### **Title: MongoDB – Aggregation for Travel App**

### **1. $match**

**Q1:** Get public entries in Japan

js

CopyEdit

db.entries.aggregate([

{ $match: { isPublic: true, "location.name": /Japan/i } }

])

### **2. $project, $addFields**

**Q2:** Add daysSincePosted to entries

js

CopyEdit

db.entries.aggregate([

{

$addFields: {

daysSincePosted: {

$divide: [{ $subtract: [new Date(), "$date"] }, 1000 \* 60 \* 60 \* 24]

}

}

}

])

### **3. $group**

**Q3:** Count how many entries each user posted

js

CopyEdit

db.entries.aggregate([

{ $group: { \_id: "$userId", totalEntries: { $sum: 1 } } }

])

**Q4:** Average countries visited by users

js

CopyEdit

db.users.aggregate([

{ $group: { \_id: null, avgVisited: { $avg: "$travelStats.countriesVisited" } } }

])

### **4. $unwind**

**Q5:** Flatten all challenges for users

js

CopyEdit

db.users.aggregate([

{ $unwind: "$challenges" },

{ $project: { username: 1, "challenges.title": 1, "challenges.status": 1 } }

])

### **5. $sort, $limit**

**Q6:** Top 3 users with highest entries

js

CopyEdit

db.users.aggregate([

{ $sort: { "travelStats.entriesCreated": -1 } },

{ $limit: 3 }

])

### **6. $lookup**

**Q7:** Join entries with user info

js

CopyEdit

db.entries.aggregate([

{

$lookup: {

from: "users",

localField: "userId",

foreignField: "\_id",

as: "userInfo"

}

},

{ $unwind: "$userInfo" }

])

**Q8:** Join users with challenge metadata

js

CopyEdit

db.users.aggregate([

{ $unwind: "$challenges" },

{

$lookup: {

from: "challenges",

localField: "challenges.challengeId",

foreignField: "\_id",

as: "challengeInfo"

}

}

])