturas inexistentes, o asignaturas que no se impartieran en ninguna titulación, e incluso **CAS** que hicieran

referencia a asignaturas sin nombre que se imparten en una titulación. El atributo **CUR** tiene condición check, con un rango especificado entre 1 y 5, pues si no fuera así, podría darse el caso de establecer asignaturas en cursos inexistentes en la titulación a la que se imparte. El atributo **CAR**, tiene definido la condición de integridad de valor nulo, ya que, puede darse el caso de que al eliminar el área a la que pertenece la asignatura, se le asigne otra área. Los atributos **CT, CP Y CL** estarán puesto de fábrica a 0.0 para así evitar confusiones, además en caso de ser entero no va a ocupar el número más de un campo y en caso de decimal no más de 3.

```
CREATE TABLE PLAN_DOCENTE
(DNI NUMBER(8),
CAS NUMBER(3),
CTA NUMBER(3,1) DEFAULT 0.0,
CPA NUMBER(3,1) DEFAULT 0.0,
CLA NUMBER(3,1) DEFAULT 0.0,
FI DATE DEFAULT SYSDATE,
FF DATE,
PRIMARY KEY (DNI, CAS, FI),
FOREIGN KEY (CAS) REFERENCES ASIGNATURA ON DELETE CASCADE,
FOREIGN KEY (DNI) REFERENCES PROFESOR ON DELETE CASCADE,
CHECK (FF >= FI));
```

-Tabla **PLAN_DOCENTE**, las claves primarias son **DNI, CAS Y FI**, de los cuales, DNI y CAS se han definido como en las tablas anteriores además FI, se ha definido como clave primaria, para evitar así que un mismo profesor tenga la misma fecha de inicio en más de una asignatura. La definición como valor por defecto 0.0 en los atributos **CTA, CPA y CLA**, es porque así están definidos en la tabla ASIGNATURA. Se ha establecido como valor por defecto la fecha del sistema para el atributo FI, para que así coincida con la fecha en la que se insertó en la tabla el plan docente al que se encuentra asociado, en caso de no especificar una fecha distinta en la inserción.

Como claves ajenas con borrado en cascada tenemos los atributos **CAS Y DNI**, de tal forma que, si se elimina una asignatura de la tabla ASIGNATURA o un profesor de la tabla PROFESOR, se elimine también el plan docente vinculado a cualquiera de estos dos atributos, pues no tiene sentido que se mantenga el plan docente de una asignatura que ya no existe o si el profesor que la imparte ya no se encuentra en la universidad. Por último en los atributos check tenemos que la fecha de finalización(**FF**) tiene que ser posterior a la fecha de inicio(**FI**) ya que no puede darse el caso en el que termines de impartir una asignatura antes de empezar a impartir la misma.

Insertar en tabla departamento

```
SQL> INSERT INTO DEPARTAMENTO
2 VALUES (1,'ANALISIS MATEMATICO');
1 row created.
```

```
SQL> INSERT INTO DEPARTAMENTO
2 VALUES (2,'ASTROFISICA');
```

1 row created.

```
SQL> INSERT INTO DEPARTAMENTO
  2 VALUES (3,'ESTADISTICA, INVESTIGACION OPERATIVA Y COMPUTACION');
1 row created.
```

```
SQL> INSERT INTO DEPARTAMENTO
  2 VALUES (4,'MATEMATICA FUNDAMENTAL');
1 row created.
```

Insertar en tabla área

```
SQL> INSERT INTO AREA
  2 VALUES (1,'ALGEBRA', 4);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (2,'ANALISIS MATEMATICO', 1);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (3,'ASTRONOMIA Y ASTROFISICA', 2);
1 row created
```

```
SQL> INSERT INTO AREA
  2 VALUES (4,'CIENCIAS DE LA COMPUTACION E INTELIGENCIA ARTIFICIAL', 3);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (5,'DIDACTICA DE LA MATEMATICA', 1);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (6,'ESTADISTICA E INVESTIGACION OPERATIVA', 3);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (7,'LENGUAJE Y SISTEMAS INFORMATICOS', 3);
1 row created.
```

```
SQL> INSERT INTO AREA
  2 VALUES (8,'MATEMATICA APLICADA', 1);
1 row created.
```

