

Sandra Ruiz Jiménez
Bases de datos

Ejercicio 1.



1.

```
CREATE TABLE ALUMNOS (  
    MATRICULA    INTEGER(4) NOT NULL PRIMARY KEY,  
    NOMBRE       VARCHAR2(20) NOT NULL  
);
```

```
CREATE TABLE ASIGNATURAS (  
    CODIGO       VARCHAR2(3) NOT NULL PRIMARY KEY,  
    TITULO       VARCHAR2(20) NOT NULL  
);
```

```
CREATE TABLE MATRICULAN (  
    MATALUMNO    INTEGER(4) NOT NULL REFERENCES ALUMNOS ON  
DELETE CASCADE,  
    CODASIGNATURA VARCHAR2(3) NOT NULL REFERENCES ASIGNATURAS ON  
DELETE CASCADE,  
    PRIMARY KEY (MATALUMNO, CODASIGNATURA)  
);
```

2. EVA ya está matriculada en BBDD, ya que su MATRICULA (PK de tabla ALUMNOS) es 5 y la asignatura BBDD tiene un CODIGO (PK) A1. En la tabla matriculan tenemos que MATALUMNO 5 está relacionado con CODASIGNATURA A1.



3.

```
UPDATE ASIGNATURAS  
    SET TITULO='Bases de datos'  
WHERE  
    TITULO='BBDD';
```



4.

```
DELETE FROM ASIGNATURAS  
WHERE  
    TITULO='Inteligencia Artificial' AND CODIGO='A4';
```



Ejercicio 2.

1.

```
CREATE TABLE MEDICO (  
    COD_MEDICO          INTEGER(3) NOT NULL PRIMARY KEY,  
    NOMBRE              VARCHAR2(30) NOT NULL,  
    ESPECIALIDAD        VARCHAR2(20) DEFAULT 'FAM',  
    COD_DIRECTOR        INTEGER(3) NULL REFERENCES MEDICO  
);
```



```
CREATE TABLE ENFERMO (  
    COD_INSCRIPCION     INTEGER(4) NOT NULL PRIMARY KEY,  
    NSS                VARCHAR2(10),  
    NOMBRE             VARCHAR2(30),  
    SEXO               VARCHAR2(10)  
);
```

```
CREATE TABLE HABITACION (  
    NUM_HAB            INTEGER(3) CHECK (NUM_HAB BETWEEN 100 AND  
399) NOT NULL PRIMARY KEY,  
    NUMERO_CAMAS       INTEGER(1) CHECK (NUMERO_CAMAS BETWEEN 1  
AND 3)  
);
```

```
CREATE TABLE INGRESO (  
    COD_INGRESO        INTEGER(10) NOT NULL PRIMARY KEY,  
    FECHA_INGRESO      DATE NOT NULL DEFAULT SYSDATE,  
    FECHA_ALTA         DATE CHECK (FECHA_ALTA > FECHA_INGRESO),  
    COD_INSCRIPCION     INTEGER(4) NOT NULL REFERENCES ENFERMO ON  
DELETE CASCADE,  
    NUM_HAB            INTEGER(3) CHECK (NUM_HAB BETWEEN 100 AND  
399) REFERENCES HABITACION  
);
```

```
CREATE TABLE VISITA (  
    COD_MEDICO          INTEGER(3) NOT NULL REFERENCES MEDICO ON  
DELETE CASCADE,  
    COD_INSCRIPCION     INTEGER(4) NOT NULL REFERENCES ENFERMO ON  
DELETE CASCADE,  
    FECHA              DATE,  
    DIAGNOSTICO         VARCHAR2(60),  
    PRIMARY KEY (COD_MEDICO, COD_INSCRIPCION)  
);
```

2.

I.

```
ALTER TABLE ENFERMO MODIFY SEXO CHAR(1) CHECK (SEXO IN 'H', 'M');
```

J. Porque la primary key o alguna de las FK de la tabla VISITA se han definido como UNIQUE, por lo tanto su valor no puede estar repetido.

Buscaríamos el valor de la restricción y suponemos que se llama VISITA_UP.

La eliminamos de la siguiente manera:



```
ALTER TABLE VISITA DROP CONSTRAINT VISITA_UQ;
```

K.

```
ALTER TABLE MEDICO MODIFY (ESPECIALIDAD VARCHAR2(3)  
ESPECIALIDAD=UPPER (ESPECIALIDAD));
```

L.

```
INSERT INTO MEDICO  
    (COD_MEDICO, NOMBRE, ESPECIALIDAD, COD_DIRECTOR)  
VALUES  
    (1, 'Manuel González Felipe', 'FOR', 4);
```

INSERT INTO ENFERMO

```
    (COD_INSCRIPCION, NSS, NOMBRE, SEXO)  
VALUES  
    (1, '2549612481', 'Juan Carlos Calero Sánchez', 'H');
```

INSERT INTO ENFERMO

```
    (COD_INSCRIPCION, NSS, NOMBRE, SEXO)  
VALUES  
    (2, '2365148541', 'Sergio Jiménez Turnes', 'H');
```

M.

