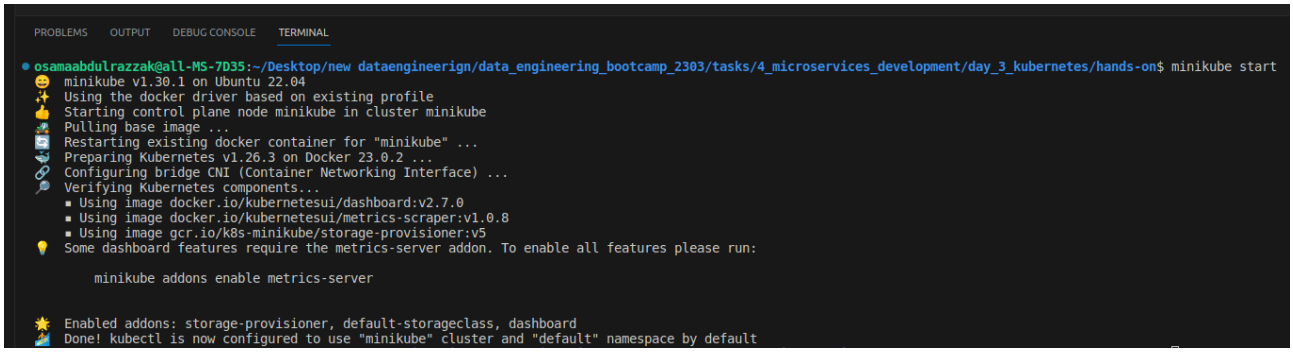
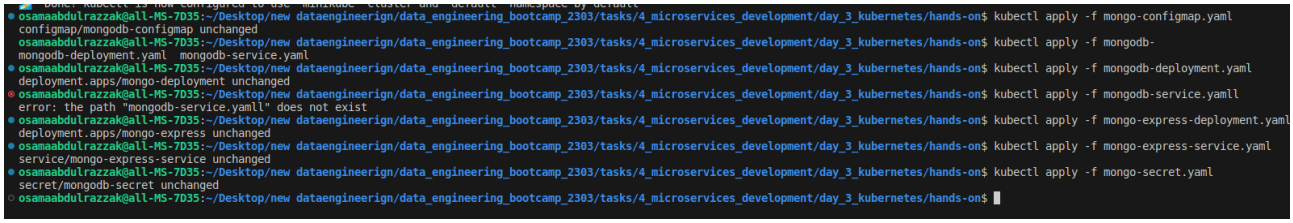
***Name: Osama Abdul Razzak(2303.KHI.DEG.029)***
***Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)***

# Assignment: 4.3

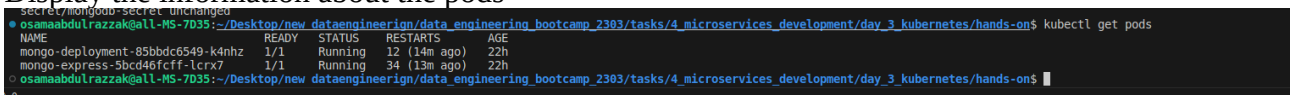
Firstly, we start single-node cluster using Minikube. It will create a virtual machine and configure it to run the kubernetes cluster locally



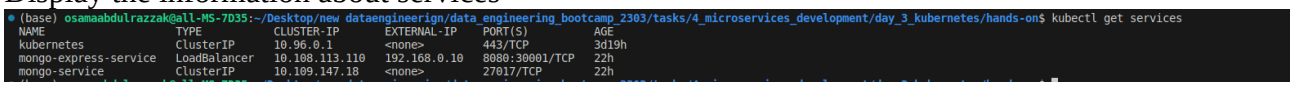

Apply the kubernetes file for setting up a mongoDB deployment with mongo express and related service




