

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00



76.522 12 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

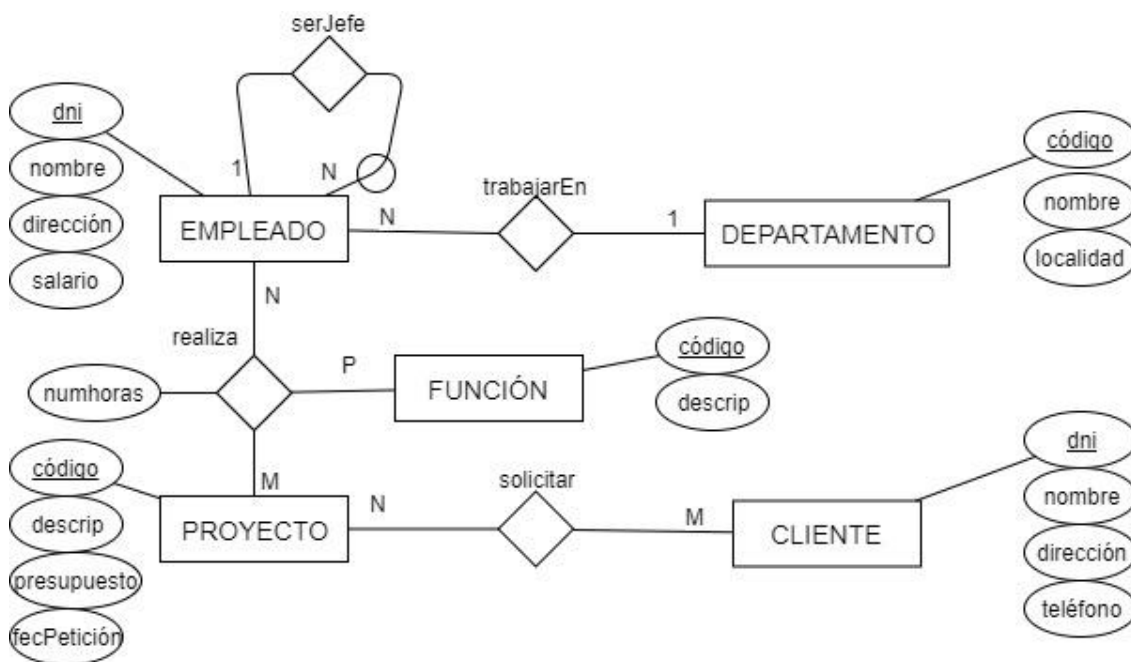
Enunciados

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

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 referencia.



Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

EMPLEADO (dni, nombre, dirección, salario, dni_jefe, código_dept)

donde {dni_jefe} referencia EMPLEADO (dni)

{ código_dept } referencia DEPARTAMENTO(código)

DEPARTAMENTO (código, nombre, localidad)

PROYECTO (código, descrip, presupuesto, fecPetición)

FUNCIÓN (código, descrip)

REALIZA (dni_emp, código_proy, código_fun, numhoras)

donde { dni_emp } referencia EMPLEADO(dni)

{ código_proy } referencia PROYECTO(código)

{código_fun } referencia FUNCION(código)

CLIENTE (dni, nombre, dirección, teléfono)

SOLICITAR (dni_cliente, código_proy)

donde {dni_cliente } referencia CLIENTE(dni)

{ código_proy } referencia PROYECTO(código)

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

Ejercicio 2 (2 puntos)

Diseñad un diagrama E-R que ilustre los conceptos y funcionalidades de una red de bancos:

De cada banco se desea almacenar un código único, dirección y población. Cada banco consta de un conjunto de sucursales, que se identifican mediante un número. Sin embargo, puede haber sucursales con el mismo número en bancos distintos pertenecientes a la red de bancos.

De cada sucursal se desea registrar en el sistema, además de su número, su dirección y población.

Es necesario conocer la sucursal en la que se contrató cada cuenta corriente. Cada cuenta dispone de un código único y también se desea almacenar la fecha en la que se abrió y su saldo.

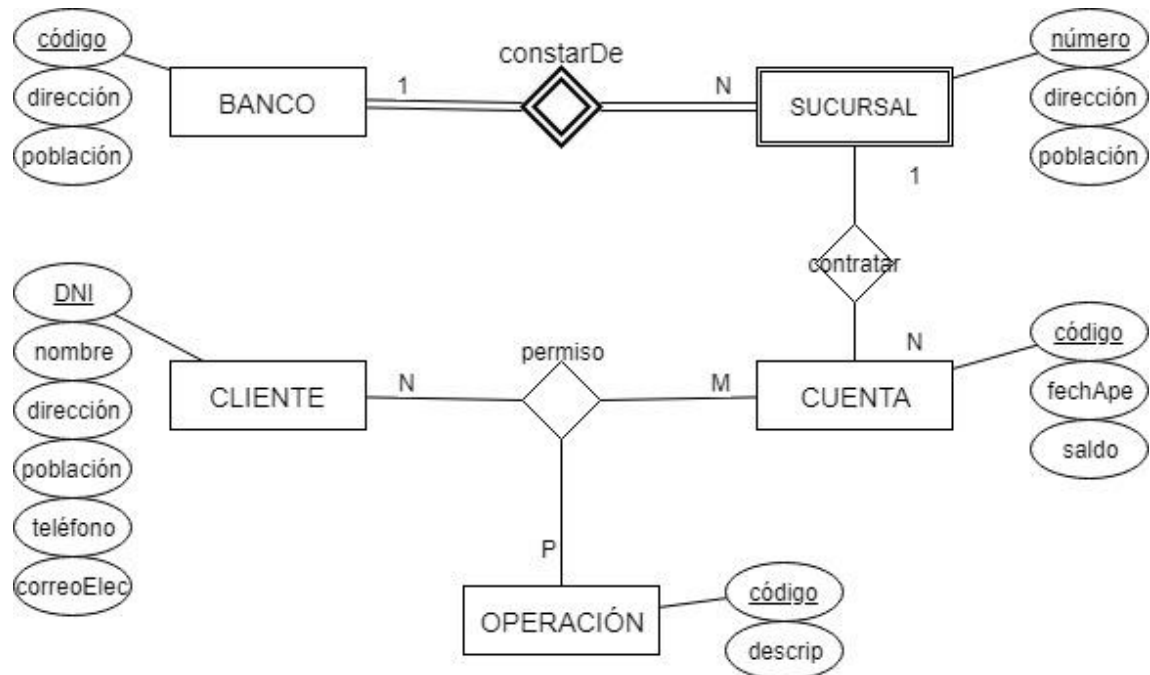
Una cuenta puede ser utilizada por varios clientes. Cada uno de los clientes que puede utilizar una cuenta podrá realizar, como mínimo, una operación sobre dicha cuenta, porque es posible que no todos los clientes puedan realizar todas las operaciones posibles sobre una cuenta. Por este motivo, es necesario conocer por cada cliente las operaciones que puede realizar sobre cada una de las cuentas a las que tiene acceso.