INSERT INTO HABITACION

```
    (NUM_HAB, NUMERO_CAMAS)  
VALUES  
    (102, 2);
```

INSERT INTO INGRESO

```
    (COD_INGRESO, FECHA_INGRESO, FECHA_ALTA, COD_INSCRIPCION,  
    NUM_HAB)  
VALUES  
    (125716, SYSDATE, , 2, 102);
```

N.

INSERT INTO VISITA

```
    (COD_MEDICO, COD_INSCRIPCION, FECHA, DIAGNOSTICO)  
VALUES  
    (1, 1, SYSDATE, 'Tontitis aguda');
```

O.

```
ALTER TABLE VISITA ADD (FECHA_PROXIMA_VISITA DATE);
```

P.

```
UPDATE ENFERMO  
    SET NOMBRE='PEPE PEREZ', SEXO='H'  
WHERE  
    COD_INSCRIPCION=1;
```

Q.
DELETE FROM ENFERMO
WHERE
 COD_INSCRIPCION=1;

R.
DROP TABLE VISITA;
DROP TABLE INGRESO;
DROP TABLE HABITACION;
DROP TABLE ENFERMO;
DROP TABLE MEDICO;

Ejercicio 3.

#Una organización puede ser de 1 tipo (1,1) y un tipo puede estar presente en 0 o varias organizaciones (0,N). La cardinalidad queda 1:N

#Relacion 1:N:M entre tipo-organización-pais

```
CREATE TABLE TIPO (  
    TIPO TEXT NOT NULL PRIMARY KEY  
);  
  
CREATE TABLE ORGANIZACION (  
    NOMBRE TEXT NOT NULL PRIMARY KEY  
);  
  
CREATE TABLE AVALAR (  
    NOMBRE TEXT NOT NULL REFERENCES ORGANIZACION,  
    NOMBRE_AVALADA TEXT NOT NULL REFERENCES ORGANIZACION,  
    PRIMARY KEY (NOMBRE, NOMBRE_AVALADA)  
);  
  
CREATE TABLE PAIS (  
    NOMBRE TEXT NOT NULL PRIMARY KEY,  
);  
  
CREATE TABLE AGRUPAR (  
    TIPO TEXT NOT NULL REFERENCES TIPO,  
    NOMBRE TEXT NOT NULL REFERENCES ORGANIZACION,  
    NOMBRE TEXT NOT NULL REFERENCES PAIS,  
    PRIMARY KEY (TIPO, NOMBRE, NOMBRE)  
);  
  
CREATE TABLE ACCIDENTE (  
    CODIGO_ACCIDENTE SMALLINT NOT NULL ,  
    NOMBRE TEXT,  
    NOMBRE TEXT NOT NULL REFERENCES PAIS  
);  
  
CREATE TABLE LAGO (  
    CODIGO SMALLINT NOT NULL,  
    VOLUMEN TEXT,  
    CODIGO_ACCIDENTE SMALLINT NOT NULL REFERENCES ACCIDENTE,  
    PRIMARY KEY (CODIGO, CODIGO_ACCIDENTE)  
);  
  
CREATE TABLE MONTANA (  
    CODIGO SMALLINT NOT NULL
```

```

        ALTURA_MAX          TEXT,
        CODIGO_ACCIDENTE     SMALLINT NOT NULL REFERENCES
ACCIDENTE,
        PRIMARY KEY (CODIGO, CODIGO_ACCIDENTE)
);

CREATE TABLE RIO (
        CODIGO                SMALLINT NOT NULL,
        CODIGO_ACCIDENTE     SMALLINT NOT NULL REFERENCES
ACCIDENTE,
        PRIMARY KEY (CODIGO, CODIGO_ACCIDENTE)
);

CREATE TABLE LOCALIDAD (
        CODIGO                SMALLINT NOT NULL PRIMARY KEY,
        NOMBRE                TEXT
);

CREATE TABLE LOCALIDAD-RIO (
        CODIGO                SMALLINT NOT NULL REFERENCES RIO,
        CODIGO_ACCIDENTE     SMALLINT NOT NULL REFERENCES
ACCIDENTE,
        CODIGO                SMALLINT NOT NULL REFERENCES
LOCALIDAD,
        KM                    TEXT,
        PRIMARY KEY (CODIGO, CODIGO_ACCIDENTE, CODIGO)
);

```

La tabla no se encuentra en 3FN porque:

- En la tabla localidad-rio, los km no depende de toda la PK, sino de una parte, por tanto no está en segunda forma normal.
- Las tablas LAGO, MONTAÑA Y RIO no necesitan una primary key propia porque heredan la de la tabla ACCIDENTE al tratarse de una jerarquía, por tanto, hay redundancia de información. No se cumple la 1FN.

Índice de comentarios

5.1 Sobre el código de lago, montaña y río