**MongoDB Aggregation – Airline Management System**

The following aggregation queries are designed to generate insights and reports for the **Airline Management System**, which consists of four collections:  
flights, passengers, bookings, and crew.

BOOKINGS

db.bookings.aggregate([

{

$lookup: {

from: "passengers",

localField: "passenger\_id",

foreignField: "\_id",

as: "passenger\_info"

}

},

{ $unwind: "$passenger\_info" },

{

$lookup: {

from: "flights",

localField: "flight\_id",

foreignField: "\_id",

as: "flight\_info"

}

},

{ $unwind: "$flight\_info" },

{

$project: {

\_id: 0,

booking\_id: "$\_id",

passenger: "$passenger\_info.name",

flight: "$flight\_info.\_id",

source: "$flight\_info.source",

destination: "$flight\_info.destination",

seat\_number: 1,

price\_paid: 1

}

}

])

**Count total bookings per flight**

db.bookings.aggregate([

{

$group: {

\_id: "$flight\_id",

total\_bookings: { $sum: 1 }

}

},

{ $sort: { total\_bookings: -1 } }

])

**Calculate total revenue per flight**

db.bookings.aggregate([

{

$group: {

\_id: "$flight\_id",

total\_revenue: { $sum: "$price\_paid" }

}

},

{ $sort: { total\_revenue: -1 } }

])

**Passengers**

db.bookings.aggregate([

{

$group: {

\_id: "$passenger\_id",

bookingsCount: { $sum: 1 }

}

},

{ $match: { bookingsCount: { $gt: 1 } } }

])

**Show total amount spent per passenger**

db.bookings.aggregate([

{

$group: {

\_id: "$passenger\_id",

total\_spent: { $sum: "$price\_paid" },

total\_bookings: { $sum: 1 }

}

},

{ $sort: { total\_spent: -1 } }

])

**FLIGHTS**

db.flights.aggregate([

{

$project: {

flight\_number: "$\_id",

total\_seats: "$aircraft.total\_seats",

booked\_seats: { $subtract: ["$aircraft.total\_seats", "$available\_seats"] },

occupancy\_percentage: {

$multiply: [

{

$divide: [

{ $subtract: ["$aircraft.total\_seats", "$available\_seats"] },

"$aircraft.total\_seats"

]

},

100

]

}

}

}

])

**Count flights per status (Scheduled, Delayed, Cancelled)**

db.flights.aggregate([

{

$group: {

\_id: "$status",

total\_flights: { $sum: 1 }

}

}

])

**Find top source-destination pairs by number of bookings**

db.bookings.aggregate([

{

$lookup: {

from: "flights",

localField: "flight\_id",

foreignField: "\_id",

as: "flight\_info"

}

},

{ $unwind: "$flight\_info" },

{

$group: {

\_id: { source: "$flight\_info.source", destination: "$flight\_info.destination" },

total\_bookings: { $sum: 1 }

}

},

{ $sort: { total\_bookings: -1 } }

])

**Crew**

db.crew.aggregate([

{

$project: {

name: 1,

role: 1,

total\_flights: { $size: "$assigned\_flights" }

}

},

{ $sort: { total\_flights: -1 } }

])

**Show all flights with their assigned crew members**

db.crew.aggregate([

{ $unwind: "$assigned\_flights" },

{

$lookup: {

from: "flights",

localField: "assigned\_flights",

foreignField: "\_id",

as: "flight\_info"

}

},

{ $unwind: "$flight\_info" },

{

$project: {

\_id: 0,

crew\_name: "$name",

role: 1,

flight\_number: "$flight\_info.\_id",

source: "$flight\_info.source",

destination: "$flight\_info.destination"

}

}

])