

# Examen Final

## Bases de Datos - Laboratorio

Juan Cabral, Sergio Canchi,  
Cristian Cardellino, Ramiro Demasi

### Parte I: SQL

Trabajaremos sobre la base de datos “**company**” cuyo diagrama se presenta en el archivo **eer-model.jpg**. El script **schema.sql** contiene los DDLs de la tablas mientras que el script **data.sql** contiene los datos de la misma.

Para cargar la base de datos ejecutar los siguientes comandos:

```
# Si va a trabajar en la computadora del laboratorio
$ mysql --host 172.18.7.140 --port 3306 -u MyUser -pMyPassword < schema.sql
$ mysql --host 172.18.7.140 --port 3306 -u MyUser -pMyPassword < data.sql
```

```
# Si va a trabajar desde su computadora personal
$ mysql --host <host> --port 3306 -u <user> -p < schema.sql
$ mysql --host <host> --port 3306 -u <user> -p < data.sql
```

### Consignas

Responda las siguientes consultas:

1. Listar nombre del proyecto y cantidad de horas trabajadas en total en el mismo de todos aquellos proyectos donde el total de horas trabajadas sea mayor o igual a 50hs ordenados por el total de horas trabajadas de forma descendente.
2. Listar SSN y nombre completo (en formato <Nombre> <Initial> <Apellido>) de aquellos empleados que no tienen empleados a cargo. Mostrar el resultado en orden alfabético.
3. Listar SSN y nombre completo de aquellos empleados que solo trabajen en proyectos controlados por el departamento de Research. Mostrar el resultado ordenados por SSN de forma descendente.
4. **(Ejercicio para Libres)** Crear un procedimiento almacenado que dado un nombre de departamento (parámetro de entrada), debe listar el nombre de proyecto, cantidad de empleados y total de horas trabajadas para cada proyecto controlado por dicho departamento

### Parte II: NoSQL

Trabajaremos sobre la base de datos “**supplies**” que contiene contiene una sola colección.

Colección	Descripción
sales	Cada documento de la colección <b>sales</b> representa una sola venta de una tienda administrada por la empresa de suministro. Cada documento contiene los artículos ( <b>items</b> ) comprados, información sobre el cliente que realizó la compra y varios otros detalles relacionados con la venta.

Para cargar la base de datos **supplies** ejecutar los siguientes comandos:

# Si va a trabajar en la computadora del laboratorio los datos ya se encuentran cargados en la base de datos asignada. Solo debe conectarse al servidor mongodb con:

```
$mongo --host 172.18.7.140 --port 27017 -u USER -pPASSWORD --authenticationDatabase
mongodb_LEGAJO_2022 mongodb_LEGAJO_2022
```

# Si va a trabajar desde su computadora personal

```
$ tar xzvf supplies.tar.gz
```

```
$ mongorestore --host <host> --port 27017 --drop --db supplies supplies/
```

## Consignas

Realizar las siguientes consultas en MongoDB.

5. Buscar las ventas donde el método de compra (*purchaseMethod*) sea 'Online' y la cantidad de *items* sea entre 5 y 8 inclusives. Listar el *email* y la edad (*age*) del cliente, y el monto total facturado, donde el monto de cada *item* se calcula como '*price \* quantity*'. Mostrar el resultado en orden alfabético.  
HINTS: El operador \$reduce puede ser de utilidad
6. Crear la vista *salesInvoiced* que calcula el monto total y monto promedio facturado por año y mes. Mostrar el resultado en orden cronológico. No se debe mostrar campos anidados en el resultado.
7. Especificar reglas de validación en la colección sales utilizando JSON Schema.
  - (a) Las reglas se deben aplicar sobre los campos: *saleDate*, *purchaseMethod* e *items* ( y todos sus campos anidados ). Inferir los tipos y otras restricciones que considere adecuados para especificar las reglas a partir de los documentos de la colección.
  - (b) Para testear las reglas de validación crear un caso de falla en la regla de validación y un caso de éxito (Indicar si es caso de falla o éxito)

## Puntos a tener en cuenta

- Mostrar únicamente los campos pedidos en la consigna y en el orden en el que se los pide (tanto a nivel fila/documento como a nivel columna/campo).
- Buscar hacer la consulta de la forma más sencilla posible.
- Resolver las consultas sin vistas salvo que se lo pida explícitamente.
- Se evaluará el correcto formato de las soluciones:

- El código entregado debe ser legible.
- Utilizar indentación de espacios de manera uniforme.
- No escribir toda la consulta en una sola línea. Usen buen criterio para separar partes de la consulta.
- Utilizar mayúsculas para denotar palabras clave de SQL (e.g. **`SELECT`**, **`INSERT`**, **`FROM`**, etc.).
- Utilizar CTEs (i.e. **`WITH`**).
- Algunas consultas pueden resolverse con `find` y `aggregate`. En estos casos se puede resolver eligiendo alguno de estos métodos.

## Entrega

- Se entregará un archivo comprimido **`soluciones.zip`** (con **`soluciones.sql`** y **`soluciones.js`** adentro) con las soluciones de los ejercicios. Separar las soluciones mediante comentarios de SQL/JavaScript.
- La entrega se hará en el [Aula Virtual](#) en el [apartado correspondiente](#).
  - Tendrán hasta las 18:30 para entregar el examen. La recomendación es empezar a subir antes de las 18 para evitar cualquier eventualidad.
  - Después de las 18:30 se cerrará la entrega y el final se considerará desaprobado.