Display the information about the pods

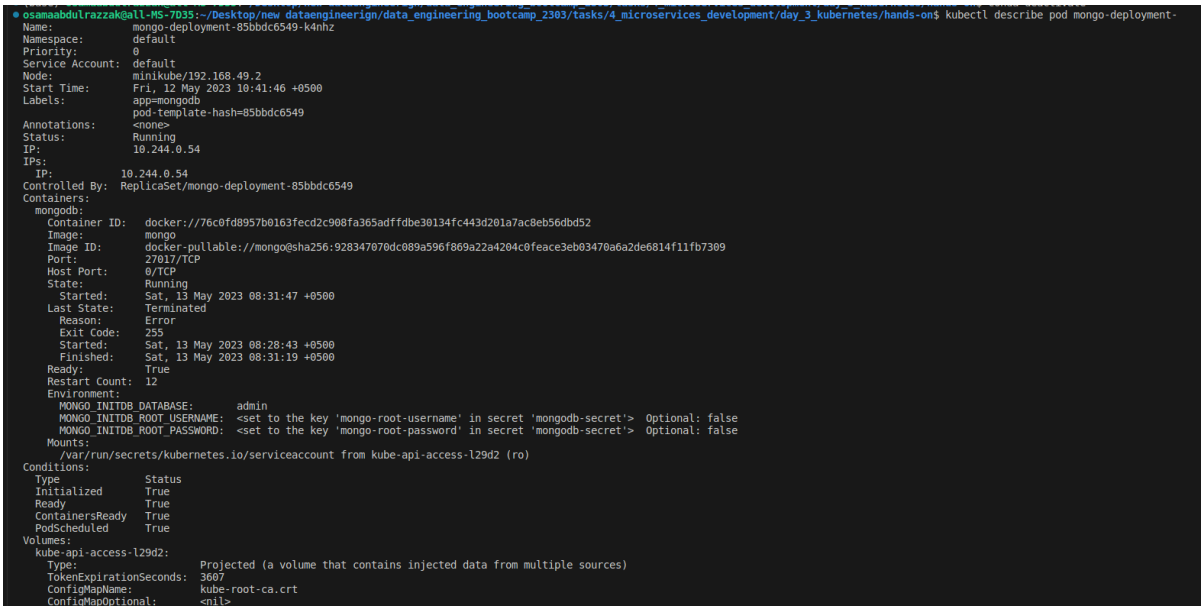



Display the information about services



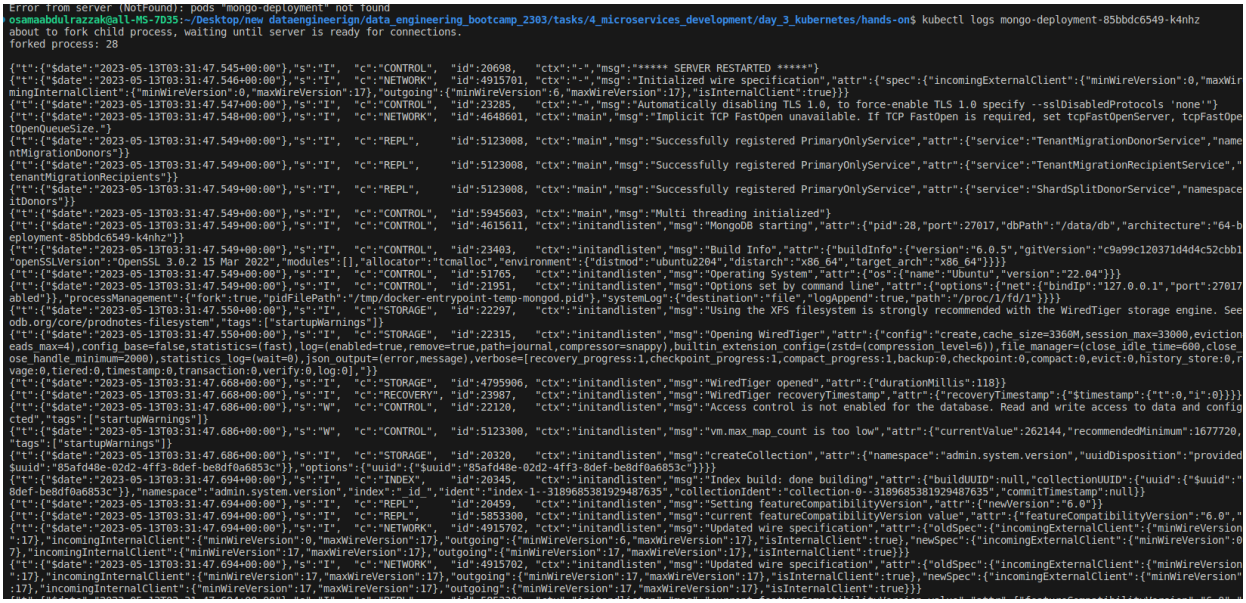
***Name: Osama Abdul Razzak(2303.KHI.DEG.029)***
***Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)***

