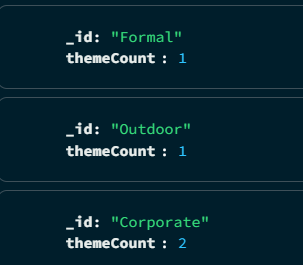
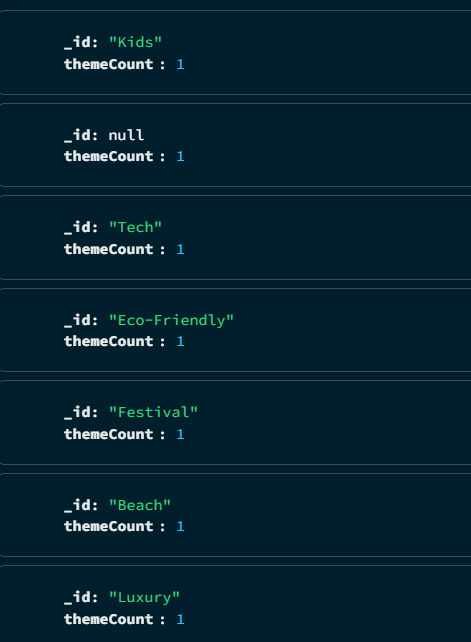
MongoDB Aggregation-Smart Event Management

# 📁 Themes Collection

## 1. Count of Themes per Category

Use: Understand how many themes exist for each category (e.g., Corporate, Luxury, Eco-Friendly).

db.themes.aggregate([  
 { $group: { \_id: "$category", themeCount: { $sum: 1 } } }  
])



## 2. Count of Themes by Primary Color

Use: Know which primary colors are most commonly used in themes.

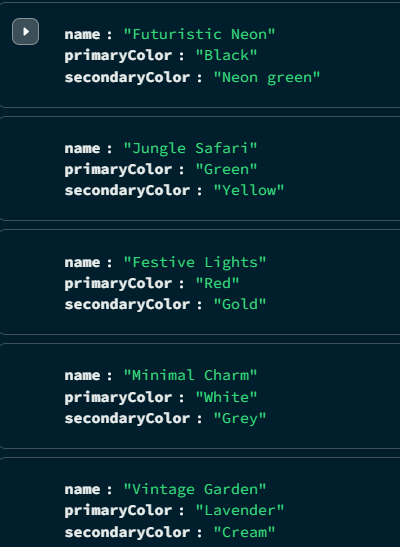
db.themes.aggregate([  
 { $group: { \_id: "$primaryColor", count: { $sum: 1 } } },  
 { $sort: { count: -1 } }  
])



## 3. List All Themes with Primary & Secondary Colors

Use: Display themes in a color palette preview or dashboard.

db.themes.aggregate([  
 {  
 $project: {  
 \_id: 0,  
 name: 1,  
 primaryColor: 1,  
 secondaryColor: 1  
 }  
 }  
])

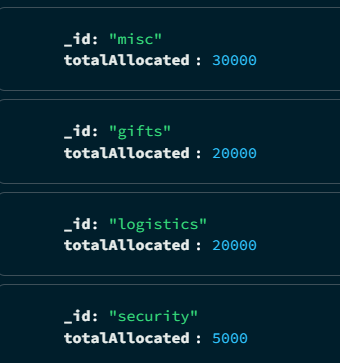
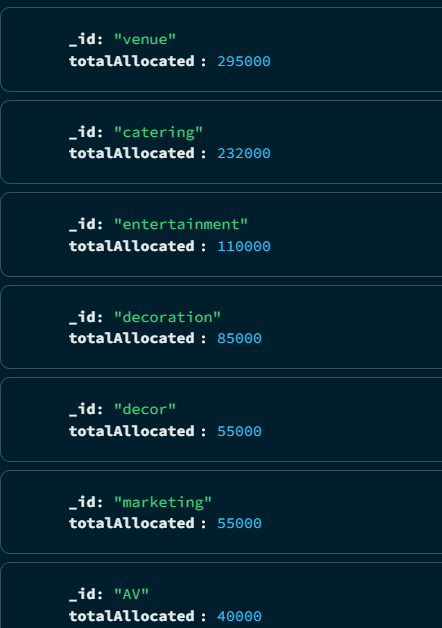


# 💰 Budget Collection

## 1. Total Allocated per Category Across All Events

Use: Analyze where most of the budget is spent overall.

db.budgets.aggregate([  
 { $replaceRoot: { newRoot: { allocations: "$categoryAllocations" } } },  
 { $project: { allocations: { $objectToArray: "$allocations" } } },  
 { $unwind: "$allocations" },  
 { $group: { \_id: "$allocations.k", totalAllocated: { $sum: "$allocations.v" } } },  
 { $sort: { totalAllocated: -1 } }  
])



