# MongoDB Aggregation Operations

## Users Aggregations

• Count users by role:  
db.users.aggregate([{ $group: { \_id: '$role', count: { $sum: 1 }}}])

• List all travelers:  
db.users.aggregate([{ $match: { role: 'traveler' }}])

## Bookings Aggregations

• Count bookings per adventure:  
db.bookings.aggregate([{ $group: { \_id: '$adventureId', total: { $sum: 1 }}}])

• Find confirmed bookings:  
db.bookings.aggregate([{ $match: { status: 'confirmed' }}])

**Join user with their bookings**

$lookup

{

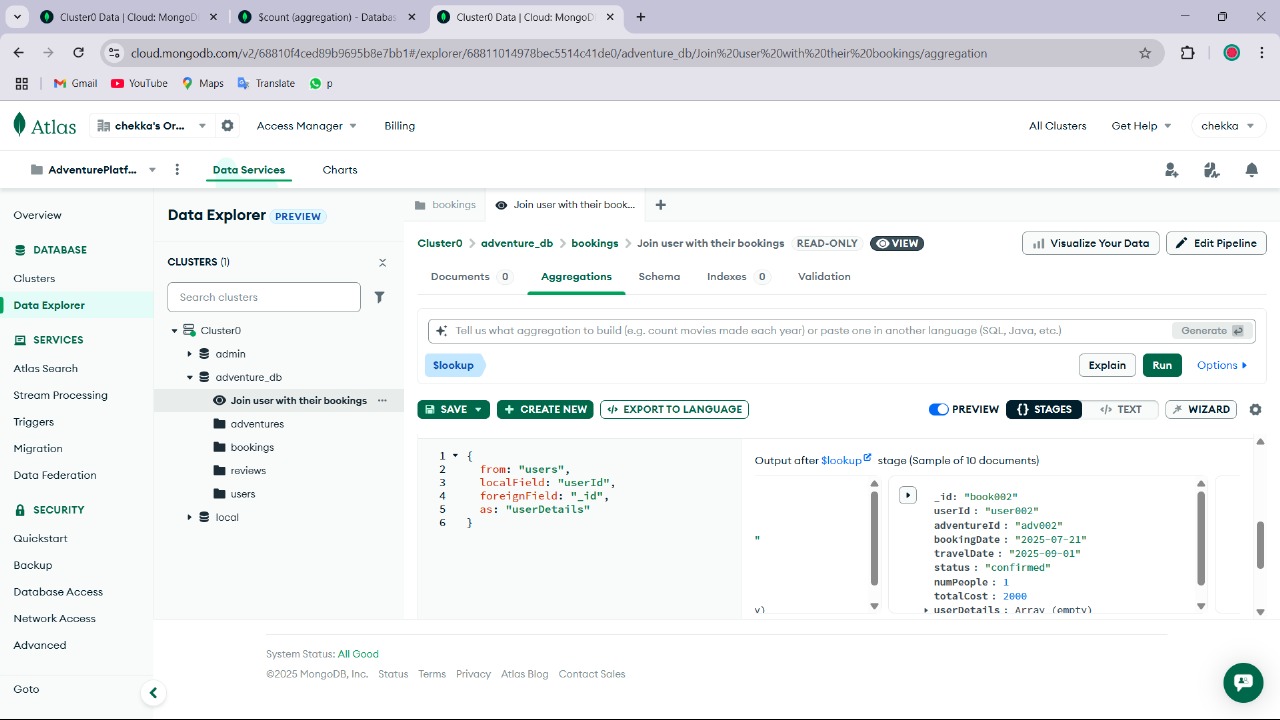
from: "bookings",

localField: "userId",

foreignField: "user\_id",

as: "user\_bookings"

