

Queries

Mark Scott
[linkedin.com/in/tripletdad99](https://www.linkedin.com/in/tripletdad99)
@tripletdad99



pluralsight 
hardcore dev and IT training

Introduction

Retrieve Documents

Model Methods:

`find`

`findById`

`findOne`

`where`

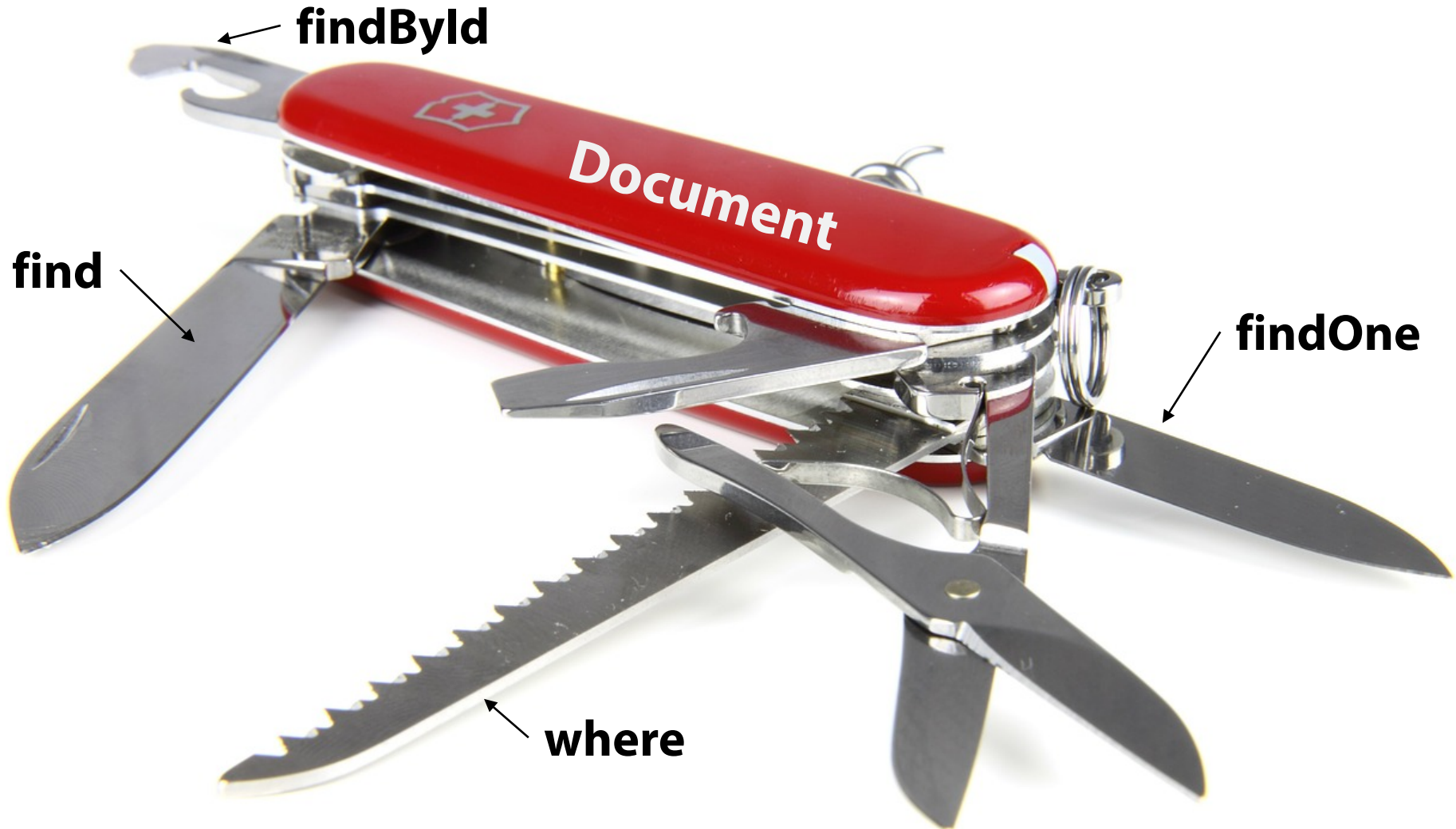
Updating & Removing Documents

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Documents and Their Helper Methods



Model.find(conditions, [fields], [options], [callback])

```
var Standup = require('../models/standup.server.model.js');

// No callback... Deferred execution
var query = Standup.find();

// With callback... Executes query immediately
Standup.find(function (err, results) {
  // handle the error... Or results here
});

// With callback and query conditions
Standup.find({ memberName: 'David' }, function (err, results) {
  // handle the error... Or results here
});

// Limit the returned fields...
Standup.find({ memberName: 'Mary' }, 'memberName impediment',
  function (err, results) {
    // handle the error... Or results here
  });
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.findOne(conditions, [fields], [options], [callback])

