> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> use pms

switched to db pms

> db

pms

> show collections

products

> db.products.find();

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45, "productQty" : 4, "productBrand" : "Reynolds" }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

> db.products.insertMany([

... {"\_id" : 11,"productName" : "Pencil","productPrice" : 30,"productQty" : 10,"productBrand" : "Apsara"},

... {"\_id" : 12,"productName" : "Pencil","productPrice" : 30,"productQty" : 10,"productBrand" : "Natraj"},

... {"\_id" :13,"productName" : "Eraser","productPrice" : 20,"productQty" : 10,"productBrand" : "Natraj"}

... ]);

{ "acknowledged" : true, "insertedIds" : [ 11, 12, 13 ] }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

> db.products.find({productName:"Pencil"}).pretty();

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

> db.products.find({productName:"Pencil"},{\_id:0,productQty:0}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productBrand" : "Apsara" }

{ "productName" : "Pencil", "productPrice" : 30, "productBrand" : "Apsara" }

{ "productName" : "Pencil", "productPrice" : 30, "productBrand" : "Natraj" }

> db.products.find({productName:"Pencil"},{\_id:0,productQty:1}).pretty();

{ "productQty" : 10 }

{ "productQty" : 10 }

{ "productQty" : 10 }

> db.products.find({productName:"Pencil"},{productName:1,productPrice:1,productQty:1}).pretty();

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10

}

> db.products.find({\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

> db.products.find({},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pen", "productPrice" : 45, "productQty" : 4 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({price:{$gt:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

> db.products.find({productPrice:{$gt:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pen", "productPrice" : 45, "productQty" : 4 }

> db.products.find({productPrice:{$lt:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({productPrice:{$lte:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({productPrice:{$gte:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pen", "productPrice" : 45, "productQty" : 4 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

> db.products.find({productPrice:{$eq:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

> db.products.find({productPrice:{$ne:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pen", "productPrice" : 45, "productQty" : 4 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({productPrice:{$in:[30,20]}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({productPrice:{$nin:[30,20]}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pen", "productPrice" : 45, "productQty" : 4 }

> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> use dummy

switched to db dummy

> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> db

dummy

> db.movies

dummy.movies

> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> db.movies.insertOne({ "movieName" : "Thor"})

{

"acknowledged" : true,

"insertedId" : ObjectId("6342db7855163092c0b11335")

}

> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

dummy 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> db.dropDatabase();

{ "ok" : 1 }

> show dbs

admin 0.000GB

bms 0.000GB

bookstore 0.000GB

cafeshop 0.000GB

config 0.000GB

expresscrud 0.000GB

local 0.000GB

meandb 0.000GB

mongodbexample 0.000GB

onlineecommerceappliation 0.000GB

pms 0.000GB

react-jwt 0.000GB

shoppinglist 0.000GB

test 0.000GB

todos 0.000GB

> use pms

switched to db pms

> db

pms

> show collections

products

> db.datadummy

pms.datadummy

> db.datadummy.insertOne({});

{

"acknowledged" : true,

"insertedId" : ObjectId("6342dbdc55163092c0b11336")

}

> show collections

datadummy

products

> db.datadummy.drop();

true

> show collections

products

>

Logical Query Operators

> db.products.find({ $and:[{productPrice : 30 }, { productName : "Pencil" } ] },{productName :1,productPrice:1});

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

db.products.find({ $or:[{productPrice : 30 }, { productName : "Pencil" } ] },{productName :1,productPrice:1});

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

> db.products.find({ $and:[{productPrice : 30 }, { productName : "Pencil" },{productQty:10} ] },{productName :1,productPrice:1});

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

> db.products.find({ $or:[{productPrice : 30 }, { productName : "Pencil" },{productQty:10} ] },{productName :1,productPrice:1});

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

>db.products.find({productPrice:{$not:{$gt:30}}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

{ "productName" : "NoteBook", "productQty" : 15 }

> db.products.find({productPrice:{$lte:30}},{\_id:0,productName:1,productPrice:1,productQty:1}).pretty();

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Pencil", "productPrice" : 30, "productQty" : 10 }

{ "productName" : "Eraser", "productPrice" : 20, "productQty" : 10 }

> db.products.find({ $or:[{productPrice : 30 }, { productName : "Pencil" },{productQty:10} ] },{productName :1,productPrice:1});

