

Proyecto Semestral: Consultas

Bases de Datos

“Prestigio-2A”

Diego Altamirano Javier Mendoza
`diego.altamiranot@sansano.usm.cl` `javier.mendoza@sansano.usm.cl`

Felipe Quintanilla Kevin Reyes
`felipe.quintanilla@sansano.usm.cl` `kevin.reyesv@sansano.usm.cl`

Carlos Buil A.
`cbuil@inf.utfsm.cl`

17 de Noviembre

Introducción

Con los modelos de datos ya diseñados, la siguiente tarea a realizar es la de generar algunas consultas que consigan simular solicitudes reales que se realizarían en una implementación real de nuestra idea.

En concreto, a cada uno de los grupos se le asignarán una serie de solicitudes que deberán resolver diseñando la consulta de cada una de ellas. Dentro de estas, será necesario considerar casos de uniones, de conteo, de clasificación y complejidades varias.

Requisitos

En específico, para esta entrega se les pedirá:

- Corregir su modelo de datos, en base al feedback recibido (de la entrega 1) y a los nuevos conocimientos que poseen sobre el enfoque *Top-Down* y el enfoque *Bottom-Up*.
- Generar las consultas SQL necesarias para responder a las solicitudes que a cada grupo se le entregará.
 - Si su modelo presentado no posee las entidades necesarias para resolver una consulta, esto deberá ser solucionado en su nueva propuesta del mismo. Si este es el caso, por favor indicarlo en alguna sección de su entrega (Por ejemplo en la introducción del Notebook).
- Responder con consultas SQL a dos solicitudes que ustedes generen, en base a criterios establecidos.
- Poblar su base de datos, opcionalmente de forma automática. De esta forma, podrán comprobar el funcionamiento de sus consultas.

Corrección Modelo de Datos

Para poder generar las consultas SQL que responden a las solicitudes que les entregaremos, deben **corregir su modelo para que sea capaz de soportar el contexto en cuestión**. Según este contexto, las solicitudes que les entregaremos son perfectamente factibles y aplicables en la realidad.

Recuerden que el Modelo de Datos debe ser capaz de reflejar la realidad que modela. Es conveniente que tengan en cuenta las siguientes consideraciones:

- Lo más importante en el modelo de una tienda es poder llevar un buen recuento de las compras y otros tipos de transacciones.
- Los juegos y otros tipos de contenido extra (DLC/OST/etc) son muy similares: Todos se compran, todos pueden ser parte de un Bundle/Paquete, todos tienen precio, etc.
- Los precios no son estáticos: un Juego puede abarataarse a través del tiempo (por dejar de ser nuevo, por ejemplo) o estar temporalmente en oferta. Interesa saber **cuánto valía** el juego (u otro contenido) un día determinado.

Generar consultas SQL

A cada grupo, de un pool de preguntas, se le entregará **8 solicitudes que deben responder con consultas SQL**. Las solicitudes que les entregaremos varían en dificultad, considerando (como se mencionó en la introducción) casos de uniones, conteos, agrupaciones, etc.

Generar solicitudes y consultarlas

En base a su modelo, deberán **generar 2 solicitudes** que permitan extraer información de interés. Además, deberán **crear las consultas SQL que responden a esas solicitudes**. En específico, sus solicitudes deben cumplir con los siguientes requerimientos:

- Cada consulta debe trabajar con al menos un JOIN.
- Al menos una solicitud debe ser resuelta utilizando operaciones de agrupación y GROUP BY.
- Al menos una solicitud debe ser resuelta utilizando Consultas Anidadas.
- Si su modelo considera elementos extras que no son considerados por las consultas dadas, considere plantear sus consultas sobre estas entidades.
- Plantee consultas que puedan entregar información significativa.

Población de BD

Las 10 solicitudes que deben resolver deben poder probarse en una base de datos ya poblada, tal que se logren comprobar y ejecutar sin problemas. Para ello, se debe considerar que:

- Se les entregará un Jupyter Notebook como plantilla, el cual tendrá una estructura básica para poder hacer transacciones desde python. De esta forma, pueden escribir las funciones que generen los datos de forma automática. La automatización del poblado de la Base de Datos es **opcional, pero altamente recomendado**.

- La cantidad de filas necesarias para las entidades principales son al menos 25. Para las entidades de intersección, considere utilizar al menos 100 filas.
- Si alguna tabla de su modelo no entra en una categoría anterior o debería tener un número diferente de filas, puede consultar con los ayudantes para determinar un número adecuado.
- Los datos deben ser adecuados a la lógica del negocio. Por ejemplo, no añada un juego de nombre *djksdsjh* o una persona llamada *League of Legends*.

Condiciones de Entrega

Las condiciones para su entrega son:

- El plazo de entrega es hasta el **Domingo 6 de Diciembre** a las **23:55 PM**, tareas atrasadas recibirán un **descuento** lineal hasta las 14:00 PM del día *Lunes 7 de Diciembre*.
- Para la **corrección del modelo de datos**, se solicitará un documento (pdf, md, docx, etc) con una imagen del nuevo modelo, los **supuestos considerados** (que sean relevantes al modelamiento) y una *breve* explicación de las entidades representadas (Ej: La tabla “*Asignatura*” representa la asignatura en el plan de estudios; la tabla “*Asignatura tomada*” representa la relación entre 1 alumno, 1 asignatura y 1 semestre). Además se debe entregar el archivo MySQLWB respectivo.
- Respecto a las **consultas**, estas deben ser adjuntas dentro del Jupyter Notebook, en el cual se deben separar las consultas por celda, y describir *brevemente* cómo esta resuelve la solicitud. Se les entregará una plantilla para completar este paso.
- Si automatizaron el ingreso de datos a la Base de Datos, deben incluir en el Jupyter Notebook las funciones que se utilizaron para poblar automáticamente la Base de Datos.

En resumen, deben entregar 3 archivos: el Documento o Informe en formato **.md**, **.pdf**, **.docx** (con imagen del modelo adjunta); el Modelo **.mwb**; y el Jupyter Notebook. Todos los archivos a entregar deben estar subidos en el repositorio git correspondiente a su grupo (el mismo donde entregaron la entrega 1), dentro de una carpeta de nombre **entrega-2**.