

MongoDB Basics

1. **Create a collection called 'games'. We're going to put some games in it.**

Ans=>
use games

2. **Add 5 games to the database. Give each document the following properties: name, genre, rating (out of 100)**

Ans=>
db.games.save({ name: "Spy Hunter", genre: "Racing", rating: 76});
db.games.save({ name: "Mario Kart 64", genre: "Racing", rating: 96});
db.games.save({ name: "Tetris", genre: "Puzzle", rating: 83});
db.games.save({ name: "Mega Man 5", genre: "Platformer", rating: 81});
db.games.save({ name: "Star Fox", genre: "Action", rating: 71});

3. **Query to return all games**

Ans=>
db.games.find()

4. **Query to format pretty**

Ans=>
Db.games.pretty()

5. **Query to find one**

Ans=>
db.games.findOne({name:"Mario Kart 64"})

6. **Query to find top 3 highest rated games**

Ans=>
db.games.find().limit(3).sort({'rating': -1})

7. **Update your two favorite games to have two achievements called 'Game Master' and 'Speed Demon' by two methods.**

Javascript to update

```
var g = db.games.findOne({'name':'Mega Man 5'})
g.achievements = []
g.achievements.push({'name':'Defeat Two Wheeler','points':200})
g.achievements.push({'name':'Beat the game','points':1200})
db.games.save(g)
```

Other way: because it's javascript, there are any number of steps they can take. Just want them to use save()

```
var g = db.games.findOne({'name':'Mega Man 5'}) g.achievements = []
g.achievements.push({'name':'Defeat Two Wheeler','points':200})
g.achievements.push({'name':'Beat the game','points':1200}) db.games.save(g)
```

8. query that returns all the games that have both the 'Game Master' and the 'Speed Demon' achievements.

Ans=>

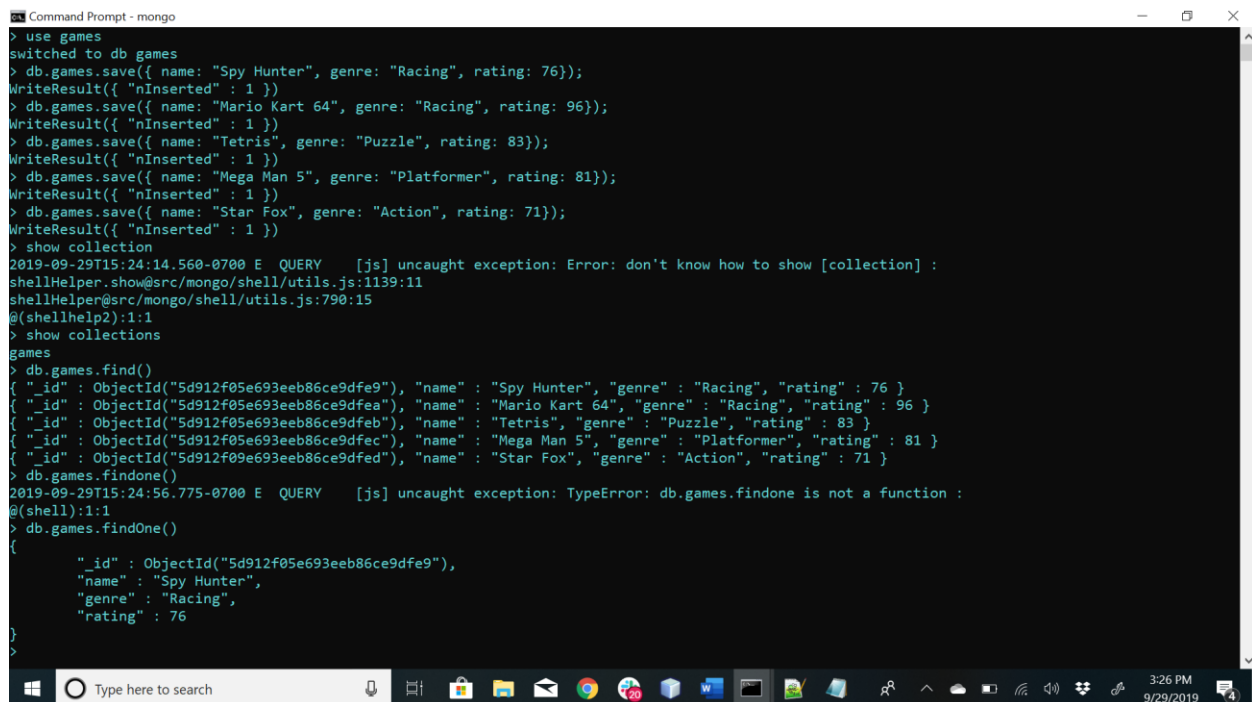
```
db.games.find({achievement: {$all: ['Game Master', 'Speed Demon']}});
```