## 2. Total Budget Per Event

Use: Show overall event budget comparisons.

db.budgets.aggregate([  
 { $project: { \_id: 0, eventId: 1, totalBudget: 1 } }  
])



## 3. Percentage Allocation per Category for Each Event

Use: Visualize spending distribution within events.

db.budgets.aggregate([  
 {  
 $project: {  
 eventId: 1,  
 totalBudget: 1,  
 allocations: { $objectToArray: "$categoryAllocations" }  
 }  
 },  
 { $unwind: "$allocations" },  
 {  
 $project: {  
 eventId: 1,  
 category: "$allocations.k",  
 amount: "$allocations.v",  
 percentage: {  
 $multiply: [  
 { $divide: ["$allocations.v", "$totalBudget"] },  
 100  
 ]  
 }  
 }  
 }  
])

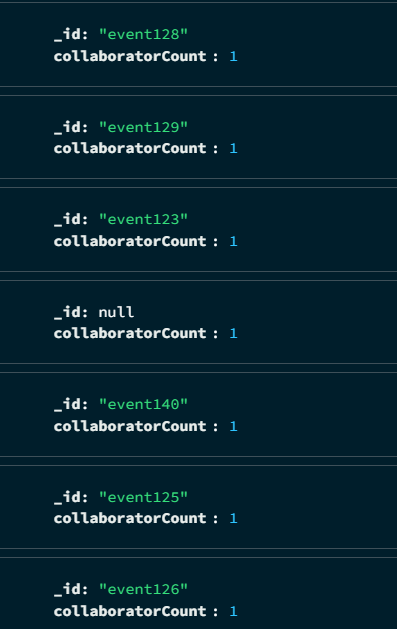


# 🤝 Collaboration Collection

## 1. Count of Collaborators per Event

Use: Shows how many collaborators are assigned to each event.

db.collaborators.aggregate([  
 {  
 $group: {  
 \_id: "$eventId",  
 collaboratorCount: { $sum: 1 }  
 }  
 },  
 { $sort: { collaboratorCount: -1 } }  
])



## 2. Top Collaborator Roles Across All Events

Use: Identify the most common roles used across events.

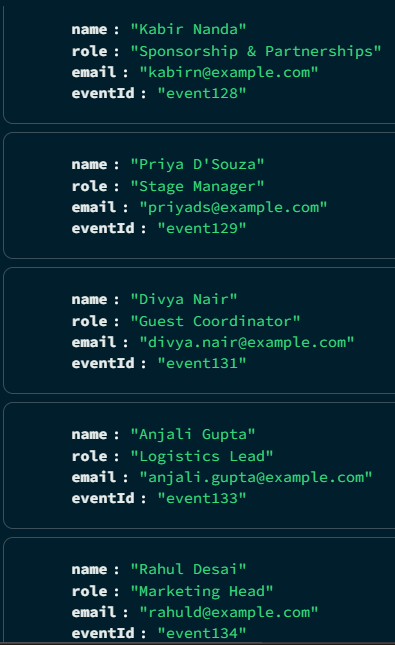
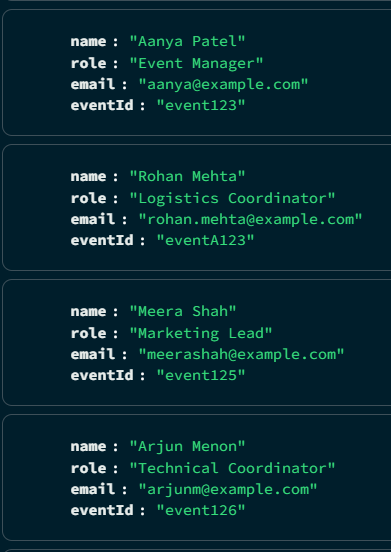
db.collaborators.aggregate([  
 {  
 $group: {  
 \_id: "$role",  
 count: { $sum: 1 }  
 }  
 },  
 { $sort: { count: -1 } }  
])



## 3. List of Collaborators with Email and Event

Use: Shows the full list of collaborators with contact info.

db.collaborators.aggregate([  
 {  
 $project: {  
 \_id: 0,  
 name: 1,  
 role: 1,  
 email: 1,  
 eventId: 1  
 }  
 }  
])



# 📅 Events Collection

## 1. Count of Events Per Location

Use: Shows how many events are happening in each city.

db.events.aggregate([  
 {  
 $group: {  
 \_id: "$location",  
 eventCount: { $sum: 1 }  
 }  
 },  
 { $sort: { eventCount: -1 } }  
])



## 2. Count of Events by Category

Use: Find out how many events fall under each category.

