

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30



76.522 19 01 19 EX

Espacio para la etiqueta identificativa con el código personal del **estudiante**.
Examen

Ficha técnica del examen

- Comprueba que el código y el nombre de la asignatura corresponden a la asignatura matriculada.
- Debes pegar una sola etiqueta de estudiante en el espacio correspondiente de esta hoja.
- No se puede añadir hojas adicionales, ni realizar el examen en lápiz o rotulador grueso.
- Tiempo total: **2 horas** Valor de cada pregunta: **Indicado en cada ejercicio**
- En el caso de que los estudiantes puedan consultar algún material durante el examen, ¿cuáles son?:
- En el caso de poder usar calculadora, de que tipo? **NINGUNA**
- En el caso de que haya preguntas tipo test: ¿descuentan las respuestas erróneas? **NO** ¿Cuánto?
- Indicaciones específicas para la realización de este examen

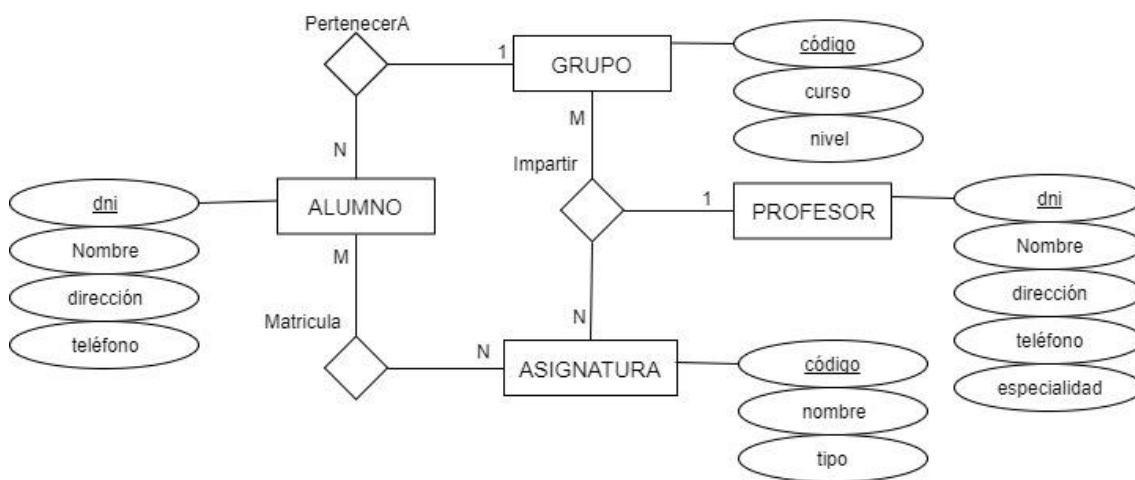
Enunciados

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

Ejercicio 1 (2 puntos)

Dado el siguiente esquema conceptual, transformadlo al modelo relacional indicando los nombres de las tablas resultantes y para cada una de las tablas, los nombres de sus atributos. Además, subrayad el/los atributo/s que forma/n la clave primaria de cada tabla, y para cada clave foránea indicad a qué tabla y atributo/s referencia.



Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

GRUPO (código, curso, nivel)

PROFESOR (dni, nombre, dirección, teléfono, especialidad)

ASIGNATURA (código, nombre, tipo)

IMPARTIR (códigoGrupo, códigoAsig, dniProfesor)

donde { códigoGrupo } referencia GRUPO(código)

{ códigoAsig } referencia ASIGNATURA(código)

{ dniProfesor } referencia PROFESOR(dni)

ALUMNO (dni, nombre, dirección, teléfono, códigoGrupo)

donde { códigoGrupo } referencia GRUPO(código)

MATRICULA (dniAlumno, códigoAsig)

donde { dniAlumno } referencia ALUMNO(dni)

{ códigoAsig } referencia ASIGNATURA(código)

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

Ejercicio 2 (2 puntos)

Diseñad un diagrama E-R que ilustre los conceptos y funcionalidades de una cadena de fruterías:

La cadena dispone de puestos de venta de fruta en los diversos mercados de una ciudad.

Cada mercado se identifica mediante un código y se requiere almacenar en la base de datos también su nombre, dirección y número de teléfono.