```
var Standup = require('../models/standup.server.model.js');

// No callback... No conditions...
var query = Standup.findOne();
query.exec(function (err, results) {
    // handle the error... Or results here
});

// With conditions...
var query = Standup.findOne({ memberName: 'Mark' });

// With condition, no field filtering, and options...
var query = Standup.findOne({ memberName: 'Bob' }, null, { limit: 10 });

// Options examples... skip, limit, sort
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.findById(id, [fields], [options], [callback])

```
var id = '541c6fefefdf9c84172162a6';
var Standup = require('../models/standup.server.model.js');

// By Id... No conditions...
var query = Standup.findById(id);
query.exec(function (err, doc) {
  // handle the error... Or results here
});

// Same as above... Chained method calls...
Standup.findById(id).exec(function (err, results) { ... });

// By Id... Return every field BUT impediment...
var query = Standup.findById(id, '-impediment');
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Comparison Query Operators



\$gt	greater than
\$gte	greater than or equal to
\$in	exists in
\$lt	less than
\$lte	less than or equal to
\$ne	not equal to
\$nin	does not exist

These are the same as found in MongoDB

```
// Example: find customers with discount >= 10%...
Customer.find({discount: {$gte: 10}, function(err, results) {
  if (err) throw err;
  console.log(results);
}});
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.where(path, [val])

```
Customer.find({discount: {$gte: 10, $lt: 20}, function(err, results) {  
  if (err) throw err;  
  console.log(results);  
});
```

```
Customer.where('discount').gte(10).lt(20).exec(function(err, results) {  
  if (err) throw err;  
  console.log(results);  
});
```

```
// Chain where methods together...  
Customer.where('discount').gte(10).lt(20)  
  .where('zipCode', '12345')  
  .exec(function(err, results) {  
    if (err) throw err;  
    console.log(results);  
  });
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Updating and Removing Documents

Update

Remove

findByIdAndUpdate

findByIdAndRemove

```
Standup.findById(id).exec(function (err, doc) {  
  // handle any errors  
  if (err) return errorHandler(err);  
  
  // update the found document  
  doc.impediment('None');  
  doc.save(function (err) {  
    if (err) return errorHandler(err);  
    console.log('Document updated');  
  });  
});
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.update(conditions, update, [options], [callback])

```
var Standup = require('../models/standup.server.model.js');

var condition = { memberName: 'Mary' };
var update = { impediment: 'None - Mary no longer works here!' };

Standup.update(condition, update, function(err, numberAffected, rawResponse) {
    // Handle error or raw results here...
});

// finding a document - then updating it...
Standup.findOne({ memberName: 'Mary' }, function(err, doc) {
    // Handle errors here... Validate document results... Etc.
    doc.impediment = 'None - Mary won the lottery and is on an island now';
    doc.save(function (err) {
        // Handle errors
    });
});
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.update(conditions, update, [options], [callback])

Option	Description	Default Value
safe	Safe mode – default to value set in schema	True
upsert	Create the document if it does not match	False
multi	Determines if multiple documents should be updated or not	False
strict	Override the strict option for this update	
overwrite	Disables the update-only mode to allow for overwrite of document	False

```
// Example: update multiple documents that match condition
var condition = { firstName: 'Bob' };
var update = { firstName: 'Robert' };

Customer.update(condition, update, { multi: true },
function(err, numberAffected, raw) {
    // Handle error, returned # affected, and raw response from MongoDB...
});
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.remove(conditions, [callback])

```
var Standup = require('../models/standup.server.model.js');

var condition = { memberName: 'Mary' };

Standup.remove(condition, function(err) {
    // Handle error here...
});

// Remove any document created on or after Halloween day
var gteDate = new Date(2014, 10, 31);
Standup.remove({ createdOn: { $gte: gteDate }}, function (err) {
    // Handle error here...
});

// Execute query w/o a callback function - does not wait on response
var query = Standup.remove({ createdOn: { $gte: gteDate }});
query.exec();
```

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Model.findByIdAndUpdate(id, update, [options], [callback])

Option	Description	Default Value
new	Set to true to return the modified document rather than the original.	True
upsert	Create the document if it does not match	False
select	Specify the document fields to return	

Model.findByIdAndRemove(id, [options], [callback])

Option	Description	Default Value
select	Specify the document fields to return	

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Demo

Querying Documents



**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.

Module Summary

**Documents
Helper Methods**

**Updating and
Removing
Documents**

**Demo
List Notes
Filter By Name**

**Do Not Place Anything
in This Space**

(Add watermark during
editing)

Note: Warning will not appear
during Slide Show view.