

## 1. Modelo Entidad-Relación (2 p)

- Dibuja el diagrama Entidad-Relación que modela el minimundo que se describe a continuación.
- Debes identificar claramente las entidades fuertes y débiles (identificando también como débil el tipo de relación que se establece entre ellas), todos los atributos e identificadores de los tipos de entidad, y los atributos, cardinalidad y participación de los tipos de relación, **utilizando la notación utilizada en clase (no use la notación mínimos-máximos)**.

Queremos diseñar la base de datos de una cadena de supermercados. De cada **supermercado** almacenamos un **código único**, **nombre**, **dirección** y **teléfonos**.

↳ Multivaluado

Almacenamos información del personal del supermercado. De cada **empleado** almacenamos el **NSS**, el **nombre**, la **dirección**, la **fecha de nacimiento**, el **número de teléfono** y el **correo electrónico**. En un momento dado, en cada **supermercado** trabajan **varios empleados**; y cada **empleado** está destinado a **un solo supermercado** y ocupa un solo **puesto** (reponedor, cajero, carnicero, etc.). Pero debemos poder consultar no solo el personal actual de cada supermercado, sino también el **personal del supermercado en cualquier momento en el pasado**. Y, **para cada empleado**, debemos poder consultar el puesto que ocupó en cada uno de los periodos que trabajó. Por ejemplo, el empleado con NSS 34555333Z puede trabajar:

- Historia {
- en el supermercado Super SUP-1 del F Inicio 07/01/20 al F.Final 07/08/20 como Puesto reponedor,
  - del 08/08/20 al 01/07/22 en el supermercado SUP-99 de supervisor,
  - y que desde el 01/07/22 a la fecha vuelve a laborar en el SUP-1 como supervisor.

Como puedes ver, puede trabajar en el mismo supermercado durante varios periodos de tiempo, con el mismo trabajo o con otro diferente.

puesto

Necesitamos almacenar los productos a la venta, que son iguales en todos los supermercados. De cada **producto** almacenamos un **código único**, **nombre** y **precio de venta ACTUAL**. Aunque los productos pueden cambiar de precio incluso dentro del mismo día, **sólo almacenamos** el precio que tiene actualmente.

Cada cierto tiempo aparecen promociones (válidas para todos los supermercados) que incluyen obsequios por la compra de determinados productos. Para cada **promoción** queremos guardar su **código** (diferente para cada promoción), **nombre** y **fechas de inicio y finalización** del periodo en el que aplica. También queremos registrar los **productos** específicos **incluidos en la promoción**, y el **regalo** específico asociado a cada uno de ellos (que puede cambiar de promoción a promoción, y en una **misma promoción** puede ser **diferente para cada producto**). Por ejemplo, en la promoción "Verano 2021" (vigente del 01/07/2021 al 31/07/2021), el producto "Yogures Maldonado Pack 4" tenía asociado el regalo "Álbum de cromos", mientras que "Gaseosa espirituosa" tenía asociado el regalo "Bolsa de playa". Y en la promoción "Otoño 2021" (del 10/01/2021 al 15/10/2021), "Gaseosa espirituosa" cambió de regalo: pasó a ser "Copa de cristal"; y "Yogures Maldonado Pack 4" ya no formaba parte de la promoción.

De cada venta a un cliente guardamos información sobre la factura simplificada (o tique de compra). Cada **factura** simplificada se genera en un supermercado y tiene un **número único** (dentro de todas las facturas simplificadas de todos los supermercados de la cadena), una **fecha** y una **hora**.

La factura debe incluir la lista de productos y el número de unidades de cada producto. Observa que dependiendo de cómo coloque el cliente los productos en la cinta de la caja, puede que el cajero (persona que cobra) pase un producto al principio del proceso, pase otros productos, y cuando lleguen al final, haya más unidades del mismo producto. Es decir, dentro de la factura simplificada A-0001 pueden aparecer:

- 4 unidades de producto X-0034.
- 5 unidades del producto C-0001.
- 2 unidades del producto X-0034 (que ya aparecía en la primera línea).

En la factura simplificada deben figurar las dos líneas de producto diferenciadas X-0034. Para diferenciar líneas dentro de la misma factura simplificada, puedes usar un **número de línea** dentro de la factura simplificada, que comienza en 1 en cada factura simplificada.

