Vaishnavi Vishwas Pawar

Assignment No. 8

Part 1. Complete the JSON file below.

{
"ICID": "ACF",
"ICName": "Acme Finance",
"ICLocation": 1-a) REPLACE_THIS_VALUE,
"MutualFund": 1-b) REPLACE_THIS_VALUE
}

1-a) Choose an appropriate data type for the ICLocation field and adding values to the field. (0.5 pt)

1-b). Complete the JSON file below by choosing an appropriate data type for the MutualFund field and adding values to the field. (0.5 pt)

```
{
   "ICID": "ACF",
   "ICName": "Acme Finance",
   "ICLocation": [
      "Chicago",
      "Denver",
      "Houston"
   ],
   "MutualFund": [
      {
         "MFID": "BG",
         "MFName": "Big Growth"
      },
      {
         "MFID": "SG",
         "MFName": "Steady Growth"
      }
   ]
}
```

Vaishnavi Vishwas Pawar

Part II: Write MongoDB queries to retrieve data from the *product* collection, which contains the six documents below. (Import product_hw6.json Download product_hw6.json to populate your MongoDB collection)

2. (Scalar, number, and $lt) Find all documents whose price value is less than *70*. (0.25 pt)

Answer:

{ price: { $lt: 70 } }

Result:

3. (Scalar and string) Find all documents whose venderName value is *"Mountain King"*. In other words, find all products that are supplied by "Mountain King". (0.25 pt)

Answer:

{"vendor.vendorName": "Mountain King"}

Vaishnavi Vishwas Pawar

Result:

ter

```
{"vendor.vendorName": "Mountain King"}
```

ERY RESULTS: 1-4 OF 4

```
_id: ObjectId('661f39aea1c354999fbccd15')
id : "2X2"
name : "Easy Boot"
price : 70
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd16')
id : "3X3"
name : "Cosy Sock"
price : 15
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (3)
```

```
_id: ObjectId('661f39aea1c354999fbccd18')
id : "5X5"
name : "Tina Tent"
price : 150
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd19')
id : "6X6"
name : "Biggy Tent"
price : 250
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (1)
```

4. (Array) Find all documents where categories value is an array that contains the string *"footwear"* as one of its elements. In other words, find all products that belong to "footwear". (0.25 pt)

Answer:

{categories: "footwear"}

Result:

Filter `{categories: "footwear"}`

QUERY RESULTS: 1-3 OF 3

```
_id: ObjectId('661f39aea1c354999fbccd15')
id : "2X2"
name : "Easy Boot"
price : 70
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd16')
id : "3X3"
name : "Cosy Sock"
price : 15
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (3)
```

```
_id: ObjectId('661f39aea1c354999fbccd17')
id : "4X4"
name : "Dura Boot"
price : 90
▸ categories : Array (2)
```

5. (Array of documents) Find all documents where the reviews array has at least one embedded document that contains the field reviewer that is equal to *"KC"*. In other words, find all products that are reviewed by KC. (0.25 pt)

Answer:

{ "reviews.reviewer": "KC" }

Vaishnavi Vishwas Pawar

Result:

Filter { "reviews.reviewer": "KC" }

QUERY RESULTS: 1-2 OF 2

```
_id: ObjectId('661f39aea1c354999fbccd16')
id : "3X3"
name : "Cosy Sock"
price : 15
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (3)
```

```
_id: ObjectId('661f39aea1c354999fbccd17')
id : "4X4"
name : "Dura Boot"
price : 90
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

6. (Exact match to a document in an array) Find all documents where an element in the reviews array matches the specified document, *{reviewer: "Sage", ratings: 4}*. (0.25 pt)

Answer:

{ reviews: { $elemMatch: { reviewer: "Sage", ratings: 4 } } }

Result:

```
{reviews: { $elemMatch: { reviewer: "Sage", ratings: 4 } } }
```

RESULTS: 1-1 OF 1

```
_id: ObjectId('661f39aea1c354999fbccd18')
id : "5X5"
name : "Tina Tent"
price : 150
▸ categories : Array (2)
▸ vendor : Object
▾ reviews : Array (2)
  ▾ 0: Object
      reviewer : "MJ"
      ratings : 5
  ▾ 1: Object
      reviewer : "Sage"
      ratings : 4
```

7. (Array and $or) Find all documents where the categories is an array that contains either *"boot"* or *"tent"*. (0.25 pt)

Answer:

{$or: [{ categories: "boot" },{ categories: "tent" }]}

Result:

```
{$or: [{ categories: "boot" },{ categories: "tent" }]}
```

Y RESULTS: 1-4 OF 4

```
_id: ObjectId('661f39aea1c354999fbccd15')
id : "2X2"
name : "Easy Boot"
price : 70
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd17')
id : "4X4"
name : "Dura Boot"
price : 90
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd18')
id : "5X5"
name : "Tina Tent"
price : 150
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (2)
```

```
_id: ObjectId('661f39aea1c354999fbccd19')
id : "6X6"
name : "Biggy Tent"
price : 250
▸ categories : Array (2)
▸ vendor : Object
▸ reviews : Array (1)
```