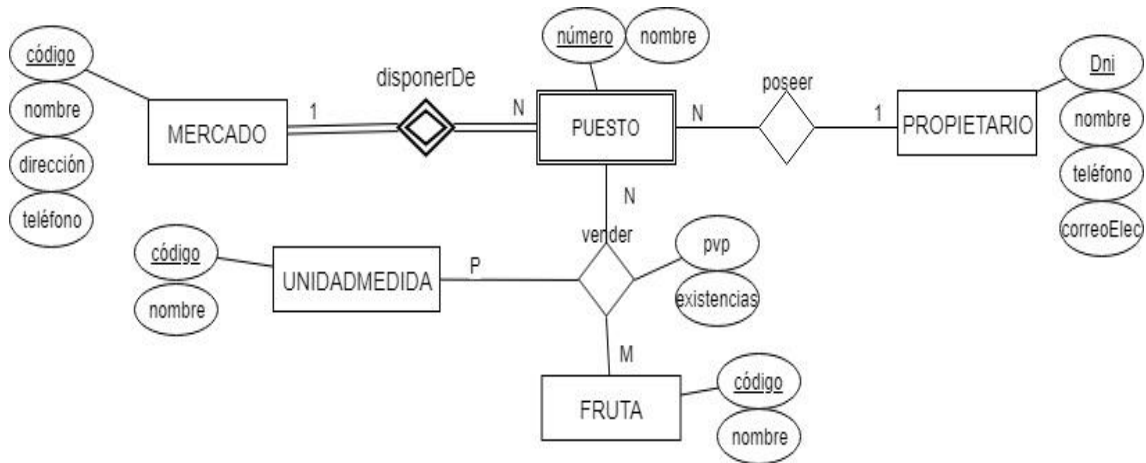
En cada mercado puede haber varios puestos de fruta. Cada puesto tendrá asignado un número, el cual se podrá repetir entre los diferentes mercados de la ciudad. Esto quiere decir que puede haber el puesto nº 1 en el mercado con el código M01, pero también en el mercado con código M02. Sobre cada puesto, además del número, se desea almacenar el nombre del puesto.

En cada puesto se venden diferentes frutas, que pueden variar de un puesto a otro. De cada fruta se desea registrar un código y nombre. La misma fruta puede ser vendida en los diferentes puestos en una o varias unidades de medida. Por ejemplo, la manzana Golden se puede vender tanto por kilogramos como por bolsas de 4 kg. Es necesario saber el número de existencias de que dispone cada puesto acerca de cada fruta y el precio de venta al público (PVP): por ejemplo, se necesitará saber que el puesto nº 1 del mercado M01 dispone de 40 kg. de manzanas Golden para su venta por kilogramos a razón de 1,10 €/kg. y de 15 bolsas de manzanas Golden de 4 kg. cada una a razón de 4,00 €/bolsa. De cada unidad de medida se requiere almacenar el código de la misma y una descripción.

Se quiere también almacenar en la base de datos la siguiente información acerca del propietario o de la propietaria de cada puesto: DNI, nombre, teléfono y correo electrónico. Se sabe que cada puesto es propiedad de una única persona, pero que una persona puede ser propietaria de varios puestos de fruta.

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30



Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

Ejercicio 3 (2 puntos)

A partir de la siguiente definición en MySQL:

Create table provincia

(n_provincia integer primary key,
nombre varchar(40) unique not null,
superficie integer not null,
id_capital integer not null);

Create table ciudad

(n_ciudad integer primary key,
nombre varchar(50) not null,
poblacion integer not null,
n_provincia integer not null,
foreign key(n_provincia) references provincia (n_provincia) on update cascade);

Alter table provincia add foreign key (id_capital) references ciudad(n_ciudad) on update cascade;

Mostrad para todas las provincias con menos población que la ciudad de Barcelona, el nombre de la provincia, su población y el número de ciudades que hay en esa provincia.

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

SELECT P.nombre, sum(poblacion) as Población, count(*) as 'Nº de ciudades'

FROM Provincia P

INNER JOIN Ciudad C

ON C.n_provincia = P.n_provincia

GROUP BY P.nombre

HAVING sum(poblacion) < (SELECT poblacion

FROM Ciudad

WHERE nombre = 'Barcelona');

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

Ejercicio 4 (2 puntos)

A partir de la base de datos del ejercicio 3, escribid el código PHP necesario para recuperar y mostrar por pantalla:

“El nombre y la población de todas las ciudades con una población comprendida entre dos valores dados que pertenezcan a una provincia de la cual conocemos su número de provincia.”