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

> db.products.find({ $nor:[{productPrice : 30 }, { productName : "Pencil" },{productQty:10} ] },{productName :1,productPrice:1});

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 24, "productName" : "NoteBook" }

> db.products.find( { productPrice:{ $exists : true } }, { productName : 1, productPrice :1 } );

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

> db.products.find( { productPrice:{ $exists : true,$gt:30 } }, { productName : 1, productPrice :1 } );

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

> db.products.find( { productPrice:{ $exists : false } }, { productName : 1, productPrice :1 } );

{ "\_id" : 24, "productName" : "NoteBook" }

{ "\_id" : 25, "productName" : "Book" }

>db.products.find( { productPrice:{ $exists : true,$lt:30 } }, { productName : 1, productPrice :1 } );

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

> db.products.find( { productPrice:{ $exists : true,$lte:30 } }, { productName : 1, productPrice :1 } );

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

> db.products.find( { productPrice:{ $type : "string" } }, { productName : 1, productPrice :1 } );

> db.products.find( { productPrice:{ $type : "number" } }, { productName : 1, productPrice :1 } );

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

> db.products.insertOne({ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [95,205], "productQty" : 15, "productBrand" : "Classmate", "productMfgDate" : ISODate("2021-05-14T00:00:00Z"),"productExpDate" : ISODate("2023-05-14T00:00:00Z") });

{ "acknowledged" : true, "insertedId" : 45 }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z")

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 25,

"productName" : "Book",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

> db.products.find( { productPrice:{ $size : 2 } }, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

> db.products.insertOne({ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [195,405], "productQty" : 15, "productBrand" : "Classmate", "productMfgDate" : ISODate("2021-05-14T00:00:00Z"),"productExpDate" : ISODate("2023-05-14T00:00:00Z") });

{ "acknowledged" : true, "insertedId" : 55 }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z")

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 25,

"productName" : "Book",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 55,

"productName" : "NoteBook",

"productPrice" : [

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

> db.products.find( { productPrice:{ $size : 2 } }, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

> db.products.insertOne({ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [95,195,405], "productQty" : 15, "productBrand" : "Classmate", "productMfgDate" : ISODate("2021-05-14T00:00:00Z"),"productExpDate" : ISODate("2023-05-14T00:00:00Z") });

{ "acknowledged" : true, "insertedId" : 65 }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z")

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 25,

"productName" : "Book",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 55,

"productName" : "NoteBook",

"productPrice" : [

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 65,

"productName" : "NoteBook",

"productPrice" : [

95,

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

> db.products.find( { productPrice:{ $size : 2 } }, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

> db.products.find({ $or: [{ productPrice:{ $size : 2 }},{ productPrice:{ $size : 3 } }]}, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

> db.products.find( { productPrice:{ $all : [30,20] } }, { productName : 1, productPrice :1 } );

> db.products.find( { productPrice:{ $all : [95,205] } }, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

> db.products.find( { productPrice:{ $all : [95,105] } }, { productName : 1, productPrice :1 } );

> db.products.find( { productPrice:{ $all : [95,195] } }, { productName : 1, productPrice :1 } );

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

> db.products.find( { productPrice:{ $elemMatch : {$lt : 95} } }, { productName : 1, productPrice :1 } );

> db.products.find( { productPrice:{ $elemMatch : {$lt : 195} } }, { productName : 1, productPrice :1 } );

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:1});

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:-1});

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:-1}).limit(2);

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:-1}).limit(5);

{ "\_id" : 65, "productName" : "NoteBook", "productPrice" : [ 95, 195, 405 ] }

{ "\_id" : 55, "productName" : "NoteBook", "productPrice" : [ 195, 405 ] }

{ "\_id" : 45, "productName" : "NoteBook", "productPrice" : [ 95, 205 ] }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:1}).limit(5);

{ "\_id" : 13, "productName" : "Eraser", "productPrice" : 20 }

{ "\_id" : 1, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

> db.products.find( { productPrice:{ $exists :1 } }, { productName : 1, productPrice :1 } ).sort({productPrice:1}).skip(2).limit(5);

{ "\_id" : 12, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : 11, "productName" : "Pencil", "productPrice" : 30 }

{ "\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"), "productName" : "Pen", "productPrice" : 45 }

