

Full Stack Lab

Experiment 8

Name:- Anvesha Raikwar
PRN:- 2107012617
Batch:- AIML A1

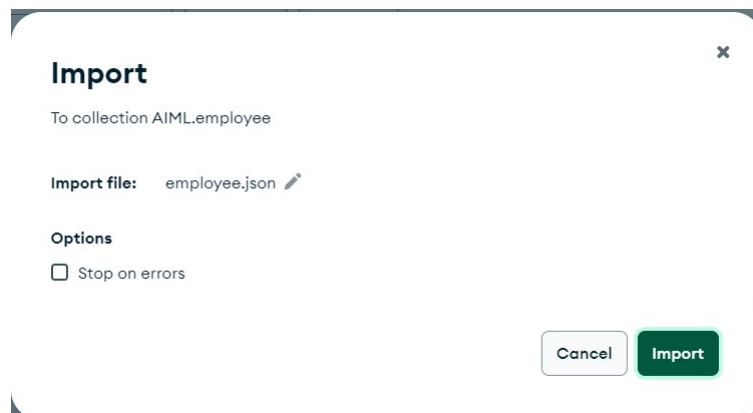
Objective: Create a database from .json file and execute NO SQL Queries.

1. Create Database: Create a MongoDB database named AIML.
2. Create Collection: Create a collection named employee within the AIML database.

```
Currently connected to localhost:27017 with default data
// Select the database to use
use('AIML');

💡 Create collection employee
db.createCollection("employee")
```

3. Import Employee Records: Import employee records from a JSON file into the employee collection. Save the following data in a json file and use it for import in MongoDB.



4. insertOne: Inserts a single document into the collection.
{ "Id": 21, "Name": "John Doe", "Project_id": 2, "Hrs_worked": 35 }
5. insertMany: Inserts multiple documents into the collection.
{ "Id": 22, "Name": "Jane Smith", "Project_id": 1, "Hrs_worked": 28 },
{ "Id": 23, "Name": "Alice Johnson", "Project_id": 3, "Hrs_worked": 42 }

```

//InsertOne: Insert a single document into the collection.
db.employee.insertOne({ "Id": 21, "Name": "John Doe", "Project_id": 2, "Hrs_worked": 35 })

//InsertMany: Insert multiple documents into the collection.
db.employee.insertMany([
  { "Id": 22, "Name": "Jane Smith", "Project_id": 1, "Hrs_worked": 28 },
  { "Id": 23, "Name": "Alice Johnson", "Project_id": 3, "Hrs_worked": 42 }
])

```

```

{} Playground Result X
1  {
2    "acknowledged": true,
3    "insertedIds": {
4      "0": {
5        "$oid": "661b737f05e952f39549c576"
6      },
7      "1": {
8        "$oid": "661b737f05e952f39549c577"
9      }
10   }
11 }

```

6. updateOne: Updates a single document that matches the filter.

```
{ "Id": 21 }, { $set: { "Hrs_worked": 40 } }
```

7. updateMany: Updates multiple documents that match the filter.

```
{ "Hrs_worked": { $gt: 30 } }, { $set: { "Overtime": true } }
```

```

//UpdateOne: Update a single document that matches the filter.
db.employee.updateOne({ "Id": 21 }, { $set: { "Hrs_worked": 40 } })

//UpdateMany: Update multiple documents that match the filter.
db.employee.updateMany({ "Hrs_worked": { $gt: 30 } }, { $set: { "Overtime": true } })

```

```
{ Playground Result X
1 {
2   "acknowledged": true,
3   "insertedId": null,
4   "matchedCount": 15,
5   "modifiedCount": 15,
6   "upsertedCount": 0
7 }
```

8. find an employee by their ID.

```
{ Playground Result X
1 {
2   "_id": {
3     "$oid": "661b737f05e952f39549c575"
4   },
5   "Id": 21,
6   "Name": "John Doe",
7   "Project_id": 2,
8   "Hrs_worked": 40,
9   "Overtime": true
10 }
```

