

[Mongodb]Lab02_Mongodb_Command

[Mongodb] 02. Mongodb Command

학습 목표

가. 기본 명령어를 실습을 통해 이해할 수 있다.

목차

[Mongodb] 02. Mongodb Command

2-1. 기본 명령어 이해

2-2. 기본 명령어 이해(2)

[Review]

[실습 1] 아래 명령을 실행시켜보고 어떤 명령인지 간단한 설명을 작성해보자.

```
> show dbs
> use local
> show collections
> db.startup_log.find()
```

2-1. 기본 명령어 이해

(가) MongoDB 실행

다른 창에서 아래 명령 실행

```
..> mongo
```

```
> show dbs
admin 0.000GB
local 0.000GB
```

데이터 베이스 사용

```
> use local
switched to db local
```

collections 을 확인하기

```
> show collections
startup_log
```

> db.startup_log.find()

```
{ "_id" : "DESKTOP-C8MCT7F-1504109488348", "hostname" : "DESKTOP-C8MCT7F", "startTime" :
ISODate("2017-08-30T16:11:28Z"), "startTimeLocal" : "Thu Aug 31 01:11:28.348", "cmdLine" : { }, "pid" :
NumberLong(3876), "buildinfo" : { "version" : "3.4.7", "gitVersion" : "cf38c1b8a0a8dca4a11737581beafef4fe120bcd",
"targetMinOS" : "Windows 7/Windows Server 2008 R2", "modules" : [ ], "allocator" : "tcmalloc", "javascriptEngine" :
"mozjs", "sysInfo" : "deprecated", "versionArray" : [ 3, 4, 7, 0 ], "openssl" : { "running" : "OpenSSL 1.0.1u-fips 22
Sep 2016", "compiled" : "OpenSSL 1.0.1u-fips 22 Sep 2016" }, "buildEnvironment" : { "distmod" : "2008plus-ssl",
"distarch" : "x86_64", "cc" : "cl: Microsoft (R) C/C++ Optimizing Compiler Version 19.00.24223 for x64", "ccflags" :
"/nologo /EHsc /W3 /wd4355 /wd4800 /wd4267 /wd4244 /wd4290 /wd4068 /wd4351 /we4013 /we4099 /we4930 /Z7
/errorReport:none /MD /O2 /Oy- /bigobj /Gw /Gy /Zc:inline", "cxx" : "cl: Microsoft (R) C/C++ Optimizing Compiler
Version 19.00.24223 for x64", "cxxflags" : "/TP", "linkflags" : "/nologo /DEBUG /INCREMENTAL:NO
/LARGEADDRESSAWARE /OPT:REF", "target_arch" : "x86_64", "target_os" : "windows" }, "bits" : 64, "debug" :
```

```
false, "maxBsonObjectSize" : 16777216, "storageEngines" : [ "devnull", "ephemeralForTest", "mmapv1", "wiredTiger" ] } }
{ "_id" : "DESKTOP-C8MCT7F-1504109726717", "hostname" : "DESKTOP-C8MCT7F", "startTime" :
ISODate("2017-08-30T16:15:26Z"), "startTimeLocal" : "Thu Aug 31 01:15:26.717", "cmdLine" : { }, "pid" :
NumberLong(15508), "buildinfo" : { "version" : "3.4.7", "gitVersion" :
"cf38c1b8a0a8dca4a11737581beafef4fe120bcd", "targetMinOS" : "Windows 7/Windows Server 2008 R2", "modules"
: [ ], "allocator" : "tcmalloc", "javascriptEngine" : "mozjs", "sysInfo" : "deprecated", "versionArray" : [ 3, 4, 7, 0 ],
"openssl" : { "running" : "OpenSSL 1.0.1u-fips 22 Sep 2016", "compiled" : "OpenSSL 1.0.1u-fips 22 Sep 2016" },
"buildEnvironment" : { "distmod" : "2008plus-ssl", "distarch" : "x86_64", "cc" : "cl: Microsoft (R) C/C++ Optimizing
Compiler Version 19.00.24223 for x64", "ccflags" : "/nologo /EHsc /W3 /wd4355 /wd4800 /wd4267 /wd4244 /wd4290
/wd4068 /wd4351 /we4013 /we4099 /we4930 /Z7 /errorReport:none /MD /O2 /Oy- /bigobj /Gw /Gy /Zc:inline", "cxx"
: "cl: Microsoft (R) C/C++ Optimizing Compiler Version 19.00.24223 for x64", "cxxflags" : "/TP", "linkflags" :
"/nologo /DEBUG /INCREMENTAL:NO /LARGEADDRESSAWARE /OPT:REF", "target_arch" : "x86_64",
"target_os" : "windows" }, "bits" : 64, "debug" : false, "maxBsonObjectSize" : 16777216, "storageEngines" : [
"devnull", "ephemeralForTest", "mmapv1", "wiredTiger" ] } }
```