}



**Join user with their bookings**

$lookup

{

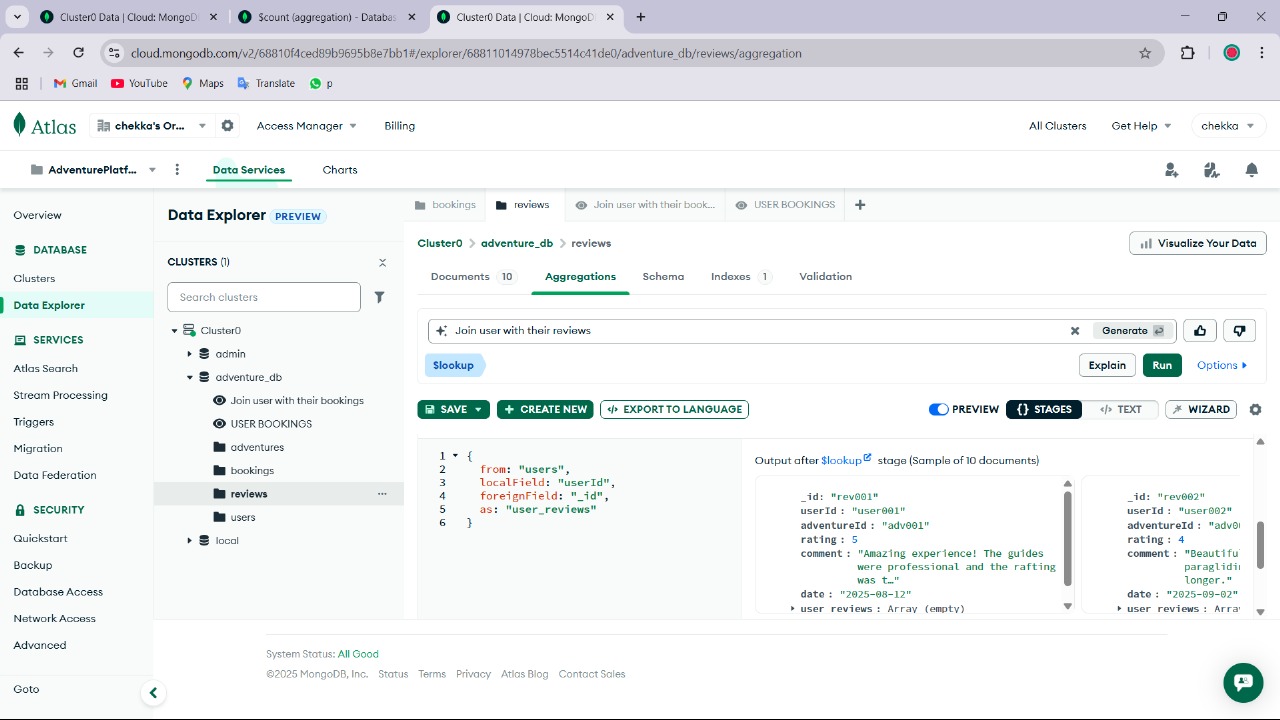
from: "bookings",

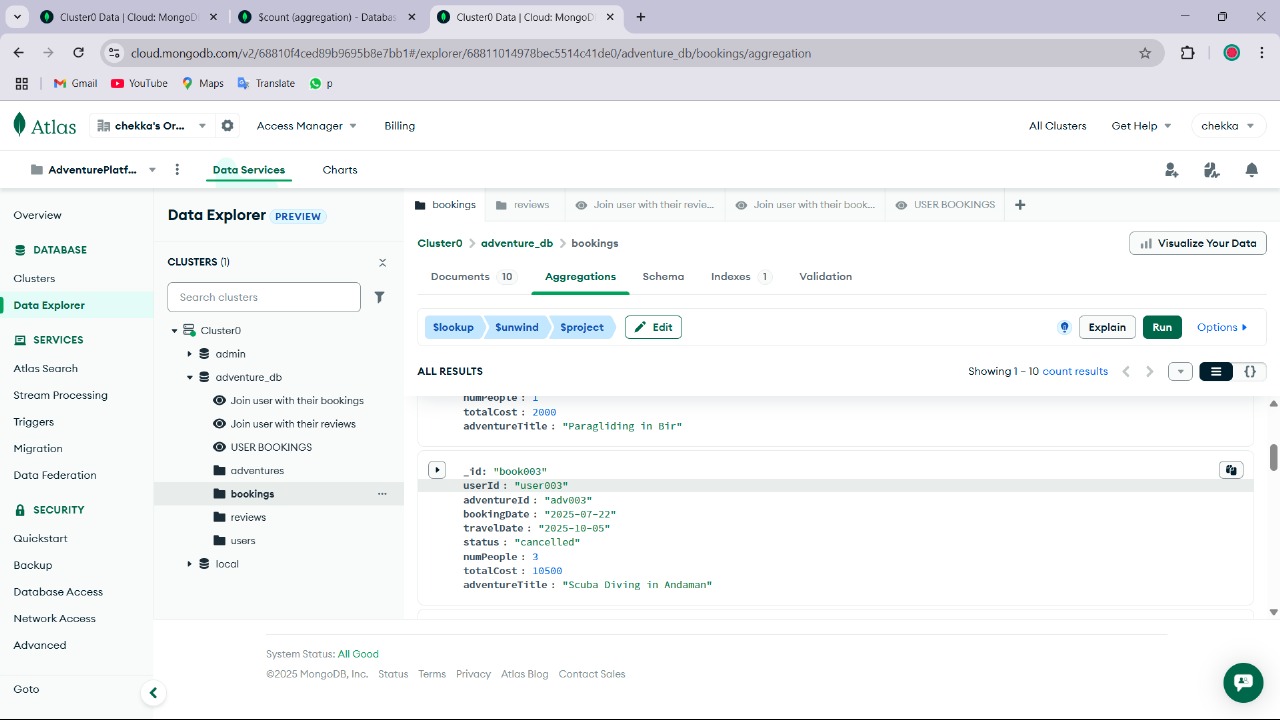