9. How would you retrieve all employees who are assigned to a specific project ID?

```
{ Playground Result X
1 {
2   Edit Document
3     "_id": {
4       "$oid": "661b72ec35caa421e60241c1"
5     },
6     "Id": 2,
7     "Name": "Frank Smith",
8     "Age": 36,
9     "Gender": "Female",
10    "Project_id": 1,
11    "Hrs_worked": 12
12  },
13  {
14    Edit Document
15      "_id": {
16        "$oid": "661b72ec35caa421e60241cc"
17      },
18      "Id": 13,
19      "Name": "Emma Jones",
20      "Age": 52,
21      "Gender": "Male",
22      "Project_id": 1,
23      "Hrs_worked": 33,
24      "Overtime": true
25    },
26    Edit Document
27      "_id": {
28        "$oid": "661b72ec35caa421e60241cf"
29      },
30      "Id": 16,
31      "Name": "Alice Moore",
32      "Age": 29,
33      "Gender": "Male",
34      "Project_id": 1,
35      "Hrs_worked": 39,
36      "Overtime": true
37    },
38  },
39  {
40    Edit Document
41      "_id": {
42        "$oid": "661b72ec35caa421e60241d2"
43      },
44      "Id": 19,
45      "Name": "Charlie Williams",
46      "Age": 59,
47      "Gender": "Female",
48      "Project_id": 1,
49      "Hrs_worked": 31,
50      "Overtime": true
51    },
52  },
53  {
54    Edit Document
55      "_id": {
56        "$oid": "661b737f05e952f39549c576"
57      },
58      "Id": 22,
59      "Name": "Jane Smith",
60      "Project_id": 1,
61      "Hrs_worked": 28
62    },
63  },
64  {
65    Edit Document
66      "_id": {
67        "$oid": "661b737f05e952f39549c575"
68      },
69      "Id": 21,
70      "Name": "John Doe",
71      "Project_id": 2,
72      "Hrs_worked": 40,
73      "Overtime": true
74    },
75  },
76  {
77    Edit Document
78      "_id": {
79        "$oid": "661b737f05e952f39549c575"
80      },
81      "Id": 21,
82      "Name": "John Doe",
83      "Project_id": 2,
84      "Hrs_worked": 40,
85      "Overtime": true
86    },
87  },
88  {
89    Edit Document
90      "_id": {
91        "$oid": "661b737f05e952f39549c575"
92      },
93      "Id": 21,
94      "Name": "John Doe",
95      "Project_id": 2,
96      "Hrs_worked": 40,
97      "Overtime": true
98    },
99  },
100 }
```

10. Write a query to find employees who have worked more than 30 hours.

```
1  [
2
3  {
4    "_id": {
5      "$oid": "661b72ec35caa421e60241c2"
6    },
7    "Id": 3,
8    "Name": "Hannah Brown",
9    "Age": 33,
10   "Gender": "Female",
11   "Project_id": 2,
12   "Hrs_worked": 36,
13   "Overtime": true
14 },
15 {
16   "_id": {
17     "$oid": "661b72ec35caa421e60241c7"
18   },
19   "Id": 8,
20   "Name": "David Williams",
21   "Age": 60,
22   "Gender": "Male",
23   "Project_id": 2,
24   "Hrs_worked": 34,
25   "Overtime": true
26 }
```

11. Can you demonstrate how to use the \$gt operator to find employees who are older than 40?

```
1  [
2
3  {
4    "_id": {
5      "$oid": "661b72ec35caa421e60241c0"
6    },
7    "Id": 1,
8    "Name": "Charlie Moore",
9    "Age": 44,
10   "Gender": "Male",
11   "Project_id": 4,
12   "Hrs_worked": 12
13 },
14 {
15   "_id": {
16     "$oid": "661b72ec35caa421e60241c4"
17   },
18   "Id": 5,
19   "Name": "Bob Davis",
20   "Age": 41,
21   "Gender": "Female",
22   "Project_id": 4,
23   "Hrs_worked": 16
24 }
```

12. Explain the purpose of sorting in MongoDB queries.

Sorting in MongoDB queries adds flexibility and control over the order in which documents are returned, making it easier to work with data and extract valuable insights from your MongoDB collections.

