

Apellidos:
Nombre: UVUS:



Bases de datos (GE/DGME)
Primer parcial oficial - Parte 1

20/01/2022

Ejercicio 1 (8 puntos) Deseamos crear una herramienta para poder consultar la información relacionada con construcciones de LEGO, como se explica a continuación:

En primer lugar, la base de datos debe almacenar los colores oficiales de LEGO, con un número identificativo, nombre y código RGB. Además, se deberán conocer las categorías de piezas, con su código y nombre (por ejemplo, “1-Placa base”, “11-Ladrillo”, etc.) Deben almacenarse todas las piezas existentes, debiendo conocer para cada una su id, su tamaño (ej: 1x1, 2x8, etc.) y el año en que se empezó a producir. Una pieza pertenecerá a una sola categoría de pieza. Además, deberemos conocer para qué colores puede estar disponible cada pieza.

Se deben recopilar temas, con su código y un nombre (por ejemplo, “22-Creator”, “52-City” o “158-Star Wars”). De algunos temas pueden depender otros temas (por ejemplo, dentro de “Creator” podemos encontrar “Creator 3-en-1” y “Creator Expert”). Se almacenarán todos los conjuntos de LEGO, disponiendo cada uno de un código, un nombre y una fecha de puesta en producción. Un conjunto pertenecerá a un tema concreto. Además, se debe conocer, para cada conjunto, de qué piezas consta, en qué color y con cuántas unidades (por ejemplo, el conjunto “31069-Modular Family Villa” tiene 1 sola unidad, en este caso en negro, de la pieza cuyo id es 63695, pero para la pieza 3004 tiene 10 unidades en blanco, y 8 en gris claro azulado). Naturalmente, los colores de las piezas dentro del conjunto deben ser colores que estén disponibles para la pieza. Deberemos conocer al final, para cada conjunto, el total de piezas que incluye. Todo conjunto de LEGO tendrá un conjunto de construcciones posibles, que se deberán almacenar ordenadas para cada conjunto (la primera, la segunda, etc.) Cada construcción constará de una serie de pasos de montaje, identificados por su número. Cada paso deberá conocer cuántas unidades de cada pieza y color debe emplear, y tendrá imágenes asociadas para ver cómo montarlo. Además, las construcciones podrán ser de dos tipos: oficial o no oficial. Para las no oficiales, deberemos conocer quién es el autor.

Se pide lo siguiente:

- (5 puntos) Obtenga el esquema conceptual asociado al problema planteado, mediante diagrama Entidad-Interrelación, de forma que capture los requisitos detallados en los puntos anteriores. Indique cualquier consideración semántica que se derive de posibles interpretaciones alternativas del enunciado, y no olvide resaltar elementos redundantes en el caso de que decida incorporarlos.
- (3 puntos) Obtenga el esquema lógico asociado al diseño conceptual anterior, incluyendo debidamente las relaciones, claves primarias y ajenas, como hemos visto en clase.

Ejercicio 2 (2 puntos) Conteste razonadamente a las siguientes cuestiones:

- (1 punto) Describa brevemente los siguientes términos que hemos visto en clase: datos, información y conocimiento, y la utilidad que vienen a proporcionarnos.
- (1 punto) Comente las principales ventajas de las bases de datos, como vimos en clase.

Apellidos:

Nombre: UVUS:



Bases de datos (GE/DGME)
Primer parcial oficial - Parte 2

20/01/2022

Ejercicio 1 (3,25 puntos) Sea el siguiente esquema lógico relacional, donde la relación **S** almacena información sobre suministradores, **P** sobre piezas, **J** sobre proyectos y **SPJ** sobre suministros (es decir, provisión de piezas - en cierta cantidad - a los proyectos por parte de suministradores).

S(s, noms, estado, ciudad) **P**(p, nomp, color, peso, ciudad)

J(j, nomj, ciudad) **SPJ**(s, p, j, cant)

Se pide emplear Álgebra Relacional para responder a las operaciones de recuperación:

- Devolver el nombre y el estado de los suministradores cuya ciudad es Londres.
- Obtener el nombre de la pieza y la cantidad provista, para los suministros cuya cantidad es menor a 100 unidades.
- Nombre de las piezas que no se han proporcionado a ningún proyecto.
- Código de suministradores que han suministrado a todos los proyectos.
- Código de los proyectos a los que se haya suministrado para el mismo proyecto tanto la pieza P4 como la P5.

