

Bases de Datos

Contestar **TODAS** las preguntas en hojas separadas. Preguntas y apartados en orden. Doblar todas las hojas y entregar juntas.
Es necesario superar un 30%

PRIMERA PREGUNTA (40% parte ejercicios)

Crear un diagrama Entidad-Relación (que modele capturando la mayor semántica posible), usando características extendidas si es necesario y convertirlo posteriormente a una serie de **tablas del modelo de datos relacional**, para una sitio web de base de datos online de videos (películas, episodios...).

- La base de datos almacena videos. Cada video tiene un identificador único, un título, una fecha de estreno y su productora.
- Hay 3 tipos de videos: películas, episodios de series y colecciones.
- Cada episodio es un video diferente, que pertenece a una colección (o más) y tiene un número de episodio.
- Las colecciones tienen un nombre descriptivo y agrupan los episodios, y contienen una o varias temporadas (que deben registrarse). Por ejemplo la colección “CSI: las temporadas de Grissom” tiene las temporadas 1,2,3,4,5,6,7,8 y 9.
- El número máximo de episodios que se puede incluir en una colección es de 50.
- En las películas no se tiene una numeración y son todas independientes (“Karate Kid”, “Karate Kid II” y “Karate Kid III” son videos diferentes). Debe conocerse la recaudación en taquilla que obtuvo la película.
- Cada video tiene asignados uno o más géneros (“Comedia”, “Drama”, “Acción”, etc.), con un porcentaje determinado de importancia del género en el video. Por ejemplo, “Karate Kid” es un 70% acción, un 20% drama y un 10% comedia.
- Cada género tiene una popularidad relativa dentro de la base de datos.
- Los usuarios otorgan puntuaciones de 0 a 5 a los videos. Concretamente se puntúa cada uno de los géneros a los que pertenece el video. Por ejemplo, un usuario puede puntuar con un 5 la parte de acción de “Karate Kid”, con un 2 la parte de drama y con un 3 la de comedia. Deben puntuarse todos los perfiles de género de la película. Un mismo usuario puede puntuar más de una vez la misma película en fechas diferentes.
- La información que se almacena de los usuarios es el identificador de usuario, el nombre de usuario y la nacionalidad.
- Los usuarios pueden ser amigos de otros usuarios, formando el embrión de una red social del sitio. Debe almacenarse la fecha en que un usuario aceptó la solicitud de amistad de otro usuario.
- Todos los videos tienen una puntuación media, que se calcula a partir de las puntuaciones de los usuarios.

Indicar en un listado las **RESTRICCIONES QUE NO QUEDAN RECOGIDAS EN EL DIAGRAMA**.

Nota: Sólo es necesario representar aquellos atributos de las entidades que sean fundamentales para comprender el esquema. Representar de manera más amplia (de la manera que parezca más razonable) los aspectos que no queden completamente definidos en el enunciado anterior. Razonar en cada caso la representación que se elige.

SEGUNDA PREGUNTA (40% parte ejercicios)

Para la base de datos del anexo, escribir expresiones en SQL para las siguientes consultas (convertirlas a ANSI SQL86 (SQL1) en caso de usar características de un SQL más avanzado).

- SQL – Eliminar toda la información relacionada con la investigación “Gortel”.
- SQL y **Álgebra Relacional** – Parejas de DNI y nombre de juez y fiscal que viven en la misma ciudad y han actuado juntos en al menos dos juicios de corrupción. (Ej: 1 Paco y 2 Carmen porque han actuado como juez y fiscal juntos en 2 juicios y los dos son de corrupción).
- SQL – DNI y nombre del ciudadano (o ciudadanos) que vive en Oviedo y que tiene el mayor número de imputaciones.
- SQL – DNI y nombre de los ciudadanos que han obtenido sentencias de “no culpable” en todos los juicios celebrados por jueces que viven en su misma ciudad. (Ej: 7 Luis, porque Luis vive en Oviedo y fue acusado en 5 juicios celebrados por jueces que viven en Oviedo, y en los 5 juicios salió no culpable).