13. Sort the Employee table based on Age in Ascending order and display.

```
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241cd"
  },
  "Id": 14,
  "Name": "Isaac Wilson",
  "Age": 21,
  "Gender": "Female",
  "Project_id": 3,
  "Hrs_worked": 32,
  "Overtime": true
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241d0"
  },
  "Id": 17,
  "Name": "David Miller",
  "Age": 25,
  "Gender": "Female",
  "Project_id": 4,
  "Hrs_worked": 32,
  "Overtime": true
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241cf"
  },
  "Id": 16,
  "Name": "Alice Moore",
  "Age": 29,
  "Gender": "Male",
  "Project_id": 1,
  "Hrs_worked": 39,
  "Overtime": true
}
```

14. Sort the Employee table based on Hrs_worked in Descending order and display.

```
Playground Result X
1  [
2    {
3      "_id": {
4        "$oid": "661b737f05e952f39549c577"
5      },
6      "Id": 23,
7      "Name": "Alice Johnson",
8      "Project_id": 3,
9      "Hrs_worked": 42,
10     "Overtime": true
11   },
12   {
13     "_id": {
14       "$oid": "661b737f05e952f39549c575"
15     },
16     "Id": 21,
17     "Name": "John Doe",
18     "Project_id": 2,
19     "Hrs_worked": 40,
20     "Overtime": true
21   },
22   {
23     "_id": {
24       "$oid": "661b72ec35caa421e60241cf"
25     },
26     "Id": 16,
27     "Name": "Alice Moore",
28     "Age": 29,
29     "Gender": "Male",
30     "Project_id": 1,
31     "Hrs_worked": 39,
32     "Overtime": true
33   }
34 ]
```

15. Find Employee whose age is greater then 30 and Has_Worked greater then 20.

```
[
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c2"
    },
    "Id": 3,
    "Name": "Hannah Brown",
    "Age": 33,
    "Gender": "Female",
    "Project_id": 2,
    "Hrs_worked": 36,
    "Overtime": true
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c3"
    },
    "Id": 4,
    "Name": "Hannah Davis",
    "Age": 31,
    "Gender": "Female",
    "Project_id": 2,
    "Hrs_worked": 22
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c5"
    },
    "Id": 6,
    "Name": "Hannah Davis",
    "Age": 57,
    "Gender": "Male",
    "Project_id": 2,
    "Hrs_worked": 28
  }
]
```

16. Find Employee whose Gender is Male or Has_Worked greater then 25.

```
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c2"
  },
  "Id": 3,
  "Name": "Hannah Brown",
  "Age": 33,
  "Gender": "Female",
  "Project_id": 2,
  "Hrs_worked": 36,
  "Overtime": true
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c5"
  },
  "Id": 6,
  "Name": "Hannah Davis",
  "Age": 57,
  "Gender": "Male",
  "Project_id": 2,
  "Hrs_worked": 28
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c6"
  },
  "Id": 7,
  "Name": "Frank Brown",
  "Age": 49,
  "Gender": "Male",
  "Project_id": 4,
  "Hrs_worked": 30
}
```

17. Find Employee whose Project_id is not 3.

```
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c1"
  },
  "Id": 2,
  "Name": "Frank Smith",
  "Age": 36,
  "Gender": "Female",
  "Project_id": 1,
  "Hrs_worked": 12
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c2"
  },
  "Id": 3,
  "Name": "Hannah Brown",
  "Age": 33,
  "Gender": "Female",
  "Project_id": 2,
  "Hrs_worked": 36,
  "Overtime": true
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c3"
  },
  "Id": 4,
  "Name": "Hannah Davis",
  "Age": 31,
  "Gender": "Female",
  "Project_id": 2,
  "Hrs_worked": 22
}
```