localField: "\_id",

foreignField: "adventureId",

as: "bookings"

}





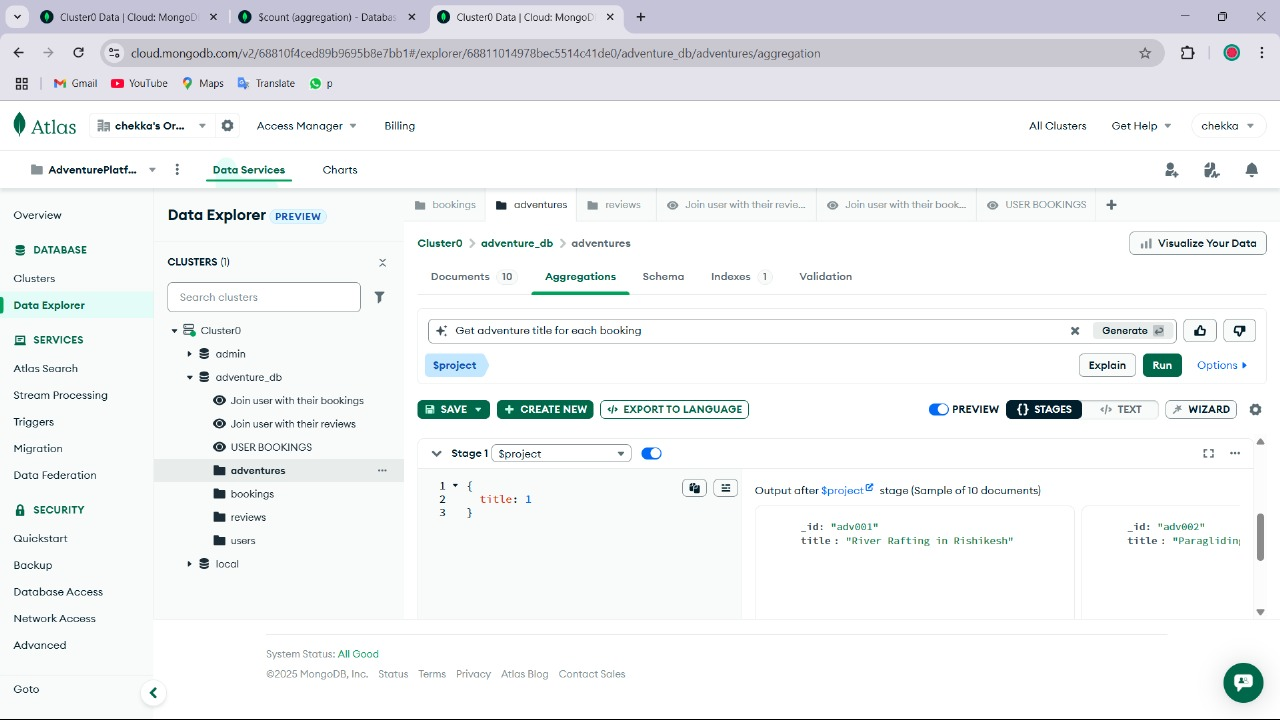
**Get adventure title for each booking**