{ "\_id" : 21, "productName" : "NoteBook", "productPrice" : 95 }

{ "\_id" : 22, "productName" : "NoteBook", "productPrice" : 95 }

> db.products.aggregate([ { $group:{ \_id : '$productName', averageQty : { $avg : '$productQty'}, }, }, ]);

{ "\_id" : "Eraser", "averageQty" : 10 }

{ "\_id" : "NoteBook", "averageQty" : 15 }

{ "\_id" : "Book", "averageQty" : 15 }

{ "\_id" : "Pencil", "averageQty" : 10 }

{ "\_id" : "Pen", "averageQty" : 4 }

> db.employees

pms.employees

> db.employees.insertMany(

... [

... {\_id:1,empName:"Rama",empSal:45000,empDept:"HR"},

... {\_id:2,empName:"Raja",empSal:35000,empDept:"HR"},

... {\_id:3,empName:"Rags",empSal:45000,empDept:"Admin"},

... {\_id:4,empName:"Ratna",empSal:55000,empDept:"Accounts"},

... {\_id:5,empName:"Rita",empSal:25000,empDept:"IT"}

... ]

... );

{ "acknowledged" : true, "insertedIds" : [ 1, 2, 3, 4, 5 ] }

> db.employees.find().pretty();

{ "\_id" : 1, "empName" : "Rama", "empSal" : 45000, "empDept" : "HR" }

{ "\_id" : 2, "empName" : "Raja", "empSal" : 35000, "empDept" : "HR" }

{ "\_id" : 3, "empName" : "Rags", "empSal" : 45000, "empDept" : "Admin" }

{

"\_id" : 4,

"empName" : "Ratna",

"empSal" : 55000,

"empDept" : "Accounts"

}

{ "\_id" : 5, "empName" : "Rita", "empSal" : 25000, "empDept" : "IT" }

> db.employees.aggregate([ { $group:{ \_id : '$empDept', averageSalary : { $avg : '$empSal'}, }, }, ]);

{ "\_id" : "Accounts", "averageSalary" : 55000 }

{ "\_id" : "HR", "averageSalary" : 40000 }

{ "\_id" : "Admin", "averageSalary" : 45000 }

{ "\_id" : "IT", "averageSalary" : 25000 }

>

> db.employees.aggregate([ { $group:{ \_id : null, maxSalary : { $max : '$empSal'}, }, }, ]);

{ "\_id" : null, "maxSalary" : 55000 }

> db.employees.aggregate([ { $group:{ \_id : null, minSalary : { $min : '$empSal'}, }, }, ]);

{ "\_id" : null, "minSalary" : 25000 }

> db.employees.updateOne({\_id:5},{$mul:{empSal:1.5}})

{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

> db.employees.find().pretty();

{ "\_id" : 1, "empName" : "Rama", "empSal" : 45000, "empDept" : "HR" }

{ "\_id" : 2, "empName" : "Raja", "empSal" : 35000, "empDept" : "HR" }

{ "\_id" : 3, "empName" : "Rags", "empSal" : 45000, "empDept" : "Admin" }

{

"\_id" : 4,

"empName" : "Ratna",

"empSal" : 55000,

"empDept" : "Accounts"

}

{ "\_id" : 5, "empName" : "Rita", "empSal" : 37500, "empDept" : "IT" }

> db.employees.updateMany({},{$mul:{empSal:1.5}})

{ "acknowledged" : true, "matchedCount" : 5, "modifiedCount" : 5 }

> db.employees.find().pretty();

{ "\_id" : 1, "empName" : "Rama", "empSal" : 67500, "empDept" : "HR" }

{ "\_id" : 2, "empName" : "Raja", "empSal" : 52500, "empDept" : "HR" }

{ "\_id" : 3, "empName" : "Rags", "empSal" : 67500, "empDept" : "Admin" }

{

"\_id" : 4,

"empName" : "Ratna",

"empSal" : 82500,

"empDept" : "Accounts"

}

{ "\_id" : 5, "empName" : "Rita", "empSal" : 56250, "empDept" : "IT" }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z")

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 25,

"productName" : "Book",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 55,

"productName" : "NoteBook",

"productPrice" : [

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 65,

"productName" : "NoteBook",

"productPrice" : [

95,

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

> db.products.updateOne({\_id:65},{$unset:{productPrice:""}})

{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productQty" : 4,

"productBrand" : "Reynolds"

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Apsara"

