

**EXAMEN EXTRAORDINARIO DE PRÁCTICAS DE BASES DE DATOS**

**15 de junio de 2004**

**Nombre:**

**LEA ATENTAMENTE LAS PREGUNTAS Y RESUELVA CADA  
EJERCICIO COMO SE INDIQUE (EN CASO CONTRARIO EL  
EJERCICIO NO PUNTUARÁ)  
CADA EJERCICIO VALE UN PUNTO**

---

Dadas las siguientes tablas:

Personas (DNI, nombre, apellidos, dirección, ciudad, país)

Coches(matricula, marca, modelo, color, conductor, seguro)

Seguros(numero, compañía, tipo, franquicia, tomador, fecha)

Con las siguientes claves ajenas:

Coches.conductor → Personas

Coches.Seguro → Seguros

Seguros.Tomador → Personas

Y las siguientes restricciones:

Coches.Seguro es un campo obligatorio

Seguros.Tipo debe tomar por defecto el valor "Terceros"

Realizar en SQL los siguientes ejercicios:

1. Crear las tablas tal y como se haría en SQL y, una vez creadas:  
Agregar a la tabla Personas la columna sexo de un solo carácter.  
Agregar a la tabla Coches la columna potencia de tipo entero.

```
CREATE TABLE Personas (  
    DNI Integer PRIMARY KEY,  
    nombre char(30),  
    apellidos char(50),  
    direccion char(50),  
    ciudad char(30),  
    pais char(30)  
);  
  
CREATE TABLE Seguros (  
    numero Integer PRIMARY KEY,  
    compañía char(20),  
    tipo char(20) DEFAULT "Terceros",  
    franquicia number(6,2),  
    tomador Integer REFERENCES Personas  
        ON DELETE SET NULL ON UPDATE CASCADE,  
    Fecha date  
);  
  
CREATE TABLE Coches (  
    matricula char(10) PRIMARY KEY,  
    marca char(15),  
    modelo char(20),  
    color char(30),  
    conductor Integer REFERENCES Personas  
        ON DELETE SET NULL ON UPDATE CASCADE,  
    seguro Integer NOT NULL REFERENCES Seguros  
        ON DELETE SET NULL ON UPDATE CASCADE);  
  
ALTER TABLE Personas  
    ADD sexo char(1);  
  
ALTER TABLE Coches  
    ADD potencia Integer;
```

2. Dar un ejemplo de inserción en cada una de las tablas.

Modificar los seguros de tipo “Todo Riesgo” para que la franquicia sea 120 €

Borrar aquellas personas que no vivan en España.

INSERT INTO Personas

VALUES (1, 'Pepe', 'Sanchez', 'C/calleja, 5', 'Madrid', 'España', 'H');

INSERT INTO Personas

VALUES (2, 'Juan', 'Martin', 'C/otra, 2', 'Madrid', 'España', 'H');

INSERT INTO Personas

VALUES (3, 'Antonio', 'Fernandez', 'C/barco, 3', 'Toledo', 'España', 'H');

INSERT INTO Personas

VALUES (4, 'Ines', 'Perez', 'C/asa, 15', 'Burgos', 'España', 'M');

INSERT INTO Personas

VALUES (5, 'Rosa', 'Rosado', 'C/pedro I, 3', 'Avila', 'España', 'M');

INSERT INTO Personas

VALUES (6, 'John', 'Smith', 'C/Down street, 2', 'London', 'UK', 'H');

INSERT INTO Seguros

VALUES (1, 'AS S.A.', 'Terceros', 120.00, 1, '1/1/2000');

INSERT INTO Seguros

VALUES (2, 'patria', 'Todo Riesgo', 0.00, 3, '1/1/2002');

INSERT INTO Seguros

VALUES (3, 'sin problemas', 'Terceros', 180.00, 2, '1/12/2000');

INSERT INTO Seguros

VALUES (4, 'AS S.A.', 'Todo Riesgo', 60.00, 4, '1/11/2001');

INSERT INTO Seguros

VALUES (5, 'el descanso', 'Terceros', 200.00, 5, '1/12/2003');

INSERT INTO Coches

VALUES ('1111 CCC', 'SIAT', 'coupe', 'rojo', 1, 1, 100);

INSERT INTO Coches

VALUES ('2222 CCC', 'FORDE', 'atlas', 'azul', 2, 3, 120);

INSERT INTO Coches

VALUES ('3333 BCC', 'WMB', 'espac', 'verde', 3, 2, 75);

