g·able

MongoDB 3.0

Server-side Javascript

Topics

- Server-side Script
- Labs

Server-side Script

- MongoDB provides the following commands, methods, and operator that perform server-side execution of JavaScript code.

- You can also specify a JavaScript file to the mongo shell to run on the server.

- MongoDB uses “Google 8 JavaScript engine”, which became the default in 2.4, to run Javascript on the server. It allows multiple JavaScript operations to execute at the same time.

Scripting

- From the system prompt, use mongo to evaluate JavaScript.
- Use the --eval option to mongo to pass the shell a JavaScript fragment, as in the following:

  ```
  mongo test –-eval “printjson(db.getCollectionNames())"
  ```

  This returns the output of db.getCollectionNames() using the mongo shell connected to the mongod or mongos instance running on port 27017 on the localhost interface.

- You can specify a .js file to the mongo shell, and mongo will execute the JavaScript directly. Consider the following example:

  ```
  mongo localhost:27017/test myjsfile.js
  ```

- You can execute a .js file from within the mongo shell, using the load() function, as in the following:

  ```
  load("myjstest.js")
  ```

Common mongo shell Helper

Shell Helpers	JavaScript Equivalents
show dbs, show databases	db.adminCommand('listDatabases')
use <db>	db = db.getSiblingDB('<db>')
show collections	db.getCollectionNames()
show users	db.getUsers()
show roles	db.getRoles({showBuiltinRoles: true})
show log <logname>	db.adminCommand({ 'getLog' : '<logname>' })
show logs	db.adminCommand({ 'getLog' : '*' })

it	cursor = db.collection.find()
	while (cursor.hasNext()){
	  cursor.next();
	}

Lab1: Create init script

1. On Desktop, create folder named “scripts”.

2. Open scripts folder, by double-click on the folder icon, then create empty documents by right-click on the empty space then choose New Document > Empty Document. Enter “init.js” for file name.

3. Open init.js file by double-click on the file icon.

4. Enter the following script, then save file.

```
conn = new Mongo();
db = conn.getDB("address_book");

db.dropDatabase();

db.contact.insert({name: "Steve", lastname: "Jobs", email: "steve@apple.com"});
db.contact.insert({name: "Tim", lastname: "Cook", email: "tim@apple.com"});
db.contact.insert({name: "Bill", lastname: "Gates", email: "bill.g@microsoft.com"});
db.contact.insert({name: "Mark", lastname: "Zuckerberg", email: "mark@facebook.com"});
```

Lab1: Run Script

5. Open Terminal.
6. Goto scripts folder.

```
cd /home/mongo-root/Desktop/scripts
```

7. Run script

```
mongo localhost:27017/address_book init.js
```

8. Open mongo shell and check the result.

```
mongo
use address_book
db.contact.find()
```

Lab2: Create Read Script

1. Under folder “scripts”, create the second script named “read.js”
2. Enter the following script, then save it.

```
var conn = new Mongo();
var db = conn.getDB("address_book");

var count = 0;
cursor = db.contact.find();
while ( cursor.hasNext() ) {
  count = count+1;
  cursor.next();
}

db.totalContact.insert({totContact: count})
```

3. Open Terminal.
4. Run script.

```
mongo read.js
```

5. Open mongo shell and check the result.

```
mongo
use address_book
db.totalContact.find()
```

References

- http://docs.mongodb.org/manual/core/server-side-javascript/