Ejercicio 2 (2,25 puntos) Sea el esquema de relación $R(AT, DEP)$ tal que:

$$\begin{aligned} AT &= \{A, B, E, J, D, T, F, S, R, G\}, \\ EP &= \{A \rightarrow B, E \rightarrow JDT, EB \rightarrow A, T \rightarrow D, D \rightarrow A, EF \rightarrow S, \\ &\quad EFS \rightarrow R, A \rightarrow G, G \rightarrow A\} \end{aligned}$$

Se pide lo siguiente (*los 2 primeros apartados puede ir haciéndose a la vez*):

- Obtener paso a paso el conjunto de dependencias funcionales mínimo (recubrimiento minimal), como se ha visto en clase, así como el cierre transitivo de cada campo individual.
- Obtener las claves candidatas, indicando el razonamiento seguido. Indicar quienes con los conjuntos P y Q (campos principales y no principales). Indicar si hay conjuntos de campos equivalentes.
- Determinar el grado de normalización, y en su caso normalizar hasta 3FN.

Ejercicio 3 (4,5 puntos) Consideremos la base de datos que vimos en clase:

PROVEEDORES

IDP	PNOMBRE	CATEGORIA	CIUDAD
P1	CARLOS	20	SEVILLA
P2	JUAN	10	MADRID
P3	JOSE	30	SEVILLA
P4	INMA	20	SEVILLA
P5	EVA	30	CACERES

COMPONENTES

IDC	CNOMBRE	COLOR	PESO	CIUDAD
C1	X3A	ROJO	12	SEVILLA
C2	B85	VERDE	17	MADRID
C3	C4B	AZUL	17	MALAGA
C4	C4B	ROJO	14	SEVILLA
C5	VT8	AZUL	12	MADRID
C6	C30	ROJO	19	SEVILLA

ARTICULOS

IDT	TNOMBRE	CIUDAD
T1	CLASIFICADORA	MADRID
T2	PERFORADORA	MALAGA
T3	LECTORA	CACERES
T4	CONSOLA	CACERES
T5	MEZCLADORA	SEVILLA
T6	TERMINAL	BARCELONA
T7	CINTA	SEVILLA

ENVIOS

IDP	IDT	IDC	CANT
P1	T1	C1	200
P1	T4	C1	700
P2	T1	C3	400
P2	T2	C3	200
P2	T3	C3	200
P2	T4	C3	500
P2	T5	C3	600
P2	T6	C3	400
P2	T7	C3	800
P2	T2	C5	100
P3	T1	C3	200
P3	T2	C4	500
P4	T3	C6	300
P4	T7	C6	300
P5	T2	C2	200
P5	T4	C2	100
P5	T4	C5	500
P5	T7	C5	100
P5	T2	C6	200
P5	T4	C1	100
P5	T4	C3	200
P5	T4	C4	800
P5	T5	C5	400
P5	T4	C6	500

Se pide obtener una sentencia SQL para cada una de las necesidades indicadas:

- Obtener los datos de artículos de SEVILLA y MALAGA.
- Obtener el componente de mayor peso que se pueda encontrar en SEVILLA.
- Obtener todos los nombre de proveedores que terminen en A.
- Devolver el nombre y la suma total de CANT para los artículos abastecidos por el proveedor P5.
- Mediante subconsulta, devolver los IDC para los componentes suministrados para cualquier artículo de SEVILLA.
- Seleccionar los nombres de los componentes que son suministrados en una cantidad media superior a 500.
- Crear una nueva tabla MATERIALES vinculada a COMPONENTES, con la que indicaremos qué materiales son necesarios para la fabricación de cada componente (un componente puede necesitar 1 o más materiales). Debe incluir los campos NOMBRE, CANTIDAD, CIUDAD.
- Insertar un nuevo registro en la tabla MATERIALES para el componente C3 con NOMBRE 'Coltan', 20 de CANTIDAD, CIUDAD Sevilla.

Apellidos:

Nombre: UVUS:



Bases de datos (GE/DGME)
Primer parcial oficial - Parte 3

20/01/2022

Ejercicio 1 (1 punto) Responda razonadamente a las siguientes cuestiones:

1. (0,5 puntos) Según el tema visto en clase, ¿cuándo debemos usar bases de datos NoSQL?
2. (0,5 puntos) Describa todo lo que sepa sobre **uno** de los siguientes acrónimos:
 - CAP
 - ACID
 - BASE

Ejercicio 2 (2 puntos) Haciendo uso de los principios de diseño analizados en clase, basados en el curso que estuvimos viendo, se pide plantear la estructura de colecciones que montaría para el problema planteado en el siguiente escenario (extraído y recortado del boletín sobre diseño conceptual), consistente en una herramienta informática para gestionar el transporte metropolitano de una ciudad (con información referente a líneas de metro, estaciones, sus diferentes accesos y los trenes que forman la red metropolitana).

