Getting Started with MongoDB

Objectives

- Access the MongoDB server using the command-line interface
- Describe the process of listing and creating collections, which contain documents, and databases, which contain one or more collections
- Perform basic operations on a collection such as inserting, counting and listing documents

Exercises

Start & Connect to Server

```
theia@theiadocker-craigtrupp8:/home/project$ start_mongo
Starting your mongodb database....
This process can take up to a minute.

Mongodb started, waiting for all services to be ready....

Your mongodb server is now ready to use and available with username: root
password: MTg1MDYtY3JhaWd0

You can access your mongodb database via:
 • The browser at:
https://craigtrupp8-8081.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cog
nitiveclass.ai
 • CommandLine: mongo -u root -p MTg1MDYtY3JhaWd0 --authenticationDatabase
admin local
theia@theiadocker-craigtrupp8:/home/project$ mongo -u root -p
MTg1MDYtY3JhaWd0 --authenticationDatabase admin local
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27017/local
MongoDB server version: 3.6.3
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
```

```
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
Server has startup warnings:
2023-10-12T21:16:40.953+0000 I STORAGE  [initandlisten]
2023-10-12T21:16:40.953+0000 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2023-10-12T21:16:40.953+0000 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2023-10-12T21:16:41.844+0000 I CONTROL  [initandlisten]
2023-10-12T21:16:41.844+0000 I CONTROL  [initandlisten] ** WARNING: You are running on a NUMA machine.
2023-10-12T21:16:41.844+0000 I CONTROL  [initandlisten] **          We suggest launching mongod like this to avoid performance problems:
2023-10-12T21:16:41.844+0000 I CONTROL  [initandlisten] **              numactl --interleave=all mongod [other options]
2023-10-12T21:16:41.844+0000 I CONTROL  [initandlisten]
>
```

Mongo Version, List Databases, Create New DB

```
> db.version()
3.6.3
> show dbs
admin   0.000GB
config  0.000GB
local   0.000GB
> use training
switched to db training
```

Create, Show, & Insert into Collections

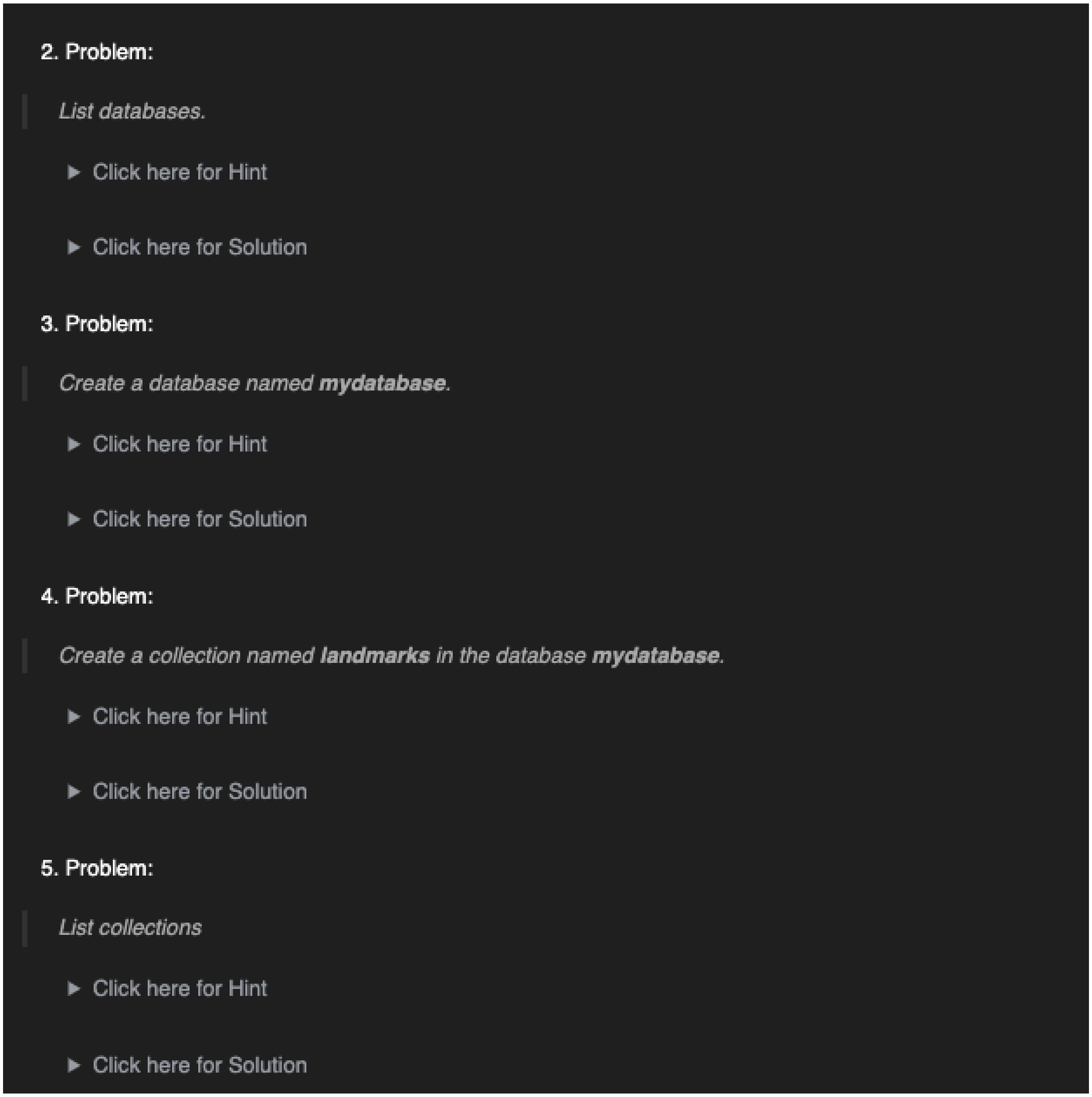
```
> db.createCollection("mycollection")
{ "ok" : 1 }
> show collections
mycollection
> db.mycollection.insert({"color":"white","example":"milk"})
WriteResult({ "nInserted" : 1 })
> db.mycollection.insert({"color":"blue","example":"sky"})
```

```
WriteResult({ "nInserted" : 1 })
> db.mycollection.insert({"color":"green", "example":"grass"})
WriteResult({ "nInserted" : 1 })
> db.mycollection.insert({"color":"grey", "example":"ocean"})
WriteResult({ "nInserted" : 1 })
> db.mycollection.insert({"color":"red", "example":"arsenal"})
WriteResult({ "nInserted" : 1 })
```