INSERT INTO Coches

```

VALUES ('1234 CDC', 'SIAT', 'tecnic', 'azul', 4, 4, 150);
INSERT INTO Coches
VALUES ('1213 CBB', 'GM', 'higg', 'plata', 1, 5, 100);
UPDATE Seguros
SET franquicia = 120.00
WHERE tipo = 'Todo Riesgo';
DELETE FROM Personas
WHERE pais <> 'España';

```

3. Seleccionar, alfabéticamente, el nombre de todas las personas que viven en ciudades cuyo nombre empieza por A (por ejemplo, Ávila, Álava, etc.).

```

SELECT nombre
FROM Personas
WHERE ciudad LIKE 'A%'
ORDER BY nombre;

```

4. Seleccionar las compañías de seguros que aseguran coches de la marca "SIAT".

```

SELECT DISTINCT Seguros.compañia
FROM Coches, Seguros
WHERE Coches.seguro = Seguros.numero
AND Coches.marca = 'SIAT';

```

5. Seleccionar las compañías de seguros que aseguran coches de la marca "SIAT" y cuyas pólizas fueron contratadas después del año 2000.

```

SELECT DISTINCT Seguros.compañia
FROM Coches, Seguros
WHERE Coches.seguro = Seguros.numero
AND Coches.marca = 'SIAT'
AND Seguros.fecha > '31/12/2000';

```

6. Devolver, para cada compañía de seguros, la cantidad de coches asegurados que tengan más de 100 CV.

```
SELECT compañía, count(coches.matricula)
FROM Coches, Seguros
WHERE Coches.seguro = Seguros.numero
      AND Coches.potencia>100
GROUP BY compañía
```

7. Seleccionar el modelo de los coches que tienen una potencia superior que cualquier coche asegurado por la compañía “AS S.A.”

```
SELECT modelo
FROM Coches
WHERE potencia > ANY (
      SELECT potencia
      FROM Coches, Seguros
      WHERE Coches.seguro = Seguros.numero
      AND Seguros.compañia='AS S.A.');
```

8. Seleccionar el nombre del conductor de los coches que tienen una potencia superior a todos los asegurados en el año 2002.

```
SELECT DISTINCT Personas.nombre
FROM Personas, Coches
WHERE Coches.conductor = Personas.DNI
      AND Coches.potencia > ALL
      ( SELECT potencia
      FROM Coches, Seguros
      WHERE Coches.seguro = Seguros.numero
      AND Seguros.fecha BETWEEN '1/1/2002' AND '31/12/2002');
```

9. Crear una vista (vehículos) que contenga los datos de las personas (nombre, apellidos y ciudad) y de los coches que conducen (matricula, marca, modelo y color). Insertar, si es posible, una fila de valores en la vista.

```
CREATE VIEW vehículos (nombre, apellidos, ciudad, matricula, marca, modelo, color)
AS (
SELECT P.nombre, P.apellidos, P.ciudad, C.matricula, C.marca, C.modelo, C.color
FROM Personas P, Coches C
WHERE C.conductor = P.DNI
);
```

10. Realizar un programa PL/SQL que, mediante un bucle, muestre las personas que viven en Francia y tienen un valor par en el campo DNI.

```
SET ServerOutput ON;  
SET VERIFY OFF;
```

```
DECLARE
```

```
    numero Personas.DNI%TYPE;  
    minimo Personas.DNI%TYPE;  
    maximo Personas.DNI%TYPE;  
    nom Personas.nombre%TYPE;  
    ape Personas.apellidos%TYPE;  
    cuenta Integer;
```

```
BEGIN
```

```
    SELECT MIN(DNI)  
    INTO minimo  
    FROM Personas;
```

```
    IF minimo % 2 = 1 THEN  
        minimo = minimo + 1  
    END IF;
```

```
    SELECT MAX(DNI)  
    INTO maximo  
    FROM Personas;
```

```
    numero = minimo;
```

```
    WHILE numero <= maximo LOOP  
        SELECT count(DNI)  
        INTO cuenta  
        FROM Personas  
        WHERE DNI = numero AND pais = 'Francia';
```

```
        IF cuenta > 0 THEN  
            SELECT nombre, apellidos  
            INTO nom, ape  
            FROM Personas  
            WHERE DNI = numero AND pais = 'Francia';  
            DBMS_OUTPUT.PUT_LINE(numero || ' ' || nom || ' ' || ape);
```

```
        ENDIF  
        numero = numero + 2;
```

```
    END LOOP;
```

```
END;
```

```
/
```