**Trabajo N°3 Bases de Datos 2**

*Darwin Herrera Melina Muñoz*

**PUNTO 1**

Los atributos de las tablas deben de ser ingresados separados por comas; el orden de poner los atributos es:  
*clavePrimaria, claveForanea (En caso de requerirse), demás atributos…*

Las consultas se deben separar por “;” en caso de realizar más de una consulta en los campos especificados. Los atributos de las tablas, los nombres de los atributos deben ser distintos entre todas las tablas, los atributos también deben ser distintos de los nombres de las tablas.

Dentro de las consultas se pueden extraer los atributos:

* Selección de varias tablas
* Uso de los atributos en la sección where.

No son validas las consultas tipo ***SELECT \* FROM...***

Para desarrollar el punto se hizo uso de formularios para recibir la información; internamente en Java se hacen el uso de estructuras de datos integradas en la librería java.util.\*, Estas estructuras tienen comportamiento de conjuntos y contienen métodos que facilitan las operaciones de conjuntos, lo cual facilito implementar las formulas empleadas en el paper.

Para el ingreso de los datos se hizo una “limpieza” de datos usando regex para quitar los espacios en blanco y no se encontrarán problemas a la hora de comparar cadenas de texto.

**d) Mejora 1:** En mi concepto se debe de mejorar la no repetición de atributos en y esto debido a que se pueden tener atributos en las dos simultáneamente lo cual hace que en mi opinión exista información redundante que tiene la misma eficiencia en las consultas, un caso para ejemplificar eso es:  
T1: a, b, c.  
T2: a, d, e ,f  
S1: d;  
 d, e;  
S2: b, d;

En este caso Q’={a,b,c T2’={d, e}, T2”={d}}

**Mejora 2:** En mi concepto se debe de implementar alguna metodología hibrida con la cual se permita hacer la creación de documentos NoSQL con base en tablas relaciones, considerando las consultas como lo hacen el paper leído pero también sin dejar por fuera atributos. Mi propuesta es tener en cuenta las consultas que se realizan para crear T’ y T” y crear un documento embebido donde se pongan los atributos no consultados, todo esto para que en un futuro no se queden atributos excluidos o no se puedan realizar consultas con los atributos necesarios porque estos no existen en la base de datos NoSQL

**e)**En la Nota 2 de la página 8, ver líneas resaltadas en azul en el artículo los autores afirman lo siguiente: “*If the relational database schema initially satisfied the normal form of BCNF, 3NF, or 4NF, then there can be no other relationships between tables T1 and T2.* ”

¿es correcta esta afirmación?

Si lo es explique porque lo es;

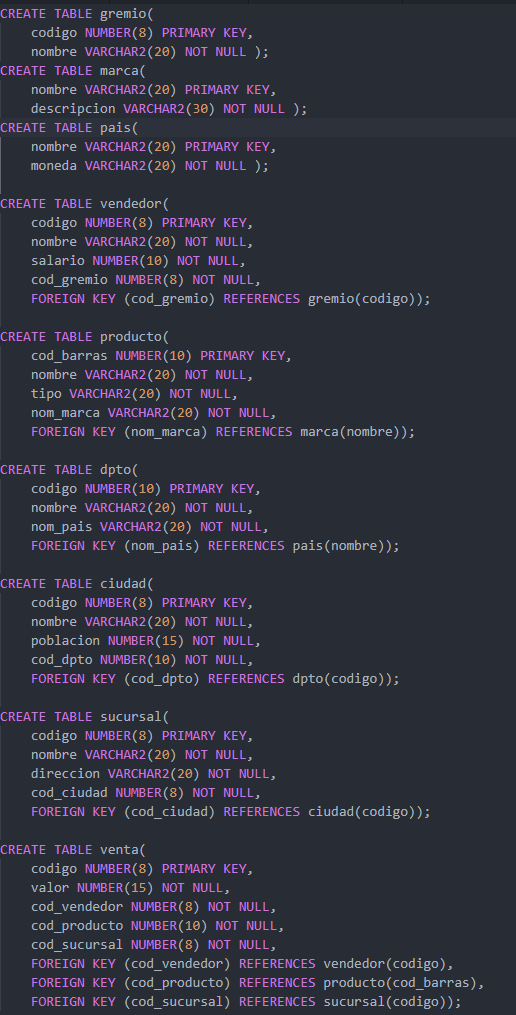
Si no lo es presente un contraejemplo (¿quizás los autores pasaron por alto algo?).

**R/**

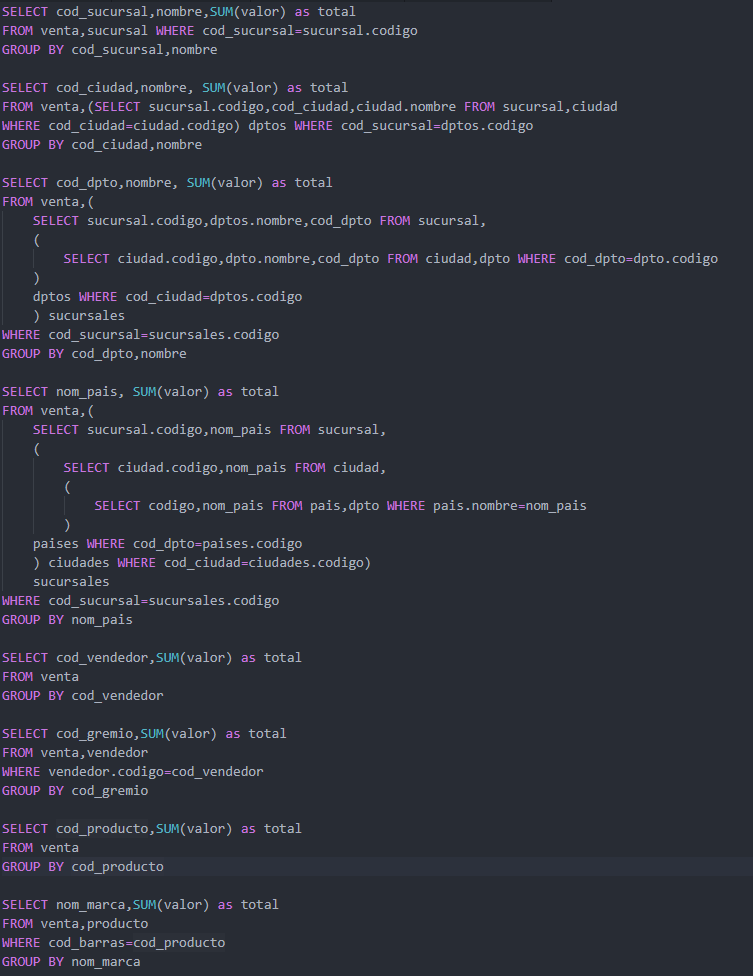
En el contexto en la que esta subrayada la oración y con base a que hablan exclusivamente de relaciones 1-M, lo cual hace que el comentario este muy acertado debido a que con el supuesto que se haces sí existen mas relaciones o tablas intermedias entonces se puede decir que no cumplía las formas normales lo cual contradice lo dicho por el autor en el texto resaltado.

**PUNTO 2**

Primero creamos las tablas como lo muestra la siguiente imagen:



Después procedemos a crear las consultas como lo muestra la siguiente imagen



Teniendo las consultas listas lo que implementamos en java es las consultas y con la información que extraemos la subimos a Mongo para después ser consultada y mostrada con el botón de “visualizar”.

Para sacar la info que sacan las consultas lo que debemos de hacer es usar el mismo driver usado en el trabajo 2 y conectarnos a la maquina local del equipo.