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productQty" : 10,

"productBrand" : "Natraj"

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z")

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 25,

"productName" : "Book",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 55,

"productName" : "NoteBook",

"productPrice" : [

195,

405

],

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

{

"\_id" : 65,

"productName" : "NoteBook",

"productQty" : 15,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z")

}

> db.products.updateMany({},{$rename:{productQty:"productQuantity"}})

{ "acknowledged" : true, "matchedCount" : 12, "modifiedCount" : 12 }

> db.products.find().pretty();

{

"\_id" : ObjectId("6342d05ca1c738d0dc8e0f64"),

"productName" : "Pen",

"productPrice" : 45,

"productBrand" : "Reynolds",

"productQuantity" : 4

}

{

"\_id" : 1,

"productName" : "Pencil",

"productPrice" : 30,

"productBrand" : "Apsara",

"productQuantity" : 10

}

{

"\_id" : 11,

"productName" : "Pencil",

"productPrice" : 30,

"productBrand" : "Apsara",

"productQuantity" : 10

}

{

"\_id" : 12,

"productName" : "Pencil",

"productPrice" : 30,

"productBrand" : "Natraj",

"productQuantity" : 10

}

{

"\_id" : 13,

"productName" : "Eraser",

"productPrice" : 20,

"productBrand" : "Natraj",

"productQuantity" : 10

}

{

"\_id" : 21,

"productName" : "NoteBook",

"productPrice" : 95,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 22,

"productName" : "NoteBook",

"productPrice" : 95,

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 24,

"productName" : "NoteBook",

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 25,

"productName" : "Book",

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 45,

"productName" : "NoteBook",

"productPrice" : [

95,

205

],

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 55,

"productName" : "NoteBook",

"productPrice" : [

195,

405

],

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

{

"\_id" : 65,

"productName" : "NoteBook",

"productBrand" : "Classmate",

"productMfgDate" : ISODate("2021-05-14T00:00:00Z"),

"productExpDate" : ISODate("2023-05-14T00:00:00Z"),

"productQuantity" : 15

}

> db.products.find({productName:'Pencil'}).explain('executionStats')

{

"explainVersion" : "1",

"queryPlanner" : {

"namespace" : "pms.products",

"indexFilterSet" : false,

"parsedQuery" : {

"productName" : {

"$eq" : "Pencil"

}

},

"maxIndexedOrSolutionsReached" : false,

"maxIndexedAndSolutionsReached" : false,

"maxScansToExplodeReached" : false,

"winningPlan" : {

"stage" : "COLLSCAN",

"filter" : {

"productName" : {

"$eq" : "Pencil"

}

},

"direction" : "forward"

},

"rejectedPlans" : [ ]

},

"executionStats" : {

"executionSuccess" : true,

"nReturned" : 3,

"executionTimeMillis" : 0,

"totalKeysExamined" : 0,

"totalDocsExamined" : 12,

"executionStages" : {

"stage" : "COLLSCAN",

"filter" : {

"productName" : {

"$eq" : "Pencil"

}

},

"nReturned" : 3,

"executionTimeMillisEstimate" : 0,

"works" : 14,

"advanced" : 3,

"needTime" : 10,

"needYield" : 0,

"saveState" : 0,

"restoreState" : 0,

"isEOF" : 1,

"direction" : "forward",

"docsExamined" : 12

}

},

"command" : {

"find" : "products",

"filter" : {

"productName" : "Pencil"

},

"$db" : "pms"

},

"serverInfo" : {

"host" : "DESKTOP-40F6E3R",

"port" : 27017,

"version" : "5.0.8",

"gitVersion" : "c87e1c23421bf79614baf500fda6622bd90f674e"

},

"serverParameters" : {

"internalQueryFacetBufferSizeBytes" : 104857600,

"internalQueryFacetMaxOutputDocSizeBytes" : 104857600,

"internalLookupStageIntermediateDocumentMaxSizeBytes" : 104857600,

"internalDocumentSourceGroupMaxMemoryBytes" : 104857600,

"internalQueryMaxBlockingSortMemoryUsageBytes" : 104857600,

"internalQueryProhibitBlockingMergeOnMongoS" : 0,

"internalQueryMaxAddToSetBytes" : 104857600,

"internalDocumentSourceSetWindowFieldsMaxMemoryBytes" : 104857600

},

"ok" : 1

}

> db.products.find({productName:'Pencil'}).explain()

