**MongoDB – CRUD**

**E-Learning Platform – Gamification Based**

1. **USERS Collection**

**Create:**

db.users.insertOne({

name: "Alice",

email: "alice@elearn.com",

role: "reader",

points: 20,

badge: "Bronze",

createdAt: new Date(),

updatedAt: new Date()

})

**Read** **:**

db.users.find({ role: "reader" })

**Update :**

db.users.updateOne(

{ email: "alice@elearn.com" },

{ $set: { badge: "Silver", updatedAt: new Date() } }

)

**Delete:**

db.users.deleteOne({ email: "alice@elearn.com" })

1. **BLOGS Collection**

**Create:**

db.blogs.insertOne({

title: "Intro to AI",

authorId: ObjectId("..."),

tags: ["AI", "Basics"],

views: 100,

createdAt: new Date(),

updatedAt: new Date()

})

**Read** **:**

db.blogs.find({ views: { $gt: 50 } })

**Update :**

db.blogs.updateOne(

{ title: "Intro to AI" },

{ $inc: { views: 1 }, $set: { updatedAt: new Date() } }

)

**Delete:**

db.blogs.deleteOne({ title: "Intro to AI" })

1. COMMENTS Collection:

**Create:**

db.comments.insertOne({

blogId: ObjectId("..."),

userId: ObjectId("..."),

text: "Great post!",

createdAt: new Date(),

updatedAt: new Date()

})

**Read** **:**

db.comments.find({ blogId: ObjectId("...") })

**Update :**

db.comments.updateOne(

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

{ $set: { text: "Updated comment", updatedAt: new Date() } }

))

**Delete:**

db.comments.deleteOne({ \_id: ObjectId("...") })

**4. LEADERBOARDS Collection**

**Create:**

db.leaderboards.insertOne({

userId: ObjectId("..."),

points: 100,

rank: 1,

createdAt: new Date(),

updatedAt: new Date()

})

**Read** **:**

db.leaderboards.find().sort({ rank: 1 })

**Update :**

db.leaderboards.updateOne(

{ userId: ObjectId("...") },

{ $set: { rank: 2, updatedAt: new Date() } }

)

**Delete:**

db.leaderboards.deleteOne({ userId: ObjectId("...") })

1. **NOTIFICATIONS Collection :**

**Create:**

db.notifications.insertOne({

userId: ObjectId("..."),

type: "comment\_reply",

message: "You got a reply!",

createdAt: new Date()

})

**Read** **:**

db.notifications.find({ userId: ObjectId("...") })

**Update :**

db.notifications.updateOne(

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

{ $set: { message: "Updated message" } }

)

**Delete:**

db.notifications.deleteOne({ \_id: ObjectId("...") })

1. **LEARNING\_PATHS Collection:**

**Create:**

db.learning\_paths.insertOne({

userId: ObjectId("..."),

blogs: [ObjectId("..."), ObjectId("...")],

completed: [],

createdAt: new Date(),

updatedAt: new Date()

})

**Read** **:**

db.learning\_paths.find({ userId: ObjectId("...") })

**Update :**

db.learning\_paths.updateOne(

{ userId: ObjectId("...") },

{ $push: { completed: ObjectId("...") }, $set: { updatedAt: new Date() } }

)

**Delete:**

db.learning\_paths.deleteOne({ userId: ObjectId("...") })

1. **BADGES Collection :**

**Create:**

db.badges.insertOne({

id: "gold",

name: "Gold Badge",

criteria: "80+ points"

})

**Read** **:**

db.badges.find({})

**Update :**

db.badges.updateOne(

{ id: "gold" },

{ $set: { criteria: "90+ points" } }

)

**Delete:**

db.badges.deleteOne({ id: "gold" })

1. **AI\_SUGGESTIONS Collection**

**Create:**

db.ai\_suggestions.insertOne({

userId: ObjectId("..."),

suggestedBlogs: [ObjectId("..."), ObjectId("...")],

createdAt: new Date()

})

**Read** **:**

db.ai\_suggestions.find({ userId: ObjectId("...") })

**Update :**

db.ai\_suggestions.updateOne(

{ userId: ObjectId("...") },

{ $set: { suggestedBlogs: [ObjectId("...")] } }

)

**Delete:**

db.ai\_suggestions.deleteOne({ userId: ObjectId("...") })