De cada **línea** hay que saber el **producto**, el **número de unidades**, el **precio** del producto (**activo al momento de la creación de la factura simplificada**) y el **subtotal**, es decir, el número de unidades por el precio del producto. *Importante: Tener en cuenta que en la información del producto solo mantenemos el precio actual del producto, mientras que para cada línea de una factura necesitamos saber el precio del producto en el momento en que se generó la factura simplificada.*

Finalmente, para cada factura simplificada necesitamos saber el **importe total**.

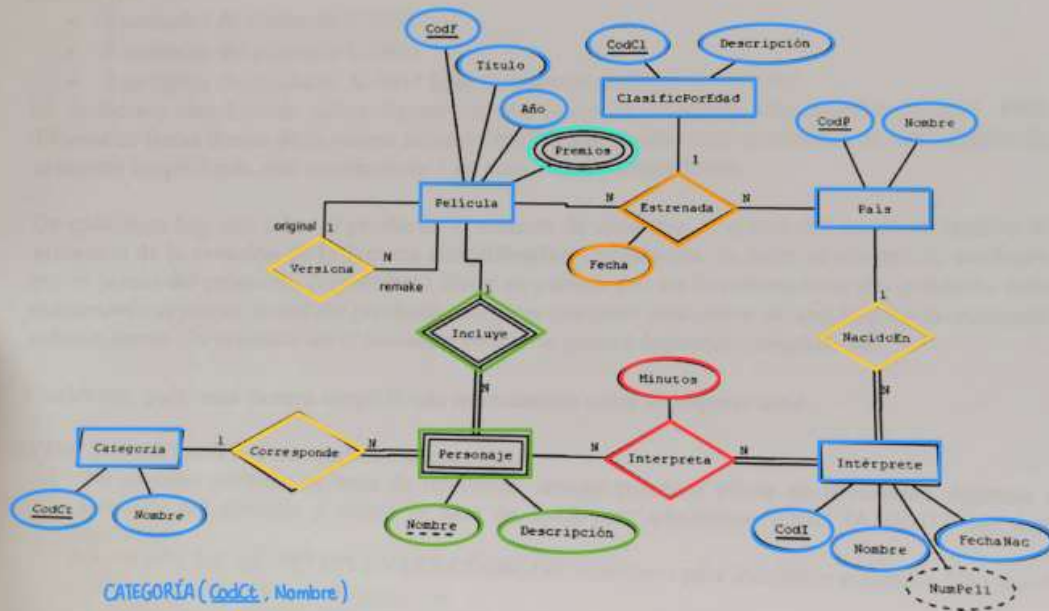
↳ Derivado

**AVISO IMPORTANTE:** Para resolver este ejercicio...

1. No inventes atributos ni tipos de relaciones, aunque parezcan existir en la realidad. Ajustese al enunciado del ejercicio y diseñe la base de datos para administrar EXACTAMENTE los datos requeridos.
2. **EN PARTICULAR, NO INVENTES CÓDIGOS** o similares para identificar entidades. Usa solo los atributos especificados en el enunciado.
3. Si consideras que algunos datos no necesitan almacenarse en la base de datos porque se puede calcular su valor, **no olvide representarlos igualmente en su modelo en forma de un atributo derivado**. El **almacenamiento de información duplicada** (de cualquier forma) resultará en una rebaja de la nota.

## 2. Paso relacional (0.75p)

Transforma el siguiente diagrama ER al esquema relacional equivalente (los esquemas de relación correspondientes al diagrama ER), indicando claramente las CLAVES PRIMARIAS y representando gráficamente (con flechas) las restricciones de integridad referencial existentes. UTILIZA LA NOTACIÓN VISTA EN CLASE.



CATEGORÍA (CodCt, Nombre)

INTÉRPRETE (CodI, Nombre, FechaNac, CodF)

PAÍS (CodP, Nombre)

CLASIFICACIÓN (CodCl, Descripción)

ESTRENADA (CodF, CodCl, CodP, Fecha)

PELÍCULA (CodF, Título, Año, Original)

PREMIOS - PELÍCULA (CodF, Premios - Película)

PERSONAJE (CodF, Nombre, Descripción, CodCt)

INTERPRETA (CodF, Nombre, CodI, Minutos)

WUOLAH



### 3. Normalización (1.25p)

#### a) Dependencias funcionales [0,4 p]

Un sitio web de información cinematográfica recopila, en su base de datos, información sobre diferentes películas (y el género cinematográfico al que corresponden). También registra información sobre el director/a (o directores/as) de dichas películas.

