

# Bases de Datos 2024

## Recuperatorio Parcial II: NoSQL

Juan Cabral, Sergio Canchi, Tadeo Cocucci,  
Ramiro Demasi, Guido Ivetta

### Contexto

La base de datos **university** contiene la siguiente colección:

Colección	Descripción
grades	Contiene información sobre las calificaciones de los alumnos en una clase determinada, incluidas las puntuaciones obtenidas en diferentes evaluaciones.

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

**Instructivo para los que usan sus computadoras personales:**

```
$ tar xzvf parcial2.tar.gz
$ mongorestore --host <host> --drop --gzip --db university university/
```

**Instructivo para los que usan las computadoras del laboratorio:**

El primer paso es obtener la imagen de mongo 8. Para ello, correr el siguiente comando:

```
$ docker pull mongo:8
```

**En caso de tener problemas**, descargar el archivo mongo8.tar.gz y usar el comando:

```
$ docker load -i mongo8.tar.gz
```

Ahora que ya tenemos la imagen, podemos levantar el contenedor y cargar los datos:

```
$ tar xzvf parcial2.tar.gz
$ docker run --name mongo-parcial2 \
    -p 27017:27017 -v mongod.conf:/etc/mongod.conf \
    -v ./university:/data/university -d mongo:8
$ docker exec -it mongo-parcial2 mongorestore \
    --drop --gzip --db university /data/university/
// (Opcional) para ejecutar el cliente
$ docker exec -it mongo-parcial2 mongosh
```

# Consignas

1. Escribir una consulta para calcular el promedio de puntuaciones de cada clase (class\_id) y compararlo con el promedio general de todas las clases. La consulta debe devolver un documento para cada clase que incluya el class\_id, el promedio de puntuaciones de esa clase y un campo adicional que indique si el promedio de la clase está por encima o por debajo del promedio general de todas las clases. Los resultados deben ordenarse de manera ascendente por class\_id y de manera descendente por average\_score.

Estructura de cada documento del output:

```
{
  "class_id": <class_id>,
  "average_score": <average_score>, // puntuación promedio de esta clase
  "comparison_to_overall_average": "above" | "below" | "equal" // comparación con el
  promedio general de todas las clases
}
```

HINT: una de las stages usa [lookup](#) con un pipeline adentro:

```
$lookup: {
  from: "grades",
  pipeline: [
    // TODO
  ],
  as: "overall_average"
}
```

2. Actualizar los documentos en la colección grades, ajustando todas las puntuaciones para que estén normalizadas entre 0 y 7

La fórmula para la normalización es:

$$\text{Valor Normalizado} = \frac{\text{Valor Original}}{100} \times 7$$

Por ejemplo:

Si un estudiante sacó un **32** y otro sacó un **62**, deberían ser actualizadas a:

- **2,24**, porque  $(32/100) \times 7 = 2,24$
- **4,34**, porque  $(62/100) \times 7 = 4,34$

HINT: usar [updateMany](#) junto con [map](#)

3. Crear una vista "top10students\_homework" que liste los 10 estudiantes con los mejores promedios para homework. Ordenar por average\_homework\_score descendiente.
4. Especificar reglas de validación en la colección grades. El único requerimiento es que se valide que los type de los scores sólo puedan ser de estos tres tipos: ["exam", "quiz", "homework"]

## Puntos a tener en cuenta

- Algunas consultas pueden resolverse con `find` y `aggregate`. En estos casos se puede resolver eligiendo alguno de estos métodos.
- Resolver las consultas sin vistas salvo que se lo pida explícitamente.
- Mostrar únicamente los campos pedidos en la consigna.
- Se piden que los campos que se devuelven sean valores escalares a menos que se pida los valores de los campos devueltos podrán ser documentos anidados, arreglos de escalares o arreglos de documentos.
- Buscar hacer la consulta de la forma más sencilla y eficiente posible.
- Se evaluará el correcto formato de las soluciones:
  - El código entregado debe ser legible.
  - Utilizar indentación de espacios de manera uniforme.

### Detalle Importante

El alumno puede usar sus notas personales e internet, pero queda prohibido utilizar modelos generativos (Chat-GPT, LLAMA, etc) y comunicarse con otras personas oral o digitalmente.

Romper alguna de estas dos restricciones invalida el examen

## Entrega

- Se entregará un archivo **`soluciones.js`** con las soluciones de los 6 ejercicios. Separar las soluciones mediante comentarios de Javascript (`/* */` o `//`).
- La entrega se hará mediante el Aula Virtual en el correspondiente apartado.
  - Tendrán hasta las 16:00 para que se considere una entrega completa. La recomendación es empezar a subir el archivo a las 15:55 para evitar cualquier eventualidad.