18. Write a MongoDB query to find all employees who are between the ages of 25 and 35.

```
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c2"
  },
  "Id": 3,
  "Name": "Hannah Brown",
  "Age": 33,
  "Gender": "Female",
  "Project_id": 2,
  "Hrs_worked": 36,
  "Overtime": true
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241c3"
  },
  "Id": 4,
  "Name": "Hannah Davis",
  "Age": 31,
  "Gender": "Female",
  "Project_id": 2,
  "Hrs_worked": 22
},
{
  "_id": {
    "$oid": "661b72ec35caa421e60241cb"
  },
  "Id": 12,
  "Name": "Isaac Smith",
  "Age": 32,
  "Gender": "Male",
  "Project_id": 2,
  "Hrs_worked": 23
}
```

19. How would you retrieve employees who have worked between 20 and 30 hours?

```
[
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c3"
    },
    "Id": 4,
    "Name": "Hannah Davis",
    "Age": 31,
    "Gender": "Female",
    "Project_id": 2,
    "Hrs_worked": 22
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c5"
    },
    "Id": 6,
    "Name": "Hannah Davis",
    "Age": 57,
    "Gender": "Male",
    "Project_id": 2,
    "Hrs_worked": 28
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241cb"
    },
    "Id": 12,
    "Name": "Isaac Smith",
    "Age": 32,
    "Gender": "Male",
    "Project_id": 2,
    "Hrs_worked": 23
  },
]
```

20. Write a query to find employees who are either working on Project 1 or Project 2.

```
[
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c1"
    },
    "Id": 2,
    "Name": "Frank Smith",
    "Age": 36,
    "Gender": "Female",
    "Project_id": 1,
    "Hrs_worked": 12
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c2"
    },
    "Id": 3,
    "Name": "Hannah Brown",
    "Age": 33,
    "Gender": "Female",
    "Project_id": 2,
    "Hrs_worked": 36,
    "Overtime": true
  },
  {
    "_id": {
      "$oid": "661b72ec35caa421e60241c3"
    },
    "Id": 4,
    "Name": "Hannah Davis",
    "Age": 31,
    "Gender": "Female",
    "Project_id": 2,
    "Hrs_worked": 22
  },
]
```



```
//Find by ID:
db.employee.findOne({ "Id": 21 })

//Retrieve employees by Project ID:
db.employee.find({ "Project_id": 1 })

//Retrieve employees who worked more than 30 hours:
db.employee.find({ "Hrs_worked": { $gt: 30 } })

//Retrieve employees older than 40:
db.employee.find({ "Age": { $gt: 40 } })

//Sorting in MongoDB:
// Sort by Age in Ascending order
db.employee.find().sort({ "Age": 1 })

// Sort by Hrs_worked in Descending order
db.employee.find().sort({ "Hrs_worked": -1 })

//Additional Queries
// Find employees older than 30 and worked more than 20 hours
db.employee.find({ "Age": { $gt: 30 }, "Hrs_worked": { $gt: 20 } })

// Find employees who are Male or worked more than 25 hours
db.employee.find({ $or: [{ "Gender": "Male" }, { "Hrs_worked": { $gt: 25 } }] })

// Find employees whose Project_id is not 3
db.employee.find({ "Project_id": { $ne: 3 } })

// Find employees between the ages of 25 and 35
db.employee.find({ "Age": { $gte: 25, $lte: 35 } })

// Find employees who worked between 20 and 30 hours
db.employee.find({ "Hrs_worked": { $gt: 20, $lt: 30 } })

// Find employees working on Project 1 or Project 2
db.employee.find({ "Project_id": { $in: [1, 2] } })
```