

**Ficheros y Bases de Datos**  
**Curso 2009-10**  
**Ingeniería Técnica de Informática**  
**Primer Parcial. 3-Septiembre-2010**

**Nombre:** \_\_\_\_\_

**Se debe entregar esta hoja**  
**2 horas**

- 1) (3,5 puntos) A partir de la información sobre la BD que se describe más abajo, se pide:**
- a) (1,5 puntos) El esquema entidad / relación, incluyendo atributos, claves y restricciones de cardinalidad y/o participación.**
  - b) (1 puntos) Pasar al modelo relacional optimizado.**
  - c) (0,5 puntos) Represente las reglas de integridad referencial resultantes.**
  - d) (0,5 puntos) Añadir una lista de restricciones (que no sean de integridad referencial) que no vengán reflejadas en el modelo relacional**

Se desea diseñar una base de datos para la gestión del servicio post-venta de una compañía fabricante de coches, a través de su red de concesionarios.

El fabricante tiene bastantes concesionarios repartidos por distintas ciudades. De cada concesionario se suele conocer un nº identificador de concesionario, su dirección, ciudad, teléfono y Director del mismo. Cada concesionario dispone de al menos un taller de post-venta (para reparaciones y revisiones de los coches de la marca).

Un taller pertenece siempre a uno o varios concesionarios, pero con un límite de tres concesionarios, por razones de negocio. Cada taller tiene siempre asignado un único Comercial post-venta. Del taller se registra un identificador único de taller, nº de mecánicos, volumen de facturación y su Comercial Post-Venta.

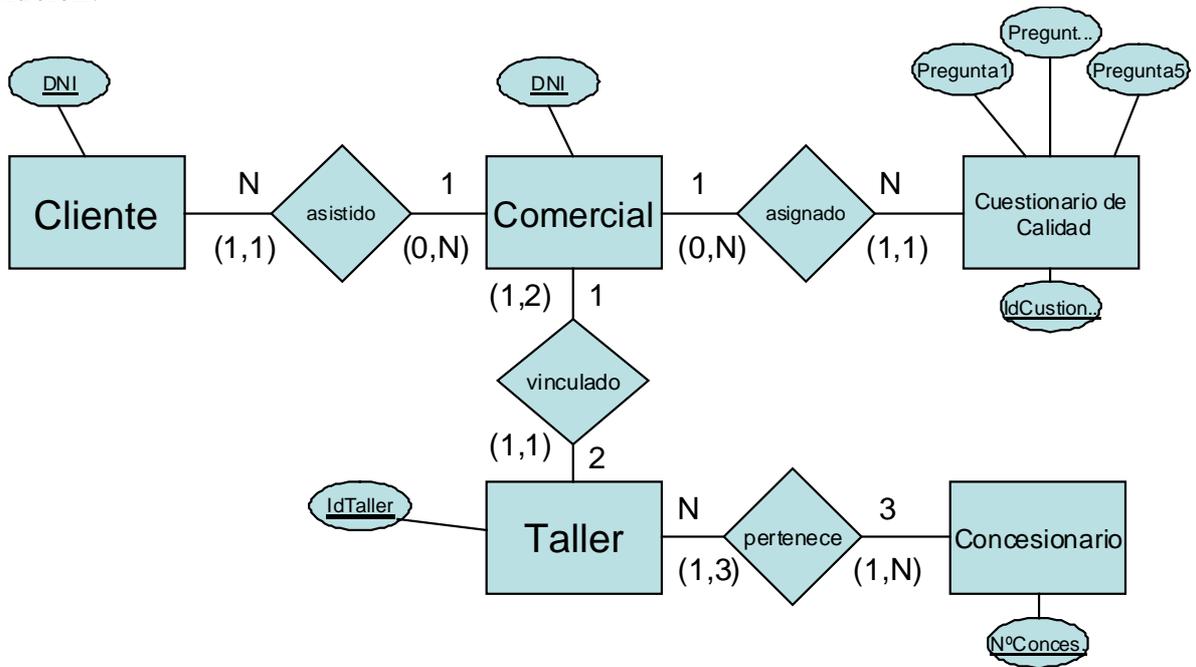
Cada Comercial post-venta tendrá asignado uno o dos talleres como máximo. También suele tener asignado un número variable de “Cuestionarios de Calidad”, aunque puede no tener ninguno. El volumen total de cuestionarios es siempre distribuido entre los comerciales disponibles.

El “Cuestionario de Calidad” es una encuesta anónima que sólo contiene 5 preguntas realizadas a los clientes sobre el servicio post-venta. Las respuestas a cada pregunta se almacenan para analizar la calidad de dicho servicio. Cada “Cuestionario de Calidad tiene un identificador único.