Considera la relación universal vinculada a la base de datos del sitio web, en la que cada tupla vincula a un director/a con una película de las dirigidas por él/ella.

**Dirigido (codP, tituloP, añoP, codG, nomG, aptoG, codD, nomD, paísD)**

Descripción detallada de los atributos:

<b>codP</b>	Código asignado a una película. <ul style="list-style-type: none"><li>Cada película recibe un código único. Dos películas no pueden tener el mismo código.</li></ul>
<b>tituloP</b>	Título de la película. <ul style="list-style-type: none"><li>Cada película recibe un solo título.</li><li>Puede haber dos películas con el mismo título.</li></ul>
<b>añoP</b>	Año de estreno de la película (cada película, un solo año de estreno).
<b>codG</b>	Código asignado al género de la película. <ul style="list-style-type: none"><li>Cada película está asociada a un único género.</li><li>Puede haber varias películas del mismo género.</li><li>Cada género recibe un código único. No puede haber dos géneros con el mismo código.</li></ul>
<b>nomG</b>	Nombre del género asignado a la película. <ul style="list-style-type: none"><li>A cada género se le da un nombre único. No puede haber dos géneros con el mismo nombre.</li></ul>
<b>aptoG</b>	Indicador asociado al género. <ul style="list-style-type: none"><li>El indicador permite especificar si el género es adecuado (o no) para público infantil.</li><li>Cada género debe estar asociado a un único valor para el indicador: "Si" o "No".</li></ul>
<b>codD</b>	Código asignado a uno de los directores de la película. <ul style="list-style-type: none"><li>Un director/a puede dirigir muchas películas.</li><li>Una película puede haber sido dirigida por varios directores/as.</li><li>Cada director/a tiene asociado un código único. Dos directores/as no pueden tener el mismo código.</li></ul>
<b>nomD</b>	Nombre completo del director/a. <ul style="list-style-type: none"><li>Puede haber dos directores/as con el mismo nombre.</li></ul>
<b>paísD</b>	País de origen del director/a. <ul style="list-style-type: none"><li>Cada director/a está asociado con un solo país.</li><li>Puede haber varios directores/as de un mismo país.</li></ul>

Indica el conjunto de dependencias funcionales completas que se producen en **Dirigido**.

**b) Forma normal y claves [0,25 p]**

Una determinada compañía telefónica almacena información en una base de datos relativa a los recibos de consumo de datos correspondientes a teléfonos móviles **con contrato en vigor**. El esquema de la relación *Recibo* es como sigue:

*Recibo* (numTelefono, fechaIniContrato, tipoTarifa, nombreTarifa, precioTarifa, nifUsu, nombreUsu, dirección, fechaRecibo, consumo, importeTotal)

Y el conjunto de dependencias funcionales completas asociadas a *Recibo* es el siguiente:

- 1) nifUsu → nombreUsu, dirección
- 2) tipoTarifa → nombreTarifa, precioTarifa
- 3) nombreTarifa → tipoTarifa, precioTarifa
- 4) numTelefono → nifUsu, nombreUsu, dirección, fechaIniContrato, tipoTarifa
- 5) numTelefono, fechaRecibo → consumo, importeTotal

Indica la/s clave/s candidata/s que existen, y la forma normal en que está *Recibo*. Puedes suponer que como mínimo, está en 1FN.