Insertar en tabla profesor

```
SQL> INSERT INTO PROFESOR
  2 VALUES(1111, 'JUAN', 6, 'CU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR
  2 VALUES(2222, 'CARLOS', 7, 'TU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(3333, 'PEDRO', 4, 'TEU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(4444, 'MARIA', 7, 'TU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(5555, 'IVAN', 1, 'CEU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(6666, 'CARMEN', 3, 'CD');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(7777, 'MARIO', 2, 'TU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(8888, 'FRANCISCO', 5, 'TU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(9999, 'ANGELA', 8, 'TEU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(1010, 'DAVID', 4, 'TU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR  
2 VALUES(2020, 'SOLEDAD', 7, 'CU');
```

1 row created.

```
SQL> INSERT INTO PROFESOR
  2 VALUES(3030, 'JOSE MANUEL', 6, 'TEU');
```

1 row created.

Insertar en tabla asignatura

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(1, 'BASE DE DATOS','GII',3,7,3,1.5,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(2, 'INTELIGENCIA ARTIFICIAL','GII',3,4,1.5,1.5,3);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(3, 'ALMACENES DE DATOS','MII',1,7,1.5,0,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(4, 'MINERIA DE DATOS','MII',1,7,1.5,0,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(5, 'INFORMATICA BASICA','GII',1,7,3,1.5,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(6, 'ALGEBRA','GII',1,1,3,3,0);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(7, 'CALCULO','GII',1,8,3,3,0);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
  2 VALUES(8, 'OPTIMIZACION','GII',1,6,3,1.5,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
```

```
2 VALUES(9, 'GESTION DE RIESGOS','GII',3,4,3,0,3);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
```

```
2 VALUES(10, 'ASTRONOMIA','GF',2,3,3,1.5,1.5);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
```

```
2 VALUES(11, 'DIDACTICA DE LA MATEMATICA','GM',2,5,6,0,0);
```

1 row created.

```
SQL> INSERT INTO ASIGNATURA
```

```
2 VALUES(12, 'ANALISIS COMPLEJO','GM',4,2,4.5,3,0);
```

1 row created.

Insertar en tabla plan docente

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(4444, 1,3,1.5,1.5, '01-SEP-11',NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(4444, 4,1.5,0,1.5, '01-SEP-08', '31-AUG-10');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(4444, 5,3,0,0, '01-SEP-10', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(1111, 8,3,1.5,1.5, '01-SEP-07', '31-AUG-09');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(1111, 8,3,0,0, '01-SEP-09', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(3030, 8,0,1.5,1.5, '01-SEP-09', NULL);
```


1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(2222, 4,1.5,0,1.5, '01-SEP-09', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(2222, 3,1.5,0,1.5, '01-SEP-06', '31-AUG-07');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(1010, 2,1.5,1.5,3, '01-SEP-05', '31-AUG-08');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(3333, 2,1.5,1.5,3, '01-SEP-08', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(1010, 9,3,0,3, '01-SEP-08', '31-AUG-09');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(1010, 9,1.5,0,1.5, '01-SEP-09', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(9999, 7,3,3,0, '01-SEP-10', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(5555, 6,3,3,0, '31-MAR-10', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE  
2 VALUES(6666, 10,3,1.5,1.5, '01-SEP-08', '31-AUG-11');
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(8888, 11,6,0,0, '01-SEP-09', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(2020, 3,1.5,0,1.5, '01-SEP-08', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(7777, 12,4.5,3,0, '01-SEP-10', NULL);
```

1 row created.

```
SQL> INSERT INTO PLAN_DOCENTE
```

```
2 VALUES(3333, 9,1.5,0,1.5, '01-SEP-09', NULL);
```

1 row created.