**Section 7: Updating Documents (8min)**

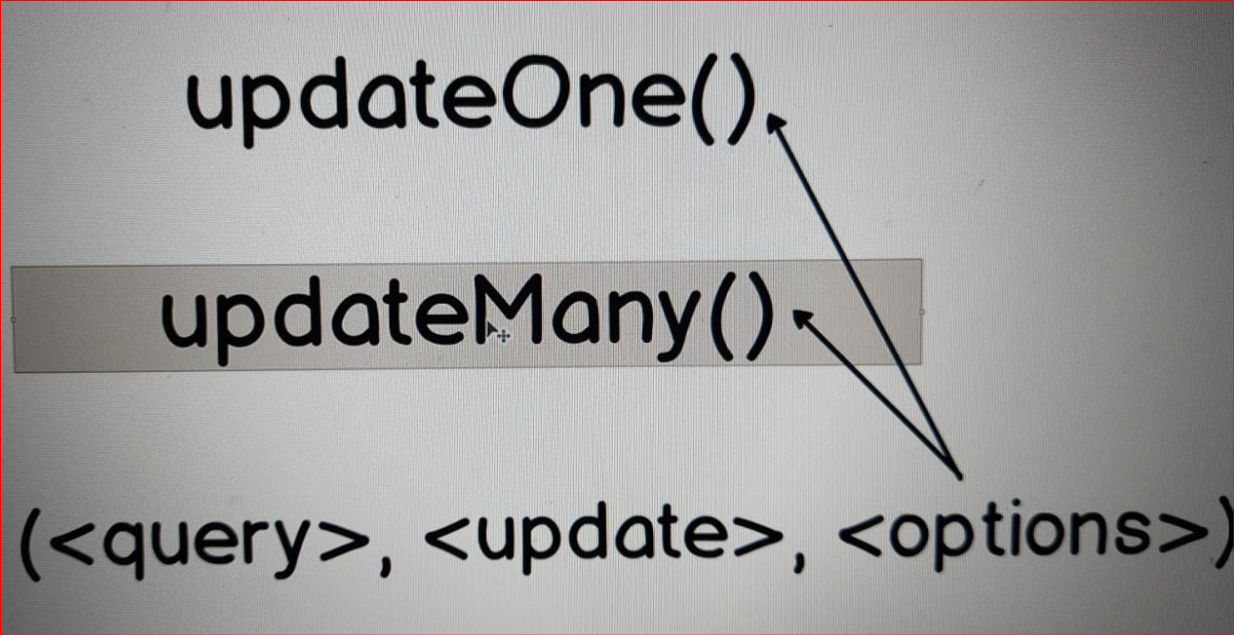
**- Update Methods updateOne() and updateMany()**

**- Update Operators**

**- PRACTICE - Updating Documents using Update Operators**

**There are two methods:**

**"updateOne" and “updateMany” that can be used for document updates.**

****

**And of course from the title you can understand that :**

**"updateOne" updates just one document**

**"updateMany" can update simultaneously several documents.**

**In the parenthesis you can pass several arguments to each of those methods and they are query, update and options.**

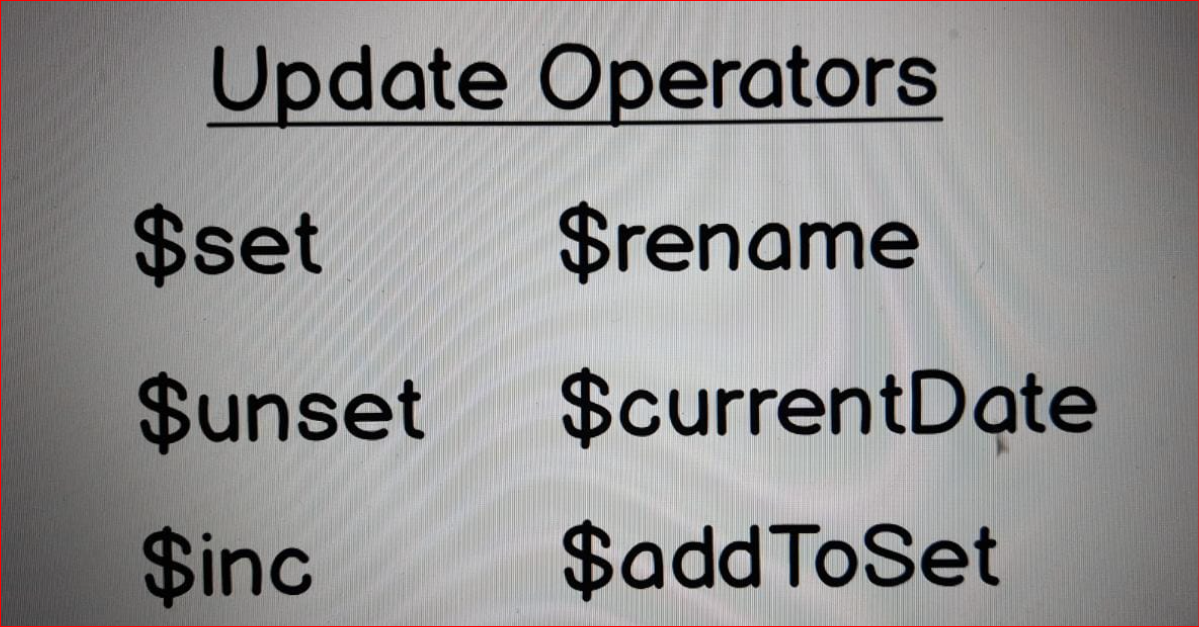
**Each of those arguments is an object.**