For describing the specific pod, we use following command



```
osamaabdulrazzak@all-MS-7D35:~/Desktop/new dataengineerign/data_engineering_bootcamp_2303/tasks/4_microservices_development/day_3_kubernetes/hands-on$ kubectl describe pod mongo-deployment-
Name:             mongo-deployment-85bbdc6549-k4nhz
Namespace:        default
Priority:         0
Service Account:  default
Node:             minikube/192.168.49.2
Start Time:       Fri, 12 May 2023 10:41:46 +0500
Labels:           app=mongodb
                  pod-template-hash=85bbdc6549
Annotations:      <none>
Status:           Running
IP:               10.244.0.54
IPs:
  IP:           10.244.0.54
Controlled By:  ReplicaSet/mongo-deployment-85bbdc6549
Containers:
  mongodb:
    Container ID:   docker://76c0fd8957b0163fecd2c908fa365adffdbe30134fc443d201a7ac8eb56dbd52
    Image:          mongo
    Image ID:       docker-pullable://mongo@sha256:928347070dc089a596f869a22a4204c0feace3eb03470a6a2de6814f11fb7309
    Port:           27017/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Sat, 13 May 2023 08:31:47 +0500
    Last State:     Terminated
      Reason:       Error
      Exit Code:    255
      Started:      Sat, 13 May 2023 08:28:43 +0500
      Finished:     Sat, 13 May 2023 08:31:19 +0500
    Ready:          True
    Restart Count:  12
    Environment:
      MONGO_INITDB_DATABASE:       admin
      MONGO_INITDB_ROOT_USERNAME:  <set to the key 'mongo-root-username' in secret 'mongodb-secret'>  Optional: false
      MONGO_INITDB_ROOT_PASSWORD:  <set to the key 'mongo-root-password' in secret 'mongodb-secret'>  Optional: false
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-l29d2 (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-l29d2:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
```