(실습 1)

(Quiz 1-1) 아래 db에 연결 후, 해당 collections을 선택 후, 내용을 확인해 보자.

admin

(Quiz 2. 빈칸 채우기) 아래와 같이 보도록 빈칸을 채워보자.

db.articles.find()._____

```
{
  "_id" : "DESKTOP-C8MCT7F-1504109488348",
  "hostname" : "DESKTOP-C8MCT7F",
  "startTime" : ISODate("2017-08-30T16:11:28Z"),
  "startTimeLocal" : "Thu Aug 31 01:11:28.348",
  "cmdLine" : {
  },
  ...
}
```

2-2. 기본 명령어 이해(2)

(가) 기본 명령어 연습

Shell command 참조 웹 사이트

<https://docs.mongodb.org/v3.0/reference/method/>

Collections을 생성하기

```
> db.createCollection("cars")
{ "ok" : 1 }
> show collections
cars
startup_log
```

cars 의 Collection 에 데이터 넣기, 확인

```
> db.cars.insert( {
  name : 'honda',
  make : 'accord',
  year : '2010'
})

> db.car.find().pretty()
{
  "_id" : ObjectId("59a6eab6f862e90eeda9093f"),
  "name" : "honda",
  "make" : "accord",
```

```
"year" : "2010"
}
```

cars 의 Collection 내의 데이터 바꾸기

```
db.car.update( {
  name : 'honda'
},
{ $set: {
  name:'ford'
}
})
```

[확인]

```
> db.car.find().pretty()
{
  "_id" : ObjectId("59a6eab6f862e90eeda9093f"),
  "name" : "ford",
  "make" : "accord",
  "year" : "2010"
}
>
```

(나) 반복문을 이용한 명령어 실습

```
for(var i=0; i<10; i++) { db.car.insert({"x":i}) }
db.car.find().pretty()
```

[실행]

```
> for(var i=0; i<10; i++) {db.car.insert({"x":i}) }
WriteResult({ "nInserted" : 1 })
```

```
> db.car.find().pretty()
{
  "_id" : ObjectId("59a6eab6f862e90eeda9093f"),
  "name" : "ford",
  "make" : "accord",
  "year" : "2010"
}
{ "_id" : ObjectId("59b53773de4c2066ef79532c"), "x" : 0 }
{ "_id" : ObjectId("59b53774de4c2066ef79532d"), "x" : 1 }
{ "_id" : ObjectId("59b53774de4c2066ef79532e"), "x" : 2 }
{ "_id" : ObjectId("59b53774de4c2066ef79532f"), "x" : 3 }
{ "_id" : ObjectId("59b53774de4c2066ef795330"), "x" : 4 }
{ "_id" : ObjectId("59b53774de4c2066ef795331"), "x" : 5 }
{ "_id" : ObjectId("59b53774de4c2066ef795332"), "x" : 6 }
{ "_id" : ObjectId("59b53774de4c2066ef795333"), "x" : 7 }
{ "_id" : ObjectId("59b53774de4c2066ef795334"), "x" : 8 }
{ "_id" : ObjectId("59b53774de4c2066ef795335"), "x" : 9 }
```

(실습 2)

x의 값을 11부터 20까지 넣어보자. 이를 확인해 보기

(생각해보기)

- (1) 중복 for문이 이용할 수 있을까?
- (2) 문자 형태로 넣을 수 있을까?

by DongJo, Lim