De cada Comercial post-venta se quiere registrar su DNI, nombre, apellidos y el contenido de todos los “Cuestionarios de Calidad” que ha realizado.

Además, cada Comercial puede asistir a un número variable de clientes. Cada cliente es siempre asistido un, y nunca por más de uno, comercial. La ficha de cada cliente contiene su DNI, nombre, apellidos, modelo de coche, antigüedad y comercial asignado.

**Solución:**



**b)**

Concesionario (N°Concesionario, dirección, ciudad, telefono, Director)

Taller\_Pertenece (N°Taller, N°Concesionario)

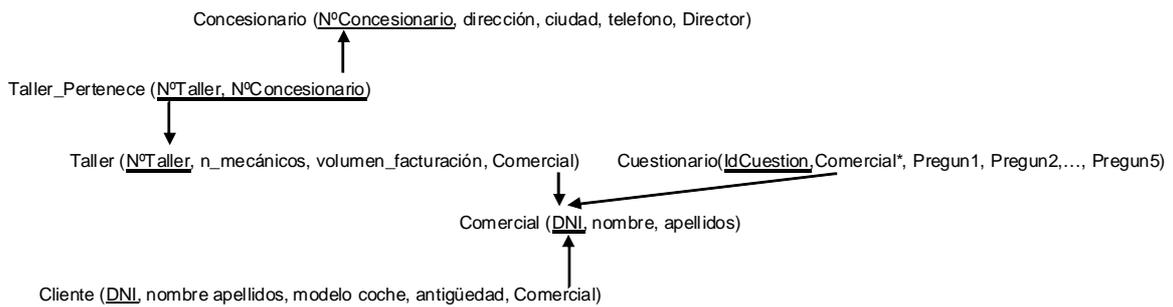
Taller (N°Taller, n\_mecánicos, volumen\_facturación, Comercial)

Cuestionario(IdCuestionario, Comercial, Pregun1, Pregun2, ..., Pregun5)

Comercial (DNI, nombre, apellidos)

Cliente (DNI, nombre apellidos, modelo coche, antigüedad, Comercial)

**c)**



**d)**

Algunas restricciones que no conserva dicho modelo relacional son:

- Participaciones totales en M a M: “Cada concesionario dispone de al menos un taller” o “Un taller pertenece siempre a uno [o varios] concesionarios”
- Participación máxima: “Un taller pertenece siempre a [...] un límite de tres concesionarios” o “Cada Comercial post-venta tendrá asignado [...] dos talleres como máximo”.

2) (4 puntos) Para organizar los proyectos en los que participa cada empleado se ha diseñado el siguiente modelo relacional:

Proyectos (numEmpleado, numProyecto, supervisor, numHoras)

Empleados(numEmpleado, nombre, apellido, salario, fechaNacimiento, ciudad)

Realizar las siguientes consultas SQL:

a) (0,75 puntos) Listar sin duplicados el nombre y apellido de los empleados que tienen un supervisor más mayor de su misma ciudad. Ordenar por nombre.

```
SELECT DISTINCT nombre, apellido
FROM Empleados E, Proyectos P, Empleados S
WHERE E.numEmpleado= P.numEmpleado
      AND supervisor = S.numEmpleado
      AND E.fechaNacimiento > S.fechaNacimiento
      AND E.ciudad = S.ciudad
ORDER BY nombre.
```

b) (0,75 puntos) Mostrar los empleados cuyo nombre empieza por 'A' que no tengan asignado ningún proyecto. Usar subconsulta correlacionada.

```
SELECT nombre, apellido
FROM Empleados E
WHERE NOT EXISTS (SELECT *
                  FROM Proyectos
                  WHERE numEmpleado = E.numEmpleado
                  )
      AND nombre LIKE 'A%'
```

c) (0,5 puntos) Mostrar los empleados cuyo nombre empieza por 'A' que no tengan asignado ningún proyecto. Usar subconsulta no correlacionada

```
SELECT nombre, apellido
FROM Empleados E
WHERE numEmpleado NOT IN (SELECT numEmpleado
                           FROM Proyectos
                           )
      AND nombre LIKE 'A%'
```

d) (0,5 puntos) Mostrar los empleados cuyo nombre empieza por 'A' que no tengan asignado ningún proyecto. No usar subconsulta.

```
SELECT nombre, apellido
FROM Empleados E LEFT OUTER JOIN Proyectos P
      ON E.numEmpleado= P.numEmpleado
WHERE nombre IS NULL
      AND E.nombre LIKE 'A%'
```

e) (0,5 puntos) Mostrar el nombre y apellido de todos los empleados con el número de proyectos que tienen asignados. Si no están en ningún proyecto debe aparecer que el número de proyectos es cero.

```
SELECT nombre, apellido, count(*) AS numProyectos
FROM Empleados E, Proyectos P
WHERE E.numEmpleado= P.numEmpleado
GROUP BY nombre, apellido

UNION ALL

SELECT nombre, apellido, 0
```

```

FROM Empleados E
WHERE NOT EXISTS (SELECT *
                  FROM Proyectos
                  WHERE numEmpleado = E.numEmpleado

