**Practical no :10**

**Aggregation operations**

Microsoft Windows [Version 10.0.26100.6584]

(c) Microsoft Corporation. All rights reserved.

C:\Users\HP\Downloads\mongosh-2.5.8-win32-x64\mongosh-2.5.8-win32-x64\bin>mongosh

Current Mongosh Log ID: 68e5c400fdaa19bae8cebea3

Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.8

Using MongoDB: 8.2.0

Using Mongosh: 2.5.8

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).

You can opt-out by running the disableTelemetry() command.

------

The server generated these startup warnings when booting

2025-10-08T07:04:48.990+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

------

test> show db;

MongoshInvalidInputError: [COMMON-10001] 'db' is not a valid argument for "show".

test> show dbs;

admin 40.00 KiB

config 36.00 KiB

demo 68.00 KiB

local 80.00 KiB

test 40.00 KiB

test> use bank;

switched to db bank

bank> db.createCollection('customer');

{ ok: 1 }

bank> db.customer.insertOne()

MongoshInvalidInputError: [COMMON-10001] Missing required argument at position 0 (Collection.insertOne)

bank> db.customer.insertOne({\_id:101,cname:'priya',bname:'SBI',branch:'pune',bal:80000});

{ acknowledged: true, insertedId: 101 }

bank> db.customer.insertOne({\_id:102,cname:'riya',bname:'SBI',branch:'mumbai',bal:90000});

{ acknowledged: true, insertedId: 102 }

bank> db.customer.insertOne({\_id:103,cname:'siya',bname:'Axis',branch:'umerga',bal:100000});

{ acknowledged: true, insertedId: 103 }

bank> db.customer.insertOne({\_id:104,cname:'miya',bname:'Axis',branch:'pune',bal:2000});

{ acknowledged: true, insertedId: 104 }

bank> db.customer.count();

DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.

4

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$sum:'$bal'}}}]);

[ { \_id: null, total\_bal: 272000 } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$max:'$bal'}}}]);

[ { \_id: null, total\_bal: 100000 } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$min:'$bal'}}}]);

[ { \_id: null, total\_bal: 2000 } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$avg:'$bal'}}}]);

[ { \_id: null, total\_bal: 68000 } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$push:'$bal'}}}]);

[ { \_id: null, total\_bal: [ 80000, 90000, 100000, 2000 ] } ]

bank> db.customer.insertOne({\_id:104,cname:'diya',bname:'Axis',branch:'pune',bal:2000});

MongoServerError: E11000 duplicate key error collection: bank.customer index: \_id\_ dup key: { \_id: 104 }

bank> db.customer.insertOne({\_id:105,cname:'diya',bname:'Axis',branch:'pune',bal:2000});

{ acknowledged: true, insertedId: 105 }

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$push:'$bal'}}}]);

[ { \_id: null, total\_bal: [ 80000, 90000, 100000, 2000, 2000 ] } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$addToSet:'$bal'}}}]);

[ { \_id: null, total\_bal: [ 2000, 100000, 80000, 90000 ] } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$first:'$bal'}}}]);

[ { \_id: null, total\_bal: 80000 } ]

bank> db.customer.aggregate([{$group:{\_id:null,total\_bal:{$last:'$bal'}}}]);

[ { \_id: null, total\_bal: 2000 } ]

bank> db.customer.find().sort({name:1});

[

{

\_id: 101,

cname: 'priya',

bname: 'SBI',

branch: 'pune',

bal: 80000

},

{

\_id: 102,

cname: 'riya',

bname: 'SBI',

branch: 'mumbai',

bal: 90000

},

{

\_id: 103,

cname: 'siya',

bname: 'Axis',

branch: 'umerga',

bal: 100000

},

{ \_id: 104, cname: 'miya', bname: 'Axis', branch: 'pune', bal: 2000 },

{ \_id: 105, cname: 'diya', bname: 'Axis', branch: 'pune', bal: 2000 }

]

bank> db.customer.find().sort({name:-1});

[

{

\_id: 101,

cname: 'priya',

bname: 'SBI',

branch: 'pune',

bal: 80000

},

{

\_id: 102,

cname: 'riya',

bname: 'SBI',

branch: 'mumbai',

bal: 90000

},

{

\_id: 103,

cname: 'siya',

bname: 'Axis',

branch: 'umerga',

bal: 100000

},

{ \_id: 104, cname: 'miya', bname: 'Axis', branch: 'pune', bal: 2000 },

{ \_id: 105, cname: 'diya', bname: 'Axis', branch: 'pune', bal: 2000 }

]

bank> db.customer.createIndex({\_id:1});

\_id\_1

bank> db.customer.createIndex({cname:1});

cname\_1

bank> db.customer.createIndex({branch:1});

branch\_1

bank> db.customer.createIndex({bname:1});

bname\_1

bank> db.customer.createIndex({bal:1});

bal\_1

bank> db.customer.getIndexes();

[

{ v: 2, key: { \_id: 1 }, name: '\_id\_' },

{ v: 2, key: { cname: 1 }, name: 'cname\_1' },

{ v: 2, key: { branch: 1 }, name: 'branch\_1' },

{ v: 2, key: { bname: 1 }, name: 'bname\_1' },

{ v: 2, key: { bal: 1 }, name: 'bal\_1' }

]