Count Documents, List Documents in Collection

```
> db.mycollection.count()
5
> db.mycollection.find()
{ "_id" : ObjectId("652863972f60334edc76730b"), "color" : "white",
"example" : "milk" }
{ "_id" : ObjectId("6528639c2f60334edc76730c"), "color" : "blue", "example"
: "sky" }
{ "_id" : ObjectId("652863bc2f60334edc76730d"), "color" : "green",
"example" : "grass" }
{ "_id" : ObjectId("652863d02f60334edc76730e"), "color" : "grey", "example"
: "ocean" }
{ "_id" : ObjectId("652863e22f60334edc76730f"), "color" : "red", "example"
: "arsenal" }
```

Practice exercises



2. Problem:

> *List databases.*

▶ Click here for Hint

▶ Click here for Solution

3. Problem:

> *Create a database named **mydatabase**.*

▶ Click here for Hint

▶ Click here for Solution

4. Problem:

> *Create a collection named **landmarks** in the database **mydatabase**.*

▶ Click here for Hint

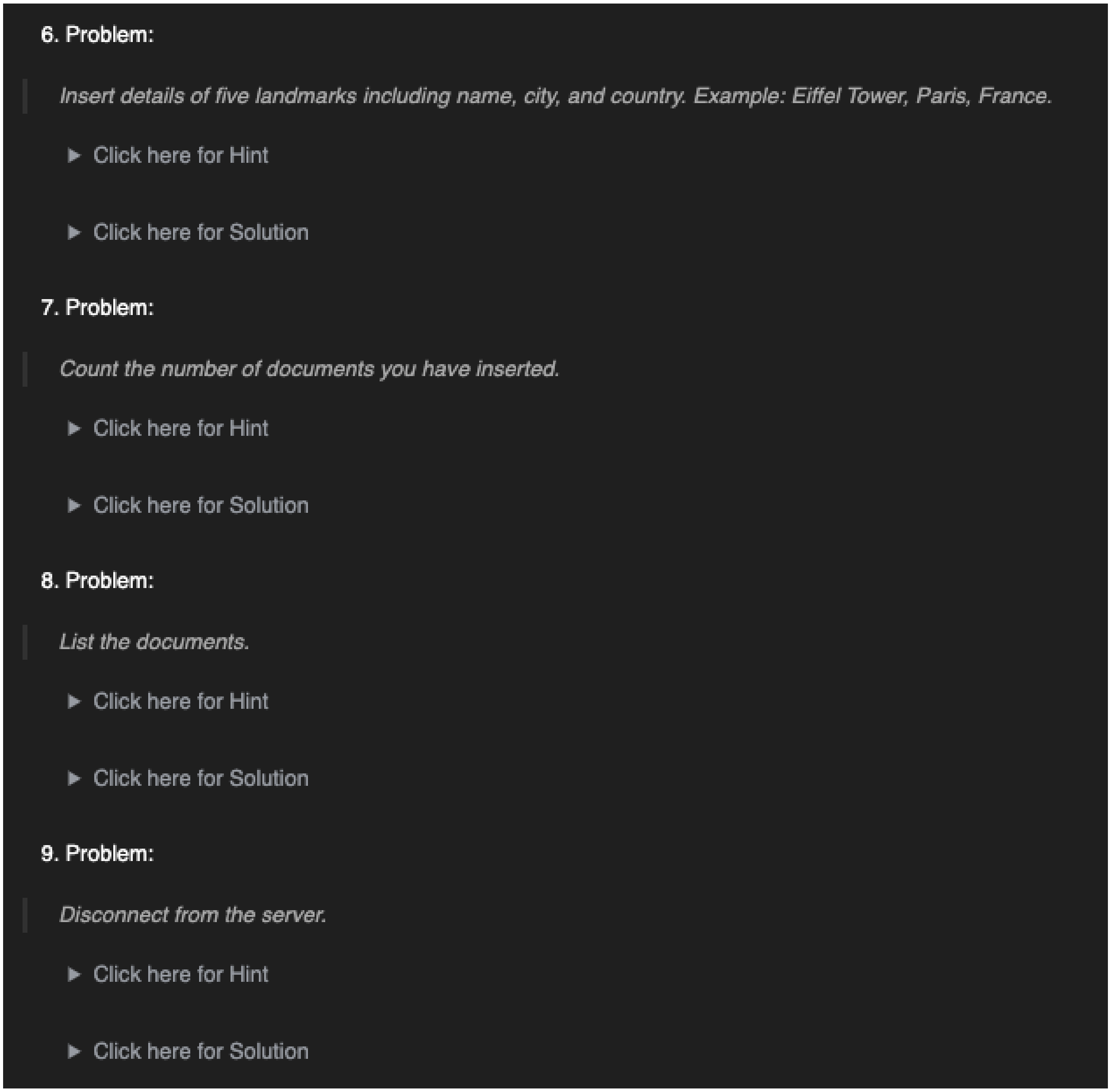
▶ Click here for Solution

5. Problem:

> *List collections*

▶ Click here for Hint

▶ Click here for Solution



6. Problem:

> *Insert details of five landmarks including name, city, and country. Example: Eiffel Tower, Paris, France.*

▶ Click here for Hint

▶ Click here for Solution

7. Problem:

> *Count the number of documents you have inserted.*

▶ Click here for Hint

▶ Click here for Solution

8. Problem:

> *List the documents.*

▶ Click here for Hint

▶ Click here for Solution

9. Problem:

> *Disconnect from the server.*

▶ Click here for Hint

▶ Click here for Solution

```
 show databases
admin     0.000GB
config    0.000GB
local     0.000GB
training  0.000GB
> use mydatabase
switched to db mydatabase
> show collections
> db.createCollection("landmarks")
```

```
{ "ok" : 1 }
> db.landmarks.insert({"name":"Eifel Tower", "city":"Paris",
"country":"France"})
WriteResult({ "nInserted" : 1 })
> db.landmarks.find()
{ "_id" : ObjectId("652865852f60334edc767310"), "name" : "Eifel Tower",
"city" : "Paris", "country" : "France" }
> db.landmarks.insert({"nomber":"Tenatchitlan", "ciudad": "Mexico City",
"pais":"Mexico"}_
... )
2023-10-12T17:31:43.559-0400 E QUERY    [thread1] SyntaxError: missing )
after argument list @(shell):1:87
> db.landmarks.insert({"nomber":"Tenatchitlan", "ciudad": "Mexico City",
"pais":"Mexico"}_
... ^C

> db.landmarks.insert({"nomber":"Tenatchitlan", "ciudad": "Mexico City",
"pais":"Mexico"})
WriteResult({ "nInserted" : 1 })
> db.landmarks.find()
{ "_id" : ObjectId("652865852f60334edc767310"), "name" : "Eifel Tower",
"city" : "Paris", "country" : "France" }
{ "_id" : ObjectId("652865c82f60334edc767311"), "nomber" : "Tenatchitlan",
"ciudad" : "Mexico City", "pais" : "Mexico" }
> db.landmarks.count()
2
```