```

f) (0,5 puntos) ¿Qué empleado de Madrid es supervisor de más proyectos? Indicar cuantos proyectos supervisa.

```

SELECT nombre, apellido, count(*) AS numProyectos
FROM Proyectos P, Empleados S
WHERE supervisor = S.numEmpleado
   AND CIUDAD = 'Madrid'
GROUP BY nombre, apellido
HAVING count(*) >= ALL (SELECT count(*)
                       FROM Proyectos P, Empleados S
                       WHERE supervisor = S.numEmpleado
                       AND CIUDAD = 'Madrid'
                       GROUP BY nombre, apellido
                       )

```

g) (0,5 puntos) Subir un 10% el salario de los supervisores que sumen más de 100 horas de trabajo supervisado.

```

UPDATE FROM Empleados
   SET Salario = Salario*1.1
   WHERE numEmpleado IN (SELECT supervisor
                         FROM Proyectos
                         GROUP BY supervisor
                         HAVING SUM(numHoras)>100
                         )

```

3) (2,5 puntos) Dado la relación **Proyectos** (numEmpleado, numProyecto, supervisor, numHoras) del ejercicio anterior, se añade la restricción de que un supervisor de empleados sólo puede serlo de un proyecto (un supervisor no puede supervisar en distintos proyectos).

**Responder las siguientes preguntas justificando las respuestas.**

- (0,5 punto) Listar un conjunto de dependencias funcionales minimal.
- (0,5 puntos) Calcular las claves candidatas de S
- (0,5 puntos) Determinar si la relación Proyectos está en 3FN.
- (0,5 puntos) Determinar si la relación Proyectos está en FNBC.
- (0,5 puntos) La descomposición en  $R_1(\underline{\text{numProyecto}}, \underline{\text{supervisor}})$  y  $R_2(\text{numEmpleado}, \text{supervisor}, \text{numHoras})$ , ¿preserva la propiedad de reunión no aditiva?. ¿preserva las dependencias funcionales?

**Solución:**

a)  $S = \{ \text{numEmpleado}, \text{numProyecto} \rightarrow \text{supervisor}, \text{numEmpleado}, \text{numProyecto} \rightarrow \text{numHoras}, \text{supervisor} \rightarrow \text{numProyecto} \}$

b) Claves candidatas de Proyectos:

{ numEmpleado, numProyecto }

{ numEmpleado, supervisor }

c) Según la definición, una relación R está en normalizada en 3FN si:

Para cualquier dependencia funcional  $X \rightarrow Y$  no trivial en  $S^+$ ,

- X es superclave en R, es decir  $X \rightarrow R$ , o bien
- Y forma parte de una clave candidata en R.

{numEmpleado, numProyecto} es superclave  
{supervisor} no es superclave, pero {numProyecto} forma parte de una clave candidata  
Luego la relación Proyectos está en 3FN

d) Según la definición, una relación R está en normalizada en FNBC si:

Para cualquier dependencia funcional  $X \rightarrow Y$  no trivial en  $S^+$ ,

X es superclave en R, es decir  $X \rightarrow R$ .

{supervisor}, NO es superclave en R, por tanto Proyectos NO está en forma normal FNBC.

e) Para ver si esta descomposición  $R_1(\text{numProyecto}, \text{supervisor})$  y  $R_2(\text{numEmpleado}, \text{supervisor}, \text{numHoras})$  preserva la propiedad de reunión no aditiva, podemos comprobar la propiedad vista en clase:

$$((R_1 \cap R_2) \rightarrow (R_1 - R_2)) \in S^+ \quad \text{o bien}$$

$$((R_1 \cap R_2) \rightarrow (R_2 - R_1)) \in S^+$$

Si tomamos este primer caso, tenemos que:

$$((R_1 \cap R_2) \rightarrow (R_1 - R_2)) = \text{supervisor} \rightarrow \text{numProyecto} \in S^+$$

Podemos afirmar que la descomposición propuesta sí preserva la propiedad de reunión no aditiva.

Nota: Otra forma de demostrarlo es ver que la descomposición dada es la salida del algoritmo que descompone en FNBC, que preserva la propiedad de reunión no aditiva

En esta descomposición se ve que no se preserva la dependencia funcional  $\text{numEmpleado}, \text{numProyecto} \rightarrow \text{numHoras}$ , pues ni  $R_1$  ni  $R_2$  contienen todos los 3 atributos