$project

{

title: 1

}



**Format review details**

$project

{

\_id: 0,

adventureId: 1,

review: {

userId: "$userId",

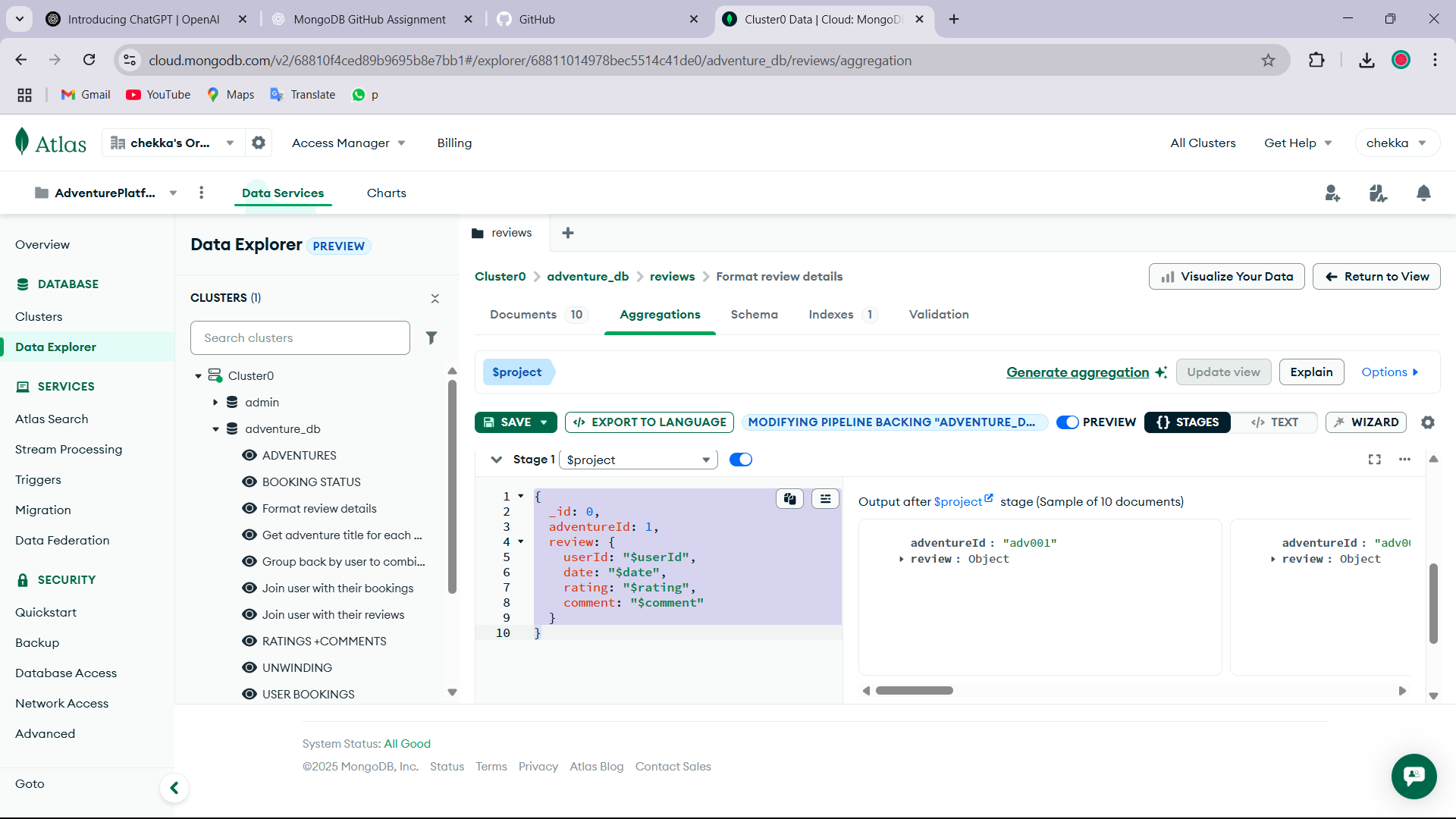
date: "$date",

rating: "$rating",

comment: "$comment"