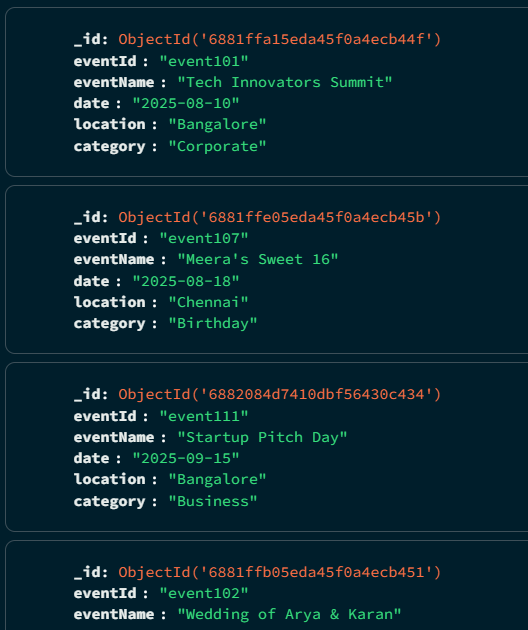
db.events.aggregate([  
 {  
 $group: {  
 \_id: "$category",  
 count: { $sum: 1 }  
 }  
 },  
 { $sort: { count: -1 } }  
])



## 3. List Upcoming Events

Use: Display events happening after today's date.

db.events.aggregate([  
 {  
 $match: {  
 date: { $gt: "2025-07-24" }  
 }  
 },  
 { $sort: { date: 1 } }  
])



# 📝 Feedback Collection

## 1. Average Rating per Event

Use: Get the average guest rating for each event.

db.feedback.aggregate([  
 {  
 $group: {  
 \_id: "$eventId",  
 averageRating: { $avg: "$rating" },  
 totalFeedbacks: { $sum: 1 }  
 }  
 },  
 { $sort: { averageRating: -1 } }  
])



## 2. List All Feedbacks for a Specific Event

Use: Show all feedback details for a selected event.

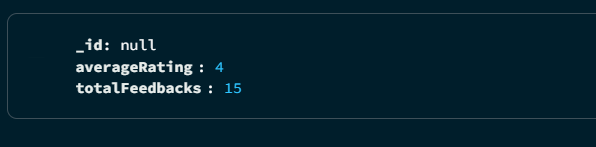
db.feedback.aggregate([  
 {  
 $match: {  
 eventId: "event101"  
 }  
 },  
 {  
 $project: {  
 \_id: 0,  
 guestName: 1,  
 rating: 1,  
 comments: 1  
 }  
 }  
])



## 3. Feedback Count and Average Rating Overall

Use: Summary stats for all feedbacks combined.

db.feedback.aggregate([  
 {  
 $group: {  
 \_id: null,  
 averageRating: { $avg: "$rating" },  
 totalFeedbacks: { $sum: 1 }  
 }  
 }  
])



# 👥 Guests Collection

## 1. Count of Guests per Event

Use: Total number of guests registered for each event.

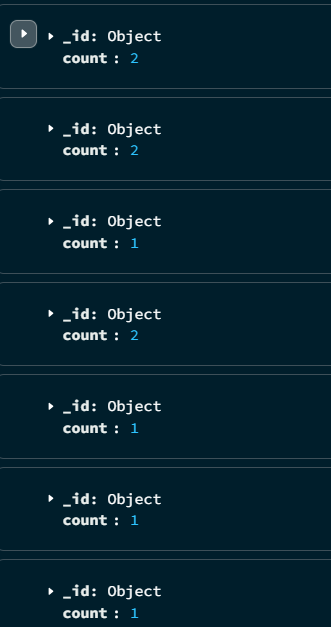
db.guests.aggregate([  
 {  
 $group: {  
 \_id: "$event\_id",  
 guestCount: { $sum: 1 }  
 }  
 },  
 { $sort: { guestCount: -1 } }  
])



## 2. RSVP Status Breakdown per Event

Use: Counts how many guests have Accepted, Declined, or are still Invited.

db.guests.aggregate([  
 {  
 $group: {  
 \_id: { event\_id: "$event\_id", rsvp\_status: "$rsvp\_status" },  
 count: { $sum: 1 }  
 }  
 },  
 { $sort: { "\_id.event\_id": 1 } }  
])



## 3. Check-In Status Report

Use: Find out how many guests checked in vs. not yet checked in.

db.guests.aggregate([  
 {  
 $group: {  
 \_id: "$check\_in\_status",  
 count: { $sum: 1 }  
 }  
 }  
])

