

# Tarea Chica 2

Profesores Vicente Domínguez y Luis Ramírez

Anunciada: 9 de septiembre de 2019

---

## Indicaciones

- Fecha de Entrega: 23:59 hrs el 16 de septiembre de 2019.
  - Debes entregar tu tarea en tu repositorio GitHub privado asignado para esta evaluación.
  - Cada hora o fracción de atraso descuenta 0,5 puntos de la nota que obtengas.
  - La tarea es *individual*. La copia será evaluada con nota 1 en el la tarea, además de las sanciones disciplinarias correspondientes.
  - La resolución de las consultas mediante [consultas anidadas](#) conllevará un descuento de 0.5 en el puntaje de esta.
  - Para realizar las consultas en SQL se recomienda el uso de [SQLiteOnline](#).
- 

## Objetivo

El objetivo de esta tarea es que aprendas a:

- Modelar la base de datos de un problema.
- Hacer consultas en una base de datos utilizando SQLite.

## Parte I: Diagrama (2.5 puntos)

Para esta tarea, tendrás que modelar una tienda online de películas. La base de datos que se encuentra a tu disposición cuenta con lo siguiente:

USERS	
id_user	integer
name	varchar(120)
email	varchar(120)
country	varchar(120)
favorite_category	integer

MOVIES	
id_movie	integer
title	varchar(120)
price	integer

CATEGORIES	
id_category	integer
name	varchar(120)

MOVIES_CATEGORIES	
id_movie	integer
id_category	integer

RATINGS	
id_user	integer
id_movie	integer
rating	float

PURCHASES	
id_purchase	integer
id_user	integer
id_movie	integer
date	date

Considera que

- Un usuario puede comprar una película a la vez.
- Un usuario puede comprar una película más de una vez
- Cada película puede tener más de una categoría asociada
- Un usuario sólo puede tener una categoría favorita
- Un usuario puede hacer rating de una película una sola vez
- Los ids en las tablas son únicos.

Debes realizar el diagrama E-R correspondiente a lo descrito anteriormente. En el informe puedes incluir los supuestos que estimes conveniente, siempre y cuando estos no contradigan el enunciado.

## Parte II: Consultas (2.5 puntos)

Deberás obtener la siguiente información, mediante una consulta:

1. Nombre y email de los usuarios que compraron el 2018 y son del país Uruguay (La combinación nombre - email no debe repetirse en las filas de la tabla). **(0.5 puntos)**.
2. Título y suma recaudada de todas las películas que empiecen con la letra 'A' mayúscula. **(0.5 puntos)**.
3. Nombre y cantidad de películas que poseen las 5 categorías con mayor cantidad de películas. **(0.5 puntos)**.
4. Título y precio de todas las películas que posean al menos 7 categorías ordenadas de mayor a menor según su precio. **(1 punto)**.

## Parte III: Preguntas (0.25 puntos c/u)

Debes responder las siguientes preguntas en la sección correspondiente:

1. De 3 razones por las cuales es útil usar una base de datos.
2. La siguiente consulta es correcta o no en cuanto a sintaxis: 'SELECT users.name, users.email FROM users, ratings WHERE users.id\_user = ratings.id\_user AND COUNT(\*) >2 GROUP BY users.name;'. En caso de no ser correcta, explique que le cambiaría para que funcionase.
3. ¿Qué es la redundancia en los datos y por qué es importante evitarla?
4. De un ejemplo de donde utilizaría una base de datos.

## Bonus I (0.5 puntos)

Entrega una consulta en lenguaje SQL que entregue lo siguiente:

- Título y rating promedio redondeado al segundo decimal de las películas que sean de la categoría Comedy o Sci-Fi. Estas deben tener un rating promedio entre 4 y 5 excluyéndolos. Lo anterior debe ser ordenado de mayor a menor según rating promedio.

## Bonus II (0.1-0.2 puntos)

Completa la siguiente imagen, **relacionándola con la materia de bases de datos**.

" \_\_\_\_\_ / \_\_\_\_\_ "



- Podrás obtener de 0.1 a 0.2 puntos dependiendo de la relación que tenga tu respuesta con la materia y que tan graciosa sea esta.
- Puedes encontrar la imagen en tu repositorio para agregarla a tu informe en caso de que desees hacer este bonus. Cabe señalar que puedes editarla si lo estimas necesario.

## Entregables

- Deberás subir a tu repositorio un informe en formato **PDF** donde incluyas el diagrama realizado en la parte I, las consultas de la parte II, las respuestas a las preguntas de la parte III y las respuestas a los bonus, si es que decidiste hacerlos.
- También debes subir un archivo llamado **consultas.txt** con las consultas de la parte II y el bonus I. Cada consulta debe estar en **una sola línea** y terminadas por el ; correspondiente. Debes seguir exactamente el siguiente formato de enumeración: el número de consulta, un punto y un espacio. Si no realizas una consulta debes numerarla de todos modos y dejar el resto en blanco.

```
1. SELECT * FROM Tabla1;  
2. SELECT * FROM Tabla2;  
3.  
Bonus. SELECT * FROM tabla3;
```

- Habrá un descuento asociado de máximo **1 pto** si es que tu informe está desordenado (esto a criterio del corrector).
- Si no subes el archivo *consultas.txt* o no sigues el formato pedido para este, tendrás un descuento de **1 pto**
- Las preguntas de bonus son independientes, puedes optar por realizar uno o ambos.