TERCERA PREGUNTA (20% parte ejercicios)

Encontrar una descomposición de producto sin pérdida (PSP), que conserve las dependencias (CD), y normalizada en 3FN o BCNF lo mejor posible (mínimo número de esquemas y de redundancia, etc.) del esquema de relaciones $R = (A, B, C, D, E, F, G, H, I, J, K)$ y el conjunto de dependencias F:

$$F = \left\{ \begin{array}{l} B \rightarrow FEG \\ FE \rightarrow HD \end{array} \right.$$

$$\begin{array}{l} I \rightarrow JK \\ CDI \rightarrow AB \end{array}$$

Indicar la forma normal de cada relación y mostrar que la descomposición encontrada efectivamente es de producto sin pérdida y conserva las dependencias.

ANEXO

Base de datos relacional que almacena información acerca de investigaciones y juicios.

Funcionario(f_DNI, f_nombre, f_ciudad)

// Datos de los funcionarios (tanto jueces como fiscales): DNI, nombre y ciudad de residencia

Ej: (1, "Paco", "Oviedo") – El funcionario 1 se llama Paco y vive en Oviedo

Ej: (2, "Carmen", "Gijón")

Fiscal(f_DNI, f_ámbito) [*f_DNI referencia Funcionario*]

// Datos de los fiscales: DNI y ámbito (civil, penal, social)

Ej: (1, "civil") – El funcionario 1 (Paco) es un fiscal de civil

Juez(f_DNI, j_número_juzgado, j_ámbito) [*f_DNI referencia Funcionario*]

// Datos de los jueces: DNI, número del juzgado del que es titular y ámbito del mismo (civil, penal, social)

Ej: (2, "5", "penal") – El juez 2 (Carmen) es titular del juzgado 5 de lo penal

Ciudadano(c_DNI, c_nombre, c_ciudad)

// Datos de los ciudadanos: DNI, nombre y ciudad de residencia

Ej: (3, "Alberto", "Oviedo") – El ciudadano 2 se llama Alberto y vive en Oviedo

Juicio(j_código, j_fecha, f_DNI-juez, f_DNI-fiscal, j_tipo) [*f_DNI-juez referencia Juez, f_DNI-fiscal referencia Fiscal*]

// Datos de los juicios: código, fecha, juez que juzga y fiscal que acusa en el juicio, tipo del juicio (corrupción, terrorismo, otros)

Ej: ("J-1", "12/05/2015", 2, 1) – El juicio J-1 se celebrará el 12/05/2015. El juez será el 2 (Carmen) y el fiscal el 1 (Paco)

Investigación(i_código, i_nombre_clave, i_código-principal) [*i_nombre_clave clave candidato*] [*i_código-principal referencia Investigación*]

// Datos de las investigaciones: código, nombre clave, investigación de la que es subinvestigación

Ej: ("I-2", "Pokemon", "I-1") – La investigación I-2 tiene nombre en clave Pokemon y es una subinvestigación de la investigación I-1 (Gortel).

Ej: ("I-1", "Gortel", -)

Imputado(i_código, c_DNI)

// Ciudadanos que están imputados en investigaciones

Ej: ("I-1", 3) – El ciudadano 3 (Alberto) está imputado en la investigación I-1 (Gortel)

Es_acusado(i_código, c_DNI, j_código, sentencia, multa)

// Ciudadanos que son acusados en un juicio derivado de una investigación, indicando la sentencia recibida (culpable, no culpable) y la multa aparejada.

// Ej: ("I-1", 3, "J-1", "culpable", 10M) – El ciudadano 3 (Alberto) es acusado en el juicio J-1 y salió culpable con 10 millones de euros de multa.

Vista(j_código, v_fecha, v_contenido, v_número) [*j_código+v_número clave candidato*]

// Vistas que se celebran de cada juicio, indicando el código del juicio, la fecha de la vista, el número que hace y su contenido)

Ej: ("J-1", "12/05/2015", "vista preliminar", 1) – El 12/05/2015 se celebró la vista preliminar del juicio J-1. Esta vista hace la primera de ese juicio.

Nota 1 : DEBEN usarse las abreviaturas
número : n // nombre : nom_ // código: cód

Nota 2 : Supónganse los dominios acostumbrados para
cada atributo

Nota 3 : Nombres alternativos para las tablas:

Funcionario fu | Juez jz | Fiscal fi | Ciudadano ci | Juicio
jc | Investigación iv | Imputado ip | Es_acusado ac | Vista vi