For describing specific logs, we use the following command



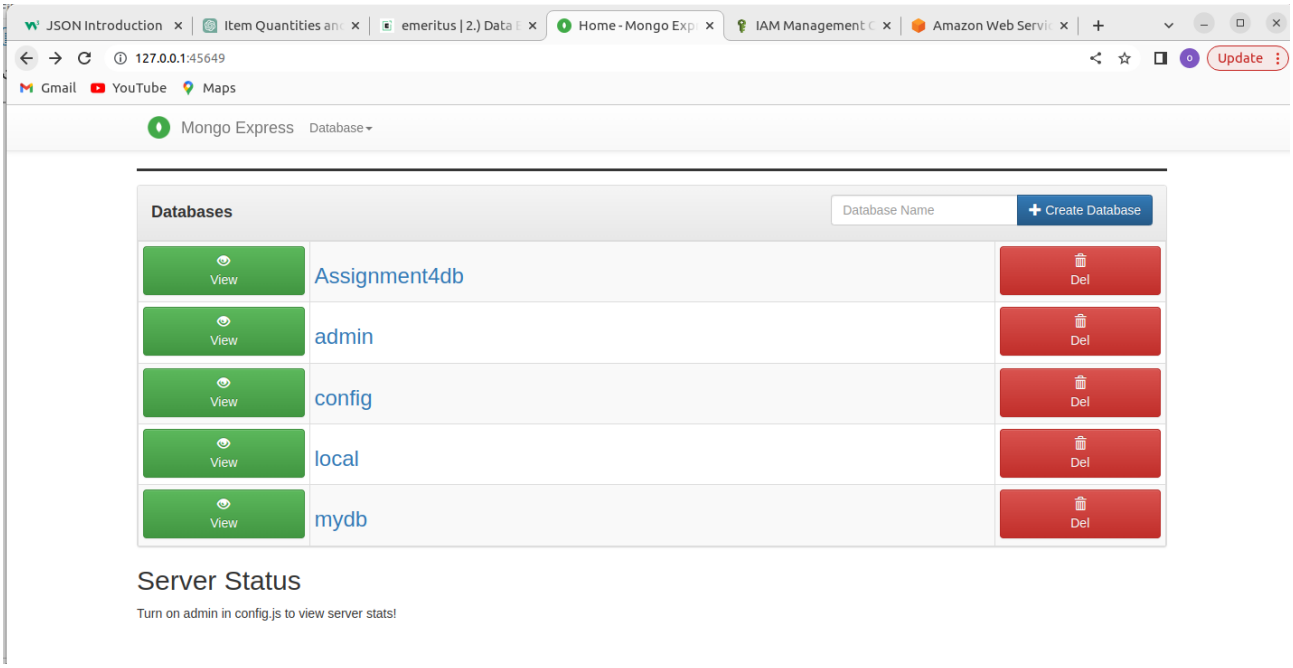
```
Error from server (NotFound): pods "mongo-deployment" not found
osamaabdulrazzak@all-MS-7D35:~/Desktop/new dataengineerign/data_engineering_bootcamp_2303/tasks/4_microservices_development/day_3_kubernetes/hands-on$ kubectl logs mongo-deployment-85bbdc6549-k4nhz
about to fork child process, waiting until server is ready for connections.
forked process: 28

{"t":{"$date":"2023-05-13T03:31:47.545+00:00"},"s":"I",  "c":"CONTROL",  "id":20698,   "ctx":"-","msg":"***** SERVER RESTARTED *****"}
{"t":{"$date":"2023-05-13T03:31:47.546+00:00"},"s":"I",  "c":"NETWORK",  "id":4915701, "ctx":"-","msg":"Initialized wire specification","attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWir
mingInternalClient":{"minWireVersion":0,"maxWireVersion":17},"outgoing":{"minWireVersion":6,"maxWireVersion":17},"isInternalClient":true}}}
{"t":{"$date":"2023-05-13T03:31:47.547+00:00"},"s":"I",  "c":"CONTROL",  "id":23285,   "ctx":"-","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2023-05-13T03:31:47.548+00:00"},"s":"I",  "c":"NETWORK",  "id":4648601, "ctx":"main","msg":"Implicit TCP FastOpen unavailable. If TCP FastOpen is required, set tcpFastOpenServer, tcpFastOpe
tOpenQueueSize."}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationDonorService","name
ntMigrationDonors"}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigrationRecipientService","
tenantMigrationRecipients"}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"ShardSplitDonorService","namespace
itDonors"}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"CONTROL",  "id":5945603, "ctx":"main","msg":"Multi threading initialized"}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"CONTROL",  "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":28,"port":27017,"dbPath":"/data/db","architecture":"64-b
eployment-85bbdc6549-k4nhz"}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"CONTROL",  "id":23403,   "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"6.0.5","gitVersion":"c9a99c120371d4d4c52cbb1
"openSSLVersion":"OpenSSL 3.0.2 15 Mar 2022","modules":[],"allocator":"tcmalloc","environment":{"distmod":"ubuntu2204","distarch":"x86_64","target_arch":"x86_64"}}}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"CONTROL",  "id":51765,   "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Ubuntu","version":"22.04"}}}
{"t":{"$date":"2023-05-13T03:31:47.549+00:00"},"s":"I",  "c":"CONTROL",  "id":21951,   "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{"net":{"bindIp":"127.0.0.1","port":27017
abled"}},"processManagement":{"fork":true,"pidFilePath":"/tmp/docker-entrypoint-temp-mongod.pid"},"systemLog":{"destination":"file","logAppend":true,"path":"/proc/1/fd/1"}}}}
{"t":{"$date":"2023-05-13T03:31:47.550+00:00"},"s":"I",  "c":"STORAGE",  "id":22297,   "ctx":"initandlisten","msg":"Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See
odb.org/core/prodnotes-filesystem","tags":["startupWarnings"]}
{"t":{"$date":"2023-05-13T03:31:47.550+00:00"},"s":"I",  "c":"STORAGE",  "id":22315,   "ctx":"initandlisten","msg":"Opening WiredTiger","attr":{"config":"create,cache_size=3360M,session_max=33000,eviction
eads_max=4),config_base=false,statistics=(fast),log=(enabled=true,remove=true,path=journal,compressor=snappy),builtin_extension_config=(zstd=(compression_level=6)),file_manager=(close_idle_time=600,close_