{

"explainVersion" : "1",

"queryPlanner" : {

"namespace" : "pms.products",

"indexFilterSet" : false,

"parsedQuery" : {

"productName" : {

"$eq" : "Pencil"

}

},

"queryHash" : "508EE34D",

"planCacheKey" : "65C9C35B",

"maxIndexedOrSolutionsReached" : false,

"maxIndexedAndSolutionsReached" : false,

"maxScansToExplodeReached" : false,

"winningPlan" : {

"stage" : "COLLSCAN",

"filter" : {

"productName" : {

"$eq" : "Pencil"

}

},

"direction" : "forward"

},

"rejectedPlans" : [ ]

},

"command" : {

"find" : "products",

"filter" : {

"productName" : "Pencil"

},

"$db" : "pms"

},

"serverInfo" : {

"host" : "DESKTOP-40F6E3R",

"port" : 27017,

"version" : "5.0.8",

"gitVersion" : "c87e1c23421bf79614baf500fda6622bd90f674e"

},

"serverParameters" : {

"internalQueryFacetBufferSizeBytes" : 104857600,

"internalQueryFacetMaxOutputDocSizeBytes" : 104857600,

"internalLookupStageIntermediateDocumentMaxSizeBytes" : 104857600,

"internalDocumentSourceGroupMaxMemoryBytes" : 104857600,

"internalQueryMaxBlockingSortMemoryUsageBytes" : 104857600,

"internalQueryProhibitBlockingMergeOnMongoS" : 0,

"internalQueryMaxAddToSetBytes" : 104857600,

"internalDocumentSourceSetWindowFieldsMaxMemoryBytes" : 104857600

},

"ok" : 1

}

Create an Index

> db.products.createIndex({productName:1})

{

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.products.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_"

},

{

"v" : 2,

"key" : {

"productName" : 1

},

"name" : "productName\_1"

}

]

> db.products.createIndex({productPrice:1})

{

"numIndexesBefore" : 2,

"numIndexesAfter" : 3,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.products.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_"

},

{

"v" : 2,

"key" : {

"productName" : 1

},

"name" : "productName\_1"

},

{

"v" : 2,

"key" : {

"productPrice" : 1

},

"name" : "productPrice\_1"

}

]

>

> db.products.find({productName:'Pencil'}).explain('executionStats')

{

"explainVersion" : "1",

"queryPlanner" : {

"namespace" : "pms.products",

"indexFilterSet" : false,

"parsedQuery" : {

"productName" : {

"$eq" : "Pencil"

}

},

"maxIndexedOrSolutionsReached" : false,

"maxIndexedAndSolutionsReached" : false,

"maxScansToExplodeReached" : false,

"winningPlan" : {

"stage" : "FETCH",

"inputStage" : {

"stage" : "IXSCAN",

"keyPattern" : {

"productName" : 1

},

"indexName" : "productName\_1",

"isMultiKey" : false,

"multiKeyPaths" : {

"productName" : [ ]

},

"isUnique" : false,

"isSparse" : false,

"isPartial" : false,

"indexVersion" : 2,

"direction" : "forward",

"indexBounds" : {

"productName" : [

"[\"Pencil\", \"Pencil\"]"

]

}

}

},

"rejectedPlans" : [ ]

},

"executionStats" : {

"executionSuccess" : true,

"nReturned" : 3,

"executionTimeMillis" : 213,

"totalKeysExamined" : 3,

"totalDocsExamined" : 3,

"executionStages" : {

"stage" : "FETCH",

"nReturned" : 3,

"executionTimeMillisEstimate" : 29,

"works" : 4,

"advanced" : 3,

"needTime" : 0,

"needYield" : 0,

"saveState" : 2,

"restoreState" : 2,

"isEOF" : 1,

"docsExamined" : 3,

"alreadyHasObj" : 0,

"inputStage" : {

"stage" : "IXSCAN",

"nReturned" : 3,

"executionTimeMillisEstimate" : 29,

"works" : 4,

"advanced" : 3,

"needTime" : 0,

"needYield" : 0,

"saveState" : 2,

"restoreState" : 2,

"isEOF" : 1,

"keyPattern" : {

"productName" : 1

},

"indexName" : "productName\_1",

"isMultiKey" : false,

"multiKeyPaths" : {

"productName" : [ ]

