MongoDB Relationships Documentation

A MongoDB Relationship refers to the logical connection between documents across different collections or within the same collection, similar to relationships between tables in relational databases. MongoDB stores data as documents (BSON/JSON format), not in rows and columns like SQL. However, relationships between data entities still exist and can be modelled using two primary methods:

1. Embedding (Embedded Documents)

   Embed related data directly inside a document

2. Referencing (Manual References / Foreign Keys)

   Reference related documents across collections using ObjectIds

Types of MongoDB Relationships

- One-to-One (1:1)
- One-to-Many (1:N)
- Many-to-One (N:1)
- Many-to-Many (M:N)

1. **One-to-One (1:1) Relationship**

   One document in a collection is associated with exactly one document in another collection.

   - **Embedding**

     ```
     {
      "_id": 1,
      "username": "john_doe",
      "profile": {
       "bio": "Software Developer",
       "location": "India"
      }
     }
     ```

   - **Referencing**

     ```
     // Users Collection
     {
      "_id": 1,
     ```

```
  "username": "john_doe"
}
// UserProfiles Collection
{
  "_id": 1,
  "user_id": 1,
  "bio": "Software Developer",
  "location": "India"
}
```

2. **One-to-Many (1:N) Relationship**

A single document in one collection relates to multiple documents in another collection.

- **Embedding**

```
{
  "_id": 1,
  "name": "J.K. Rowling",
  "books": [
    { "title": "Harry Potter 1" },
    { "title": "Harry Potter 2" }
  ]
}
```

- **Referencing**

```
{
  "_id": 101,
  "title": "Harry Potter 1",
  "author_id": 1
}
```

3. **Many-to-One (N:1) Relationship**

Multiple documents in a collection relate to one document in another collection. It’s the inverse of One-to-Many.

```
//Authors Collection
{
  "_id": 1,
  "name": "J.K. Rowling"
}
```

//Books Collection with author reference)

```
{
  "_id": 101,
  "title": "Harry Potter 1",
  "author_id": 1
},
{
  "_id": 102,
  "title": "Harry Potter 2",
  "author_id": 1
}
```

4. **Many-to-Many (M:N) Relationship**

Many documents in one collection relate to many documents in another collection. Both sides can have multiple links to each other.

Array of References

//Students Collection:

```
{
  "_id": 1,
  "name": "Alice",
  "enrolled_courses": [101, 102]
}
```

//Courses Collection

```
{
  "_id": 101,
  "course_name": "Mathematics"
},
{
  "_id": 102,
```

"course_name": "Science"

}

Join (Linking) Collection Approach

//Students Collection

{ "_id": 1, "name": "Alice" }

{ "_id": 2, "name": "Bob" }

//Courses Collection

{ "_id": 101, "course_name": "Math" }

{ "_id": 102, "course_name": "Science" }

//Enrollments Collection (Join Table)

{ "_id": 1, "student_id": 1, "course_id": 101 }

{ "_id": 2, "student_id": 1, "course_id": 102 }

{ "_id": 3, "student_id": 2, "course_id": 102 }