

MongoDB Assignment 3:

By Esaq A

Update Documents:

- Update one single document:

```
assignment> db.inventory.updateOne(
|   { item: "paper" },
|   {
|     $set: { "size.uom": "cm", status: "P" },
|     $currentDate: { lastModified: true }
|   }
| )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

- Update multiple documents

```
assignment> db.inventory.updateMany(
|   { "qty": { $lt: 50 } },
|   {
|     $set: { "size.uom": "in", status: "P" },
|     $currentDate: { lastModified: true }
|   }
| )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
```

- Replace a document

```
assignment> db.inventory.replaceOne(
|   { item: "paper" },
|   { item: "paper", instock: [ { warehouse: "A", qty: 60 }, { warehouse: "B", qty: 40 } ] }
| )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Delete Documents:

- Delete one document which matches the condition

```
[assignment> db.inventory.deleteOne( { status: "D" } )  
{ acknowledged: true, deletedCount: 1 }
```

- Delete all documents which matches the condition

```
[assignment> db.inventory.deleteMany({ status : "A" })  
{ acknowledged: true, deletedCount: 4 }
```

- Delete all the documents

```
[assignment> db.inventory.deleteMany({})  
{ acknowledged: true, deletedCount: 5 }
```

Atomicity and Transactions:

In MongoDB, a write operation is atomic on the level of a single document, even if the operation modifies multiple values. The conditions are checked, even when the command happens in parallel.

```
[assignment> db.games.insertOne({_id: 1, score: 80})  
{ acknowledged: true, insertedId: 1 }
```

```

assignment> // Update A
| db.games.updateOne(
|   { score: 80 },
|   {
|     $set: { score: 90 }
|   }
| )
|
| // Update B
| db.games.updateOne(
|   { score: 80 },
|   {
|     $set: { score: 100 }
|   }
| )
[ ]
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}

```

After the Update A, the value of score became 90, which doesn't satisfy the condition of Update B. Therefore, Update B won't work.

```

assignment> db.games.find({})
[ { _id: 1, score: 90 } ]

```

As you can see here, the Update B didn't work.

But if we change the conditions a little bit, we can see

```
assignment> // Update A
| db.games.updateOne(
|   { _id: 1 },
|   {
|     $inc: { score: 10 }
|   }
| )
|
| // Update B
| db.games.updateOne(
|   { _id: 1 },
|   {
|     $inc: { score: 20 }
|   }
| )
| {
|   acknowledged: true,
|   insertedId: null,
|   matchedCount: 1,
|   modifiedCount: 1,
|   upsertedCount: 0
| }
```

Because `_id` field of the document remains the same after update A, so the update B also will happen after update A.

```
assignment> db.games.find({})
[ { _id: 1, score: 120 } ]
```

Here, as you can see the value of the score field has increased from 90 to 100 in update A, and 100 to 120 in update B. So both the Updates worked.

Projection:

- Projecting specific fields from the document:

```
assignment> db.inventory.find({}, {item:1}).pretty()
[
  { _id: ObjectId('6881b748039a334bcf03c0c9'), item: 'journal' },
  { _id: ObjectId('6881b748039a334bcf03c0ca'), item: 'notebook' },
  { _id: ObjectId('6881b748039a334bcf03c0cb'), item: 'paper' },
  { _id: ObjectId('6881b748039a334bcf03c0cc'), item: 'planner' },
  { _id: ObjectId('6881b748039a334bcf03c0cd'), item: 'postcard' }
]
assignment> db.inventory.find({}, {qty:1}).pretty()
[
  { _id: ObjectId('6881b748039a334bcf03c0c9'), qty: 25 },
  { _id: ObjectId('6881b748039a334bcf03c0ca'), qty: 50 },
  { _id: ObjectId('6881b748039a334bcf03c0cb'), qty: 100 },
  { _id: ObjectId('6881b748039a334bcf03c0cc'), qty: 75 },
  { _id: ObjectId('6881b748039a334bcf03c0cd'), qty: 45 }
]
```

- Projecting multiple fields from document:

```
assignment> db.inventory.find({}, {item:2,qty:1}).pretty()
[
  {
    _id: ObjectId('6881b748039a334bcf03c0c9'),
    item: 'journal',
    qty: 25
  },
  {
    _id: ObjectId('6881b748039a334bcf03c0ca'),
    item: 'notebook',
    qty: 50
  },
  {
    _id: ObjectId('6881b748039a334bcf03c0cb'),
    item: 'paper',
    qty: 100
  },
  {
    _id: ObjectId('6881b748039a334bcf03c0cc'),
    item: 'planner',
    qty: 75
  },
  {
    _id: ObjectId('6881b748039a334bcf03c0cd'),
    item: 'postcard',
    qty: 45
  }
]
```

- Exclusion of `_id` field from projecting document:

```
assignment> db.inventory.find({}, {item:2,qty:1,_id:0}).pretty()
[
  { item: 'journal', qty: 25 },
  { item: 'notebook', qty: 50 },
  { item: 'paper', qty: 100 },
  { item: 'planner', qty: 75 },
  { item: 'postcard', qty: 45 }
]
```