- a) *Una línea está compuesta por una serie de estaciones en un orden determinado. Es muy importante recoger la información sobre este orden.*
- b) *Cada estación pertenece al menos a una línea, pudiendo pertenecer a varias. A su vez cada estación puede tener varios accesos, pero consideramos que un acceso sólo puede pertenecer a una única estación.*
- c) *Cada línea tiene asignados una cantidad de trenes. Un tren sólo deja de estar asignado a una línea cuando se encuentra en reparación.*
- d) *Algunas estaciones (no todas) tienen cocheras (la capacidad de almacenamiento de una cochera puede variar significativamente de una estación a otra) y cada tren tiene reservada una única cochera. Un tren puede cambiar de cochera reservada, pero no quedar sin ella. Cada cochera puede estar reservada para uno o varios trenes.*
- e) *Hay dos tipos de estaciones: simples y mixtas. Las primeras sólo tiene servicio de metro, mientras que las segundas ofrecen conexión con otros servicios (estación de tren, estación de autobuses, aeropuerto,...). Para las estaciones mixtas interesa conocer con qué otros servicios ofrece conexión.*
- f) *Cada tren tiene un código interno que lo identifica. Asimismo interesa conocer el modelo del tren, el número de vagones, la antigüedad, el color, el nombre de la empresa fabricante, la capacidad y si posee o no acceso para minusválidos (estas tres últimas propiedades sólo dependen del modelo del tren).*

Nota: los siguientes ejercicios emplean una BD MongoDB sobre alojamientos y sus *reviews*. Puede generarla mediante el archivo `listingsAndReviews.json`. Tenga en cuenta la estructura del archivo a la hora de importar con ciertos parámetros.

Ejercicio 3 (2,5 puntos) Realice las siguientes operaciones de consulta sobre la base de datos MongoDB anterior, empleando las funciones básicas de consulta (sin agregaciones):

- a) Obtener los distintos tipos de habitaciones ordenadas alfabéticamente.
- b) Consultar el número de habitaciones privadas disponibles en España.
- c) Devolver el nombre y el precio del alojamiento con menor precio cuyo campo `amenities` contenga Wifi, Kitchen y Washer.
- d) Obtener los alojamientos en Barcelona cuyo precio se encuentre en el rango `[50.0,100.0]`.
- e) Obtener los alojamientos en cuya descripción aparezca la palabra `cozy`

Ejercicio 4 (1,5 puntos) Realice las siguientes operaciones de manipulación de los datos sobre la base de datos MongoDB anterior, empleando las funciones vistas en clase para la actualización, inserción y eliminación:

- a) Añadir un nuevo campo `pays_taxes` a todos los documentos con valor por defecto `false`.
- b) Aumentar en 3 el valor `beds` del documento con `_id` 10021707.
- c) Añadir una nueva review al documento 10021707.
- d) Insertar un nuevo alojamiento utilizando únicamente los campos, `_id`, `name`, `room_type`, `price` y el campo anidado `address.market`.
- e) Eliminar el documento que acaba de insertar.

Ejercicio 5 (3 puntos) Realice las siguientes operaciones de consulta sobre la base de datos MongoDB anterior, empleando ahora agregaciones:

- a) Calcular el precio medio para los alojamientos de New York.
- b) Obtener el país con mayor número de reviews.
- c) Calcular el ratio reviews/alojamientos en España.
- d) Obtener el alojamiento de mayor precio por cada país.
- e) Utilizando el operador `unwind`, obtener el nombre del reviewer con mayor número de reviews.

Notas:

- Escriba su nombre, apellidos y UVUS en cada hoja del examen, incluido enunciado.
- Cumplimente los ejercicios en los folios proporcionados en el aula.
- Almacene el modelo de MySQLWorkbench como `parcial1_uvus.mwb`, las sentencias SQL como `parcial1_uvus.sql` y las consultas de MongoDB como `parcial1_uvus.js`, donde `uvus` debe ser su usuario virtual de la US.