9. query that returns only games that have achievements

Ans=>

```
db.games.find({ achievements: {$elemMatch: {}}
```

Screenshots for reference:



```
Command Prompt - mongo
> use games
switched to db games
> db.games.save({ name: "Spy Hunter", genre: "Racing", rating: 76});
WriteResult({ "nInserted" : 1 })
> db.games.save({ name: "Mario Kart 64", genre: "Racing", rating: 96});
WriteResult({ "nInserted" : 1 })
> db.games.save({ name: "Tetris", genre: "Puzzle", rating: 83});
WriteResult({ "nInserted" : 1 })
> db.games.save({ name: "Mega Man 5", genre: "Platformer", rating: 81});
WriteResult({ "nInserted" : 1 })
> db.games.save({ name: "Star Fox", genre: "Action", rating: 71});
WriteResult({ "nInserted" : 1 })
> show collection
2019-09-29T15:24:14.560-0700 E QUERY [js] uncaught exception: Error: don't know how to show [collection] :
shellHelper.show@src/mongo/shell/utils.js:1139:11
shellHelper@src/mongo/shell/utils.js:790:15
@(shellhelp2):1:1
> show collections
games
> db.games.find()
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfe9"), "name" : "Spy Hunter", "genre" : "Racing", "rating" : 76 }
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfea"), "name" : "Mario Kart 64", "genre" : "Racing", "rating" : 96 }
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfec"), "name" : "Tetris", "genre" : "Puzzle", "rating" : 83 }
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfef"), "name" : "Mega Man 5", "genre" : "Platformer", "rating" : 81 }
{ "_id" : ObjectId("5d912f09e693eeb86ce9dfed"), "name" : "Star Fox", "genre" : "Action", "rating" : 71 }
> db.games.findone()
2019-09-29T15:24:56.775-0700 E QUERY [js] uncaught exception: TypeError: db.games.findone is not a function :
@(shell):1:1
> db.games.findOne()
{
  "_id" : ObjectId("5d912f05e693eeb86ce9dfe9"),
  "name" : "Spy Hunter",
  "genre" : "Racing",
  "rating" : 76
}
```

```

Command Prompt - mongo
> d)
2019-09-29T15:33:33.458-0700 E QUERY [js] uncaught exception: SyntaxError: unexpected token: ')' :
@(shell):1:1
> db.games.findOne({name:"Mario Kart 64"})
{
  "_id" : ObjectId("5d912f05e693eeb86ce9dfe9"),
  "name" : "Mario Kart 64",
  "genre" : "Racing",
  "rating" : 96
}
> db.games.find().limit(3).sort({'rating': -1})
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfe9"), "name" : "Mario Kart 64", "genre" : "Racing", "rating" : 96 }
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfeb"), "name" : "Tetris", "genre" : "Puzzle", "rating" : 83 }
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfec"), "name" : "Mega Man 5", "genre" : "Platformer", "rating" : 81, "achievements" : [ { "name" : "Defeat Two Wheeler", "points" : 200 }, { "name" : "Beat the game", "points" : 1200 } ] }
> db.games.update(
... {name:"Star Fox"},
... {
...   $set: {'achievements':[
...     {'name':'Collect 50 rings', 'points':100},
...     {'name':'Finish under 2 minutes','points':135}
...   ]
... }
... })
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 0 })
>

```

```

Command Prompt - mongo
> use games
switched to db games
> db.games.update(
... {name:"Star Fox"},
... {
...   $set: {'achievements':[
...     {'name':'Collect 50 rings', 'points':100},
...     {'name':'Finish under 2 minutes','points':135}
...   ]
... }
... })
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 0 })
> db.games.find({achievement: {$all: ['Game Master', 'Speed Demon']}});
> db.games.find(
... achievements: {$elemMatch: {}}
... })
{ "_id" : ObjectId("5d912f05e693eeb86ce9dfec"), "name" : "Mega Man 5", "genre" : "Platformer", "rating" : 81, "achievements" : [ { "name" : "Defeat Two Wheeler", "points" : 200 }, { "name" : "Beat the game", "points" : 1200 } ] }
{ "_id" : ObjectId("5d912f09e693eeb86ce9dfed"), "name" : "Star Fox", "genre" : "Action", "rating" : 71, "achievements" : [ { "name" : "Collect 50 rings", "points" : 100 }, { "name" : "Finish under 2 minutes", "points" : 135 } ] }
> db.games.find({achievement: { $exists: true}});
>

```