**In the query you specify which documents you want to update.**

**In the update you specify what you want to update, which operations you want to perform.**

**And of course those two methods are collection methods.**

**You cannot use those methods without update operators!**

**let's talk about update operators and after that of course we will try update methods in action.**

* **Update operators.**

**They are used in this update argument that is passed to "updateOne" and "updateMany" methods. That is the second argument.**

**There are several popular operators.**

**Each update operator, same as query operator, has dollar sign $**

**Most popular operator are:**

**$set:**

* **It sets value of certain field.**

**$unset:**

* **It unsets certain field.**

**$inc:**

* **It increments certain field by specified amount.**

**$rename:**

* **It renames field and so on.**

**31. PRACTICE - Updating Documents using Update Operators**

**// updateOne({query},{update},{options})**

**// updateOne with $set operation**

**// Let's take 2nd document, for example "postId" 2618 and then let's set "shared" to "true".**

**// Now it is "false".**

db.getCollection("posts").findOne({postId: 2618})

db.posts.updateOne(

{postId: 2618},

{$set: {shared: true}}

)

**// Output:**

**//{**

**// "acknowledged" : true,**

**// "insertedId" : null,**

**// "matchedCount" : 1.0, It means one document found**

**// "modifiedCount" : 1.0, It means that one document is updates**

**// "upsertedCount" : 0.0**

**//}**

**// Verify the Output:**

db.getCollection("posts").findOne({postId: 2618})

**// updateOne**

**// Please find document with "postId" 1151 and change it's title.**

**// Title should be: "What is the average salary of the senior frontend developer?"**

db.getCollection("posts").findOne({postId: 1151})

db.posts.updateOne(

{postId: 1151},

{$set: {title: "What is the average salary of the 'senior' frontend developer"}}

);

**// Sucessfully Update:**

**//{**

**// "acknowledged" : true,**

**// "insertedId" : null,**

**// "matchedCount" : 1.0,**

**// "modifiedCount" : 1.0,**

**// "upsertedCount" : 0.0**

**//}**

db.getCollection("posts").findOne({postId: 1151})

**// $unset operation. with updateMany()**

**// $unset deletes certain field.**

**// let's for example delete "tags" field here in this document where "tags" array is empty.**

**// And let's for example delete "tags" field here in this document where "tags" array is empty.**

**// Let's first find all documents with empty "tags" array. "tags" equal to empty array [].**

**db.getCollection("posts").findOne({tags: []});**

**// let's instead of "updateOne" use "updateMany". And we will simply $unset "tags" field.**

**// Instead of string simply pass 1.**

db.posts.updateMany(

{tags: []},

{$unset: {tags: 1}}

)

**// Output: there are actually 2 documents with empty tags**

**//{**

**// "acknowledged" : true,**

**// "insertedId" : null,**

**// "matchedCount" : 2.0,**

**// "modifiedCount" : 2.0,**

**// "upsertedCount" : 0.0**

**//}**

db.getCollection("posts").findOne({tags: []});

**// Output: NULL (we got null means no document with empty array)**

db.getCollection("posts").findOne({postId: 2618}); **// No tags array in this document now!**

**// $inc (increment operator) with updateOne**

**// Please use $inc operator and increment number of the comments in document with postId = 8451**

**// Increment this number by one.**

db.getCollection("posts").findOne({postId: 8451}); **// Currentlyy it has Comments: 2.0**

db.posts.updateOne(

{postId: 8451},

{$inc : {comments: 1}}

)

db.getCollection("posts").findOne({postId: 8451});

**// Output :**

**// It got updated "comments" : 3.0**

**// That is a short overview of the update operators**

**// and in the next section we will discuss delete methods and will delete some of the documents in our collection.**