}

}



**Booking Status**

{

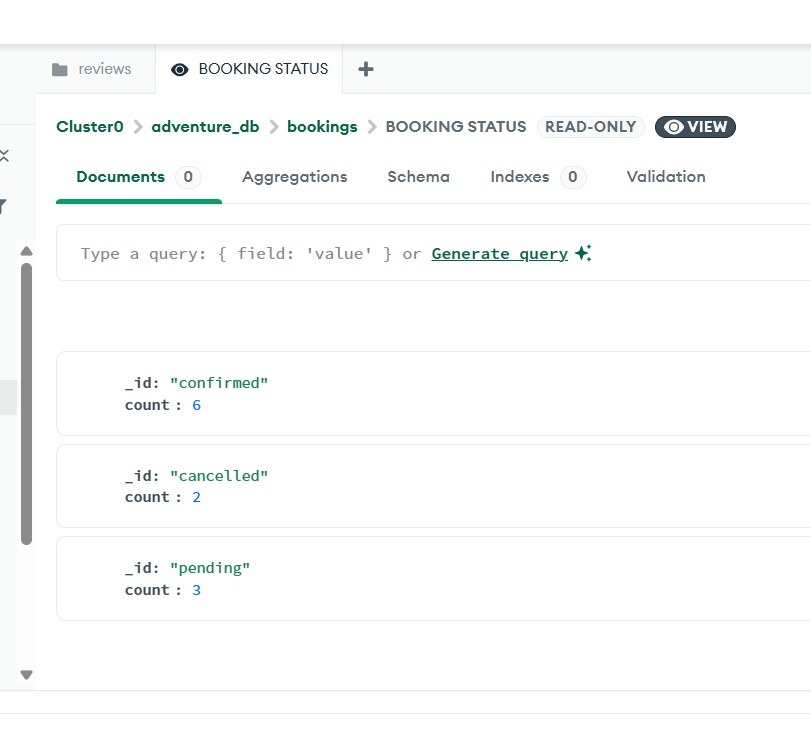
\_id: "$status",

count: {

$sum: 1

}

}

****

**Adventures**

{

\_id: 0,

adventureId: 1,

review: {

userId: "$userId",

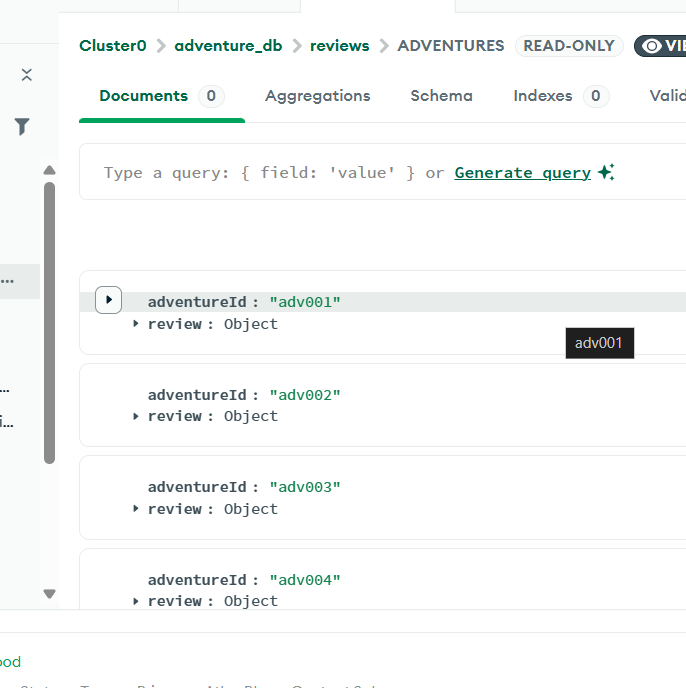
date: "$date",

rating: "$rating",

comment: "$comment"

}

}

****

**Group back by user to combine all data**

$group

{

\_id: "$\_id",

date: {

$first: "$date"

},

role: {

$first: "$role"

},

phone: {

$first: "$phone"

},

members: {

$first: "$members"

},

name: {

$first: "$name"