Esto quiere decir que, si se proporcionan como datos el número de provincia 39 y los valores 5000 y 15000, se deberán mostrar el nombre y la población de todas las ciudades de la provincia con el número 39 que tengan una población entre 5000 y 15000 habitantes, ambos valores incluidos.

El número de la provincia, la población mínima y la población máxima se pasarán desde un formulario con el método POST.

Nota: No será necesario crear el código de conexión a la Base de datos, se asumirá que la conexión se ha realizado con éxito y se encuentra referenciada por la variable \$con

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

```
<!DOCTYPE html>
<?php
header('Content-Type: text/html; charset="UTF-8"');
/*Recuperación del número de la provincia, de la población mínima y de la
población máxima*/
$numprov = $_POST['numprov'];
$poblamin = $_POST['poblamin'];
$poblamax = $_POST['poblamax'];

$ciudades = "SELECT nombre, poblacion
              FROM Ciudad
              WHERE poblacion between ". $poblamin . " and ". $poblamax.
              " and n_provincia = ". $numprov;
$resultado = mysqli_query($con, $ciudades);
?>
<html lang="en">
<head>
</head>
<body>
<?php>
if (mysqli_num_rows($resultado) > 0) {
    //Se muestra el resultado de la consulta
    while ($fila = mysqli_fetch_assoc($resultado)){
        echo $fila['nombre']. ' ' . $fila['poblacion'];
        echo "<br/>";
    }
}
?>
</body>
</html>
```

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

Ejercicio 5 (2 puntos)

Una empresa de formación desea crear una base de datos para almacenar información sobre los cursos que imparte y los alumnos asistentes a los mismos.

Cada curso tiene asignado un código, un nombre y un profesor/a que lo imparte. De los alumnos que asisten a los cursos se desea saber: su DNI, nombre y teléfono. Además, se sabe que un alumno tiene un número de matrícula distinto por cada curso al que asiste. También se desea saber que calificación ha obtenido cada alumno en cada curso realizado

Imaginad que se dispone de la siguiente relación universal:

Formación (CodCurso, DNIAlumno, NomCurso, Profesor, NomAlumno, TfnoAlumno, Calificación, NumMatricula)

1. Indicad si la relación está en 2FN, razonando la respuesta.
2. Si la relación no está en 2FN, normalizadla hasta la 2FN.
3. Indicad si la/s relación/es resultantes del paso 2 está/n en 3FN, razonando la respuesta.
4. Si la/s relación/es no está/n en 3FN, normalizadla/s hasta la 3FN.

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	19/01/2019	18:30

1.- No está en 2FN porque hay atributos que no dependen de la clave primaria completa. Así, NomCurso y Profesor mantienen una DF de CodCurso.

Además, NomAlumno y TfnoAlumno mantienen una DF de DNIALumno.

Los otros dos atributos no clave (Calificación y NumMatrícula) si tienen una DF plena respecto de toda la clave primaria.

2.- Como hay atributos que no dependen de toda la clave primaria (concretamente, NomCurso, Profesor, NomAlumno y TfnoAlumno), se eliminan de la tabla Formación y se llevan a nuevas tablas:

- Crearemos una nueva tabla Curso con NomCurso y Profesor más el atributo del que dependen (CodCurso), el cual será la clave primaria.

- Crearemos una nueva tabla Alumno con los atributos NomAlumno y TfnoAlumno más el atributo del que dependen (DNIALumno), el cual será la clave primaria.

En la tabla original se mantienen solo como atributos no clave Calificación y NumMatrícula, que son los únicos dos atributos que tienen una dependencia plena respecto de la clave primaria (CodCurso, DNIALumno).

Formación (CodCurso, DNIALumno, Calificación, NumMatrícula)

donde {CodCurso} referencia a Curso

{DNIALumno} referencia a Alumno

Curso (CodCurso, NomCurso, Profesor)

Alumno (DNIALumno, NomAlumno, TfnoAlumno)

3.- Las relaciones resultantes del paso 2 están en 3FN porque todos los atributos no clave dependen directamente de la clave primaria, es decir, no hay dependencias transitivas.