Parte 4:

Mini-proyecto:

Cada grupo debe modelar la base de datos para un sistema de reservas de vuelos en MongoDB usando una sola colección.

Cada documento representará una reserva e incluirá toda la información necesaria.

Requisitos:

* Pasajero: Nombre, contacto, identificación.
* Vuelo: Aerolínea, origen, destino, horario.
* Reserva: Código de reserva, estado (confirmada/cancelada), fecha de creación.

Preguntas obligatorias

1-Obtener todas las reservas confirmadas para vuelos cuyo destino sea "Madrid" y cuya fecha de vuelo sea posterior a "2025-06-01".

2-Contar cuántas reservas hay por aerolínea ($group).

3-Modificar solo las reservas con destino "París", asignando un descuento (nuevo campo).

**Modelo de documento (una sola colección)**

Colección llamada reservas.

Cada documento representa una reserva completa:

{

"\_id": ObjectId("..."),

"codigo\_reserva": "ABC123",

"estado": "confirmada",

"fecha\_creacion": "2025-05-20",

"pasajero": {

"nombre": "Juan Pérez",

"contacto": "+54113456789",

"identificacion": "DNI 34567890"

},

"vuelo": {

"aerolinea": "Iberia",

"origen": "Buenos Aires",

"destino": "Madrid",

"horario": "2025-06-15T10:00:00Z"

}

}

**1. Obtener todas las reservas confirmadas con destino "Madrid" y vuelo posterior a "2025-06-01"**

db.reservas.find({

"estado": "confirmada",

"vuelo.destino": "Madrid",

"vuelo.horario": { $gt: new Date("2025-06-01") }

});

**2. Contar cuántas reservas hay por aerolínea ($group)**

db.reservas.aggregate([

{

$group: {

\_id: "$vuelo.aerolinea",

total\_reservas: { $sum: 1 }

}

}

]);

**3. Modificar solo las reservas con destino "París", asignando un nuevo campo descuento**

db.reservas.updateMany(

{ "vuelo.destino": "París" },

{ $set: { descuento: 0.25 } }

);