**.To create a collections for customer\_details and set multiple address fields using array, minimum 3 address.**

products> show collections

customer\_details

product\_details

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'coimbatore', 'coonoor' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'trichy', 'coimbatore', 'madurai' ]

}

]

**3. To remove first address from Arvind document using pop method.**

products> db.customer\_details.update({name:"aravind"},{$pop:{address:-1}})

DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'coimbatore', 'coonoor' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai' ]

}

]

**4. To remove last address from karthik document using pop method.**

products> db.customer\_details.updateOne({name:"karthik"},{$pop:{address:1}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'coimbatore' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai' ]

}

]

**To insert two new address to the person "Arvind" and "Karthik" in the address field using push method.**

products> db.customer\_details.updateOne({name:"karthik"},{$push:{address:"madurai"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.updateOne({name:"karthik"},{$push:{address:"kanyakumari"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'coimbatore', 'madurai', 'kanyakumari' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai' ]

}

]

products> db.customer\_details.updateOne({name:"aravind"},{$push:{address:"kanyakumari"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.updateOne({name:"aravind"},{$push:{address:"chennai"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'coimbatore', 'madurai', 'kanyakumari' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai', 'kanyakumari', 'chennai' ]

}

]

**5. To remove middle address from karthik document using pull method.**

products> db.customer\_details.updateOne({name:

... "karthik"},{$pull:{address:"coimbatore"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'madurai', 'kanyakumari' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai', 'kanyakumari', 'chennai' ]

}

]

**6. To insert new field as discount:"Aadi sales" to all the customer.**

products> db.customer\_details.updateMany({}, { $push: {discount:"adisales" } })

{

acknowledged: true,

insertedId: null,

matchedCount: 3,

modifiedCount: 3,

upsertedCount: 0

}

products> db.customer\_details.find().pretty()

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

age: 20,

address: [ 'ooty', 'gudalur', 'coonoor' ],

discount: [ 'adisales', 'adisales' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

age: 21,

address: [ 'metupalayam', 'madurai', 'kanyakumari' ],

discount: [ 'adisales', 'adisales' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

age: 26,

address: [ 'coimbatore', 'madurai', 'kanyakumari', 'chennai' ],

discount: [ 'adisales' ]

}

]

**7. To delete a specify document from the collection.**

products> db.customer\_details.deleteMany({age:26})

[

{

\_id: ObjectId("6443e5008664732b0a3da62c"),

name: 'surya',

address: [ 'ooty', 'gudalur', 'coonoor' ],

discount: [ 'adisales', 'adisales' ]

},

{

\_id: ObjectId("6443e5328664732b0a3da62d"),

name: 'karthik',

address: [ 'metupalayam', 'madurai', 'kanyakumari' ],

discount: [ 'adisales', 'adisales' ]

},

{

\_id: ObjectId("6443e5988664732b0a3da62e"),

name: 'aravind',

address: [ 'coimbatore', 'madurai', 'kanyakumari', 'chennai' ],

discount: [ 'adisales' ]

}

]

**8. To delete a repeated document from the collection.**

products> db.customer\_details.deleteMany({},{age:26})

{ acknowledged: true, deletedCount: 3 }

products> db.customer\_details.find().pretty()

**9. To drop the collection**

products> db.customer\_details.drop()

true

products> show collections

product\_details

**10. To drop the database.**

products> db.dropDatabase()

{ ok: 1, dropped: 'products' }

products> show dbs

admin 40.00 KiB

config 108.00 KiB

first 136.00 KiB

local 72.00 KiB

showroom 40.00 KiB

student 72.00 KiB