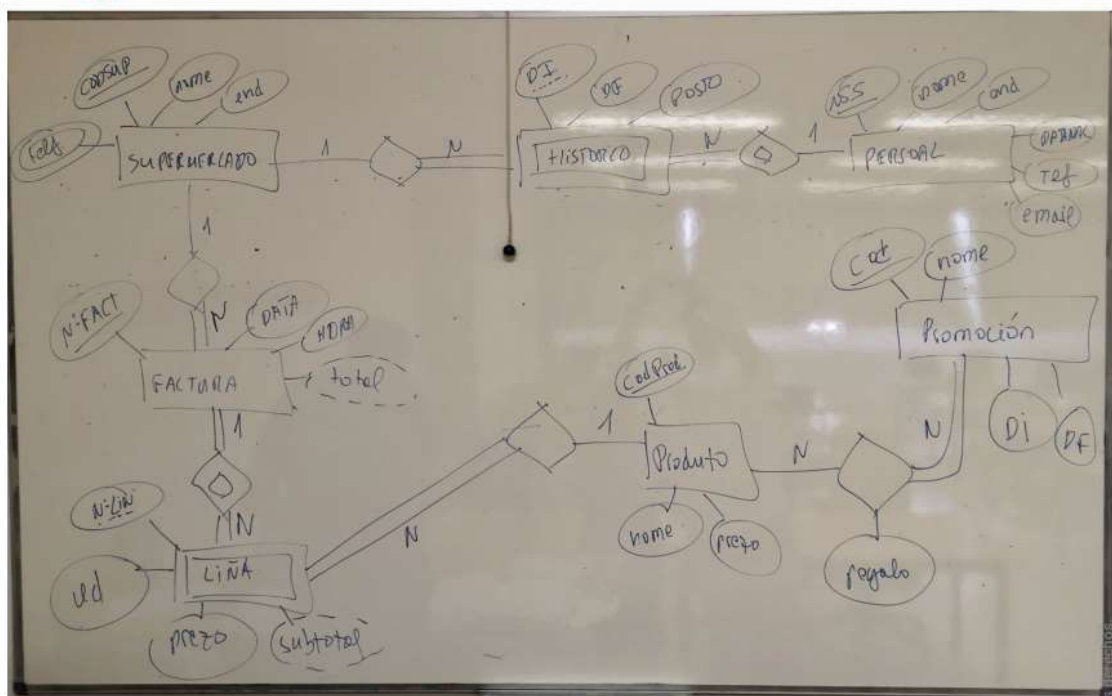
**ADVERTENCIA:** si hay varias claves candidatas (y no una sola clave candidata que consta de varios atributos), indícalo claramente. CC: A, B NO ES LO MISMO QUE CC1: A; CC2: B

**c) Descomposición [0,6 p]**

Descompón la relación *Recibo* del apartado (b) hasta encontrar una descomposición donde no se pierda información y todas las relaciones estén en FNBC. Escriba el **resultado FINAL** de la descomposición, con los **esquemas** de las relaciones, las **claves candidatas** y las **dependencias funcionales completas** de cada una de ellas.

**RECUERDA: INDICA SÓLO EL RESULTADO FINAL. NO INCLUYAS NINGUNA DE LAS RELACIONES DESCARTADAS DURANTE EL PROCESO DE DESCOMPOSICIÓN.**

1



3

a) Df

CodP → títuloP, añoP, CodG

CodG → nomG, aptoG

nomG → CodG, aptoG

CodD → nomD, paisD

b) FN y clases

(C: numtelefono, fechaRecibo)

1 a 2

c) R1 (nifusu, nomelusu, direccion)

F1 = { nifusu → nomelusu, direccion }

CC: nifusu

R2 (tipotarifa, nombretarifa, preciotarifa)

F2 = { tipotarifa → nombretarifa, preciotarifa }

nombretarifa → tipotarifa, preciotarifa

CC1: tipotarifa CC2: nombretarifa

R3 (numtelefono, nifusu, fechaRecibo, tipotarifa)

F3 = { numtelefono → nifusu, fechaRecibo, tipotarifa }

CC: numtelefono

R4 (numtelefono, fechaRecibo, consumo, importetotal)

F4 = { numtelefono, fechaRecibo → consumo, importetotal }

CC: numtelefono, fechaRecibo

WUOLAH

#### 4. Cuestionario (2p)

Todas las preguntas tienen una única respuesta correcta. Cada respuesta correcta suma 0,2. Cada respuesta incorrecta resta 0,1.

Marca las respuestas correctas en la siguiente tabla. Las respuestas marcadas en las propias preguntas no serán corregidas.

Pregunta	A	B	C
1	X		
2			X
3		X	
4	X		
5		X	

Pregunta	A	B	C
6			X
7		X	
8		X	
9			X
10			X

1. Si queremos realizar una lectura secuencial de un fichero, ¿qué es más beneficioso?

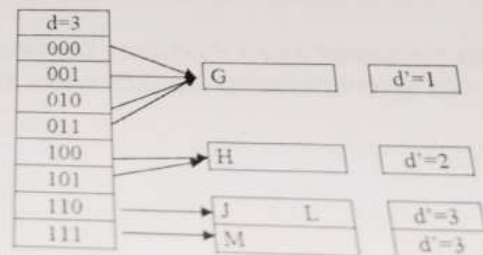
- ☒ a) Usar un tamaño de bloque físico grande.
- ☐ b) Usar un tamaño de bloque físico pequeño.
- ☐ c) El tamaño del bloque físico no es relevante puesto que se van a leer todos los registros.