Cada cliente se identifica mediante su DNI y además se desea registrar su nombre, dirección, población, teléfono y correo electrónico.

Por cada operación que se puede efectuar sobre una cuenta es necesario registrar un código único y la descripción de la operación.

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00



Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

Ejercicio 3 (2 puntos)

A partir de la siguiente definición en MySQL:

```
Create table departamento  
(codigo int(3) primary key,  
nombre varchar(40) not null,  
localidad varchar(40) not null);
```

```
Create table empleado  
(codigo int(4) primary key,  
nombre varchar(20) not null,  
apellido varchar(40) not null,  
puesto varchar(20),  
fechaAlta date,  
salario float(7,2) not null,  
comision float(7,2),  
codigoDep int(3) not null,  
foreign key(codigoDep) references departamento(codigo) on update cascade on delete cascade);
```

Mostrad para todos los departamentos en los que trabajen más de 5 empleados, el nombre del departamento y la localidad en la que se encuentra, así como el número de empleados que trabajan en él y el salario y la comisión media de estos empleados. El resultado se tiene que presentar ordenado por salario medio (de mayor a menor valor).

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

**SELECT D.nombre, D.localidad, count(*) as NumEmpleados, avg(salario) as SalarioMedio,
avg(comision) as ComisionMedia**

FROM Departamento D

INNER JOIN Empleado E

ON D.codigo = E.codigoDep

GROUP BY D.nombre, D.localidad

HAVING COUNT(*) > 5

ORDER BY SalarioMedio DESC;

Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

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:

“Los nombres y apellidos de todos los empleados que trabajen en un departamento dado, conocido su código de departamento”

El código del departamento se pasará 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

```
<!DOCTYPE html>
<?php
header('Content-Type: text/html; charset="UTF-8"');
//Recuperación del código del departamento
$coddep = $_POST['coddep'];
$descripciones = "SELECT nombre, apellido
                  FROM Empleado WHERE codigoDep = ".$coddep;
$resultado = mysqli_query($con, $descripciones);
?>
<html lang="en">
<head>
</head>
<body>
<?php>
if (mysqli_num_rows($resultado) > 0) {
    //Se muestra el resultado de la consulta
    while ($fila = mysqli_fetch_assoc($resultado)){
        echo $fila['nombre'], ' ', $fila['apellido'];
        echo "<br/>";
    }
}
?>
</body>
</html>
```


Examen 2018/19-1

Asignatura	Código	Fecha	Hora inicio
Diseño de bases de datos	76.522	12/01/2019	09:00

Ejercicio 5 (2 puntos)

Una tienda desea crear una base de datos con información sobre los pedidos que reciben de los clientes. Cada pedido está identificado por un código y también se desea almacenar la fecha en que fue solicitado. Se desean registrar datos de los artículos solicitados en el pedido. Cada artículo se identifica por un código y además tiene una descripción y un precio (igual para todos los clientes). También es necesario registrar el número de unidades que de cada artículo se solicita en cada pedido.

Imaginad que se dispone de la siguiente relación universal:

Pedido (CodPedido, CodArticulo, FechaPedido, DescriArticulo, PrecioArticulo, Unidades)

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	12/01/2019	09:00

1.- No está en 2FN porque hay atributos que no dependen de la clave primaria completa. Así, FechaPedido mantiene una DF de CodPedido.

Además, DescrArticulo y PrecioArticulo mantienen una DF de CodArticulo

2.- Como hay atributos que no dependen de toda la clave primaria (concretamente, FechaPedido, DescrArticulo y PrecioArticulo), se eliminan de la tabla Pedido y se llevan a nuevas tablas:

- Crearemos una nueva tabla InfoPedido con FechaPedido más el atributo del que depende (CodPedido), el cual será la clave primaria.

- Crearemos una nueva tabla Articulo con los atributos DescrArticulo y PrecioArticulo más el atributo del que dependen (CodArticulo), el cual será la clave primaria.

En la tabla original, que la podemos renombrar como LineaPedido, se mantiene solo como atributo no clave Unidades, que es el único que tiene una dependencia plena respecto de la clave primaria (CodPedido, CodArticulo).

LineaPedido (CodPedido, CodArticulo, Unidades)

donde {CodPedido} referencia a Articulo

{CodArticulo} referencia a Articulo

InfoPedido (CodPedido, FechaPedido)

Articulo (CodArticulo, DescrArticulo, PrecioArticulo)

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.