},

destination: {

$first: "$destination"

},

email: {

$first: "$email"

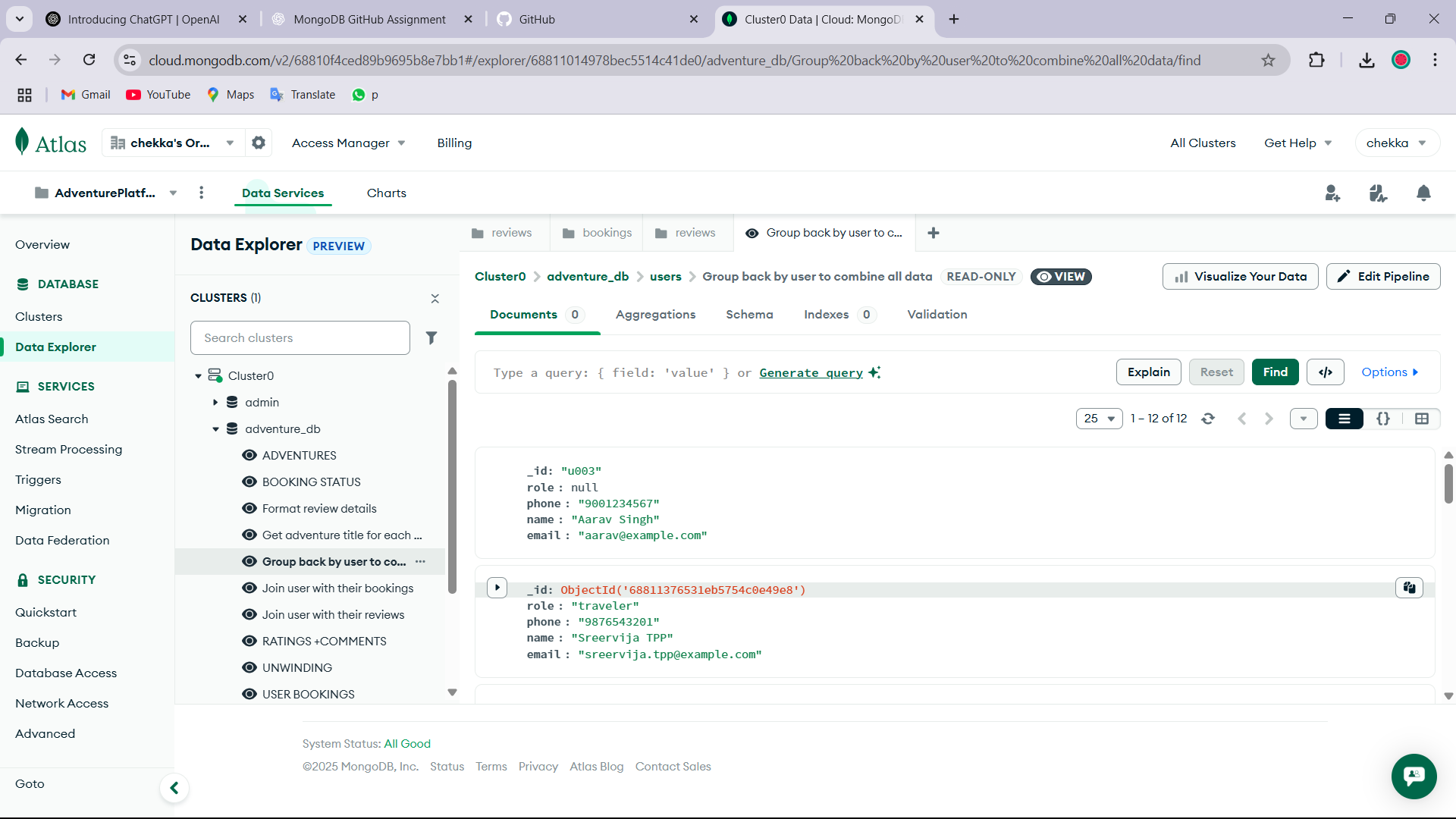
},

status: {

$first: "$status"

}

}



**RATINGS +COMMENTS**

$project

{

\_id: "$adventureId",

averageRating: {

$avg: "$rating"

},

comments: {

$push: "$comment"

}

}