2. En una estructura de bloques con slots (o estrategia slotted-page), ¿qué es cierto?

- ☐ a) Se almacenan registros de tamaño fijo que se insertan rellenando el hueco de otro registro previamente borrado.
- ☐ b) Cada bloque está compuesto por varios slots, y cada uno de ellos puede ser transferido a memoria independientemente de los demás.
- ☒ c) El número de registros que caben como máximo en cada bloque es variable, puesto que se almacenan registros de tamaño variable.

3. Hash extensible: borramos el registro L en este fichero, conservando el J. ¿Qué valor tendrán después  $d$  y  $d'$  (para el bloque que contenga a J)?

- ☐ a)  $d=3, d'=3$
- ☒ b)  $d=2, d'=2$
- ☐ c)  $d=3, d'=2$



4. En un árbol homogéneo ...

- ☒ a) Cada nodo posee dos tipos de punteros.
- ☐ b) Las búsquedas siempre llegan a nivel de nodos hoja.
- ☐ c) Las dos respuestas anteriores son correctas.

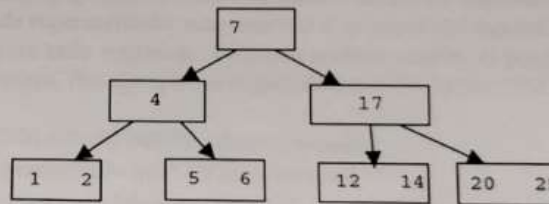


5. En la tabla PEDIDO (*código, fecha, producto, marca, sección, unidades*), que físicamente se guarda en un **fichero montículo** (*heap*), hay almacenadas miles de tuplas, pero nunca hay más de dos o tres tuplas que compartan el mismo valor del campo *fecha* (sea cual fuere) ¿Crearías un índice sobre el atributo *fecha*?

- a) No, el índice puede que no ayude mucho en las búsquedas por el atributo *fecha*, e incluso puede que las empeore.
- b) Sí, el índice ayudará en operaciones de búsqueda sobre el campo *fecha*.**
- c) El índice sería incorrecto y no sería útil para ningún tipo de búsqueda.

6. Árboles B: si insertamos el valor 30 en este árbol, con  $d=1$ , ¿qué valor/es almacenará el nodo que ahora contiene el valor 17?

- a) 17
- b) 17 y 20
- c) 17 y 25**



7. El protocolo de bloqueo *riguroso* de dos fases:

- a) Especifica que se tienen que adquirir todos los bloqueos justo en el momento de comenzar la transacción.
- b) Especifica que se tienen que liberar todos los bloqueos justo en el momento de terminar la transacción.**
- c) Las dos respuestas anteriores son correctas.

8. Transacciones:

- a) Una ejecución concurrente de dos transacciones es siempre correcta si ambas usan bloqueos de lectura/escritura para acceder a los datos que necesitan.
- b) Para que una ejecución concurrente de dos transacciones se considere correcta debe producir el mismo resultado final que alguna planificación serie de ambas transacciones.**
- c) Las dos respuestas anteriores son correctas.



9. Si en una transacción se produce el problema de la *lectura fantasma*:
- a) Quiere decir que se hace una lectura que no devuelve ningún dato, cuando realmente sí que había datos en la base de datos que deberían haber aparecido en el resultado de la consulta.
  - b) Quiere decir que se hace una consulta, y el valor devuelto de un dato es demasiado viejo (hay una versión más actualizada de ese dato).
  - c) Quiere decir que durante la transacción se hace dos veces la misma consulta, y en la segunda ocasión se recupera un número de filas diferente a la primera vez que se ejecutó la consulta (sin que la transacción haya hecho ninguna actualización que pueda haber dado lugar a ese cambio).

10. Supongamos el siguiente ejemplo de utilización de *esquema de multiversión* para control de concurrencia. ¿Qué ocurrirá si una transacción, cuya marca temporal asociada es 2, quiere escribir el dato X?

- a) El contenido de la versión Q1 será sobrescrito.
- b) Se creará una nueva versión Q2.
- c) La transacción será abortada.

Versión X	ML	ME
Q <sub>1</sub>	4	1