},

"isUnique" : false,

"isSparse" : false,

"isPartial" : false,

"indexVersion" : 2,

"direction" : "forward",

"indexBounds" : {

"productName" : [

"[\"Pencil\", \"Pencil\"]"

]

},

"keysExamined" : 3,

"seeks" : 1,

"dupsTested" : 0,

"dupsDropped" : 0

}

}

},

"command" : {

"find" : "products",

"filter" : {

"productName" : "Pencil"

},

"$db" : "pms"

},

"serverInfo" : {

"host" : "DESKTOP-40F6E3R",

"port" : 27017,

"version" : "5.0.8",

"gitVersion" : "c87e1c23421bf79614baf500fda6622bd90f674e"

},

"serverParameters" : {

"internalQueryFacetBufferSizeBytes" : 104857600,

"internalQueryFacetMaxOutputDocSizeBytes" : 104857600,

"internalLookupStageIntermediateDocumentMaxSizeBytes" : 104857600,

"internalDocumentSourceGroupMaxMemoryBytes" : 104857600,

"internalQueryMaxBlockingSortMemoryUsageBytes" : 104857600,

"internalQueryProhibitBlockingMergeOnMongoS" : 0,

"internalQueryMaxAddToSetBytes" : 104857600,

"internalDocumentSourceSetWindowFieldsMaxMemoryBytes" : 104857600

},

"ok" : 1

}

>

> db.products.dropIndexes()

{

"nIndexesWas" : 3,

"msg" : "non-\_id indexes dropped for collection",

"ok" : 1

}

> db.products.getIndexes()

[ { "v" : 2, "key" : { "\_id" : 1 }, "name" : "\_id\_" } ]

> db.products.createIndex({productPrice:1})

{

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.products.createIndex({productName:1})

{

"numIndexesBefore" : 2,

"numIndexesAfter" : 3,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.products.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_"

},

{

"v" : 2,

"key" : {

"productPrice" : 1

},

"name" : "productPrice\_1"

},

{

"v" : 2,

"key" : {

"productName" : 1

},

"name" : "productName\_1"

}

]

> db.products.dropIndex('productName\_1');

{ "nIndexesWas" : 3, "ok" : 1 }

> db.products.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_"

},

{

"v" : 2,

"key" : {

"productPrice" : 1

},

"name" : "productPrice\_1"

}

]

> db.products.dropIndexes()

{

"nIndexesWas" : 2,

"msg" : "non-\_id indexes dropped for collection",

"ok" : 1

}

> db.products.getIndexes()

[ { "v" : 2, "key" : { "\_id" : 1 }, "name" : "\_id\_" } ]

> db.products.createIndex({productName:1,productPrice:1})

{

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.products.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_"

},

{

"v" : 2,

"key" : {

"productName" : 1,

"productPrice" : 1

},

"name" : "productName\_1\_productPrice\_1"

}

]

> db.books.createIndex({bookName:1},{unique:true});

{

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"createdCollectionAutomatically" : false,

"ok" : 1

}

> db.books.insertOne({\_id:1241,bookName:'Meluha',bookPrice:345,bookAuthor:'Chetan Bhagat'});

WriteError({

"index" : 0,

"code" : 11000,

"errmsg" : "E11000 duplicate key error collection: pms.books index: bookName\_1 dup key: { bookName: \"Meluha\" }",

"op" : {

"\_id" : 1241,

"bookName" : "Meluha",

"bookPrice" : 345,

"bookAuthor" : "Chetan Bhagat"

}

}) :

WriteError({

"index" : 0,

"code" : 11000,

"errmsg" : "E11000 duplicate key error collection: pms.books index: bookName\_1 dup key: { bookName: \"Meluha\" }",

"op" : {

"\_id" : 1241,

"bookName" : "Meluha",

"bookPrice" : 345,

"bookAuthor" : "Chetan Bhagat"

}

})

WriteError@src/mongo/shell/bulk\_api.js:465:48

mergeBatchResults@src/mongo/shell/bulk\_api.js:871:49

executeBatch@src/mongo/shell/bulk\_api.js:940:13

Bulk/this.execute@src/mongo/shell/bulk\_api.js:1182:21

DBCollection.prototype.insertOne@src/mongo/shell/crud\_api.js:264:9

@(shell):1:1

Above error will be thrown while inserting same book because of unique index