ose_handle_minimum=2000),statistics_log=(wait=0),json_output=(error,message),verbose=[recovery_progress:1,checkpoint_progress:1,compact_progress:1,backup:0,checkpoint:0,compact:0,evict:0,history_store:0,r
vage:0,tiered:0,timestamp:0,transaction:0,verify:0,log:0],"}}
{"t":{"$date":"2023-05-13T03:31:47.668+00:00"},"s":"I",  "c":"STORAGE",  "id":4795906, "ctx":"initandlisten","msg":"WiredTiger opened","attr":{"durationMillis":118}}
{"t":{"$date":"2023-05-13T03:31:47.668+00:00"},"s":"I",  "c":"RECOVERY", "id":23987,   "ctx":"initandlisten","msg":"WiredTiger recoveryTimestamp","attr":{"recoveryTimestamp":{"$timestamp":{"t":0,"i":0}}}}
{"t":{"$date":"2023-05-13T03:31:47.686+00:00"},"s":"W",  "c":"CONTROL",  "id":22120,   "ctx":"initandlisten","msg":"Access control is not enabled for the database. Read and write access to data and config
cted","tags":["startupWarnings"]}
{"t":{"$date":"2023-05-13T03:31:47.686+00:00"},"s":"W",  "c":"CONTROL",  "id":5123300, "ctx":"initandlisten","msg":"vm.max_map_count is too low","attr":{"currentValue":262144,"recommendedMinimum":1677720,
"tags":["startupWarnings"]}
{"t":{"$date":"2023-05-13T03:31:47.686+00:00"},"s":"I",  "c":"STORAGE",  "id":20320,   "ctx":"initandlisten","msg":"createCollection","attr":{"namespace":"admin.system.version","uuidDisposition":"provided
$uuid":"85afd48e-02d2-4ff3-8def-be8df0a6853c"},"options":{"uuid":{"$uuid":"85afd48e-02d2-4ff3-8def-be8df0a6853c"}}}}
{"t":{"$date":"2023-05-13T03:31:47.694+00:00"},"s":"I",  "c":"INDEX",    "id":20345,   "ctx":"initandlisten","msg":"Index build: done building","attr":{"buildUUID":null,"collectionUUID":{"uuid":{"$uuid":"
8def-be8df0a6853c"}},"namespace":"admin.system.version","index":"_id_","ident":"index-1--3189685381929487635","collectionIdent":"collection-0--3189685381929487635","commitTimestamp":null}}
{"t":{"$date":"2023-05-13T03:31:47.694+00:00"},"s":"I",  "c":"REPL",     "id":20459,   "ctx":"initandlisten","msg":"Setting featureCompatibilityVersion","attr":{"newVersion":"6.0"}}
{"t":{"$date":"2023-05-13T03:31:47.694+00:00"},"s":"I",  "c":"REPL",     "id":5853300, "ctx":"initandlisten","msg":"current featureCompatibilityVersion value","attr":{"featureCompatibilityVersion":"6.0","
{"t":{"$date":"2023-05-13T03:31:47.694+00:00"},"s":"I",  "c":"NETWORK",  "id":4915702, "ctx":"initandlisten","msg":"Updated wire specification","attr":{"oldSpec":{"incomingExternalClient":{"minWireVersion
":17},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":17},"outgoing":{"minWireVersion":6,"maxWireVersion":17},"isInternalClient":true},"newSpec":{"incomingExternalClient":{"minWireVersion":0
7},"incomingInternalClient":{"minWireVersion":17,"maxWireVersion":17},"outgoing":{"minWireVersion":17,"maxWireVersion":17},"isInternalClient":true}}}
{"t":{"$date":"2023-05-13T03:31:47.694+00:00"},"s":"I",  "c":"NETWORK",  "id":4915702, "ctx":"initandlisten","msg":"Updated wire specification","attr":{"oldSpec":{"incomingExternalClient":{"minWireVersion
":17},"incomingInternalClient":{"minWireVersion":17,"maxWireVersion":17},"outgoing":{"minWireVersion":17,"maxWireVersion":17},"isInternalClient":true},"newSpec":{"incomingExternalClient":{"minWireVersion
:17},"incomingInternalClient":{"minWireVersion":17,"maxWireVersion":17},"outgoing":{"minWireVersion":17,"maxWireVersion":17},"isInternalClient":true}}}
```

***Name: Osama Abdul Razzak(2303.KHI.DEG.029)***
***Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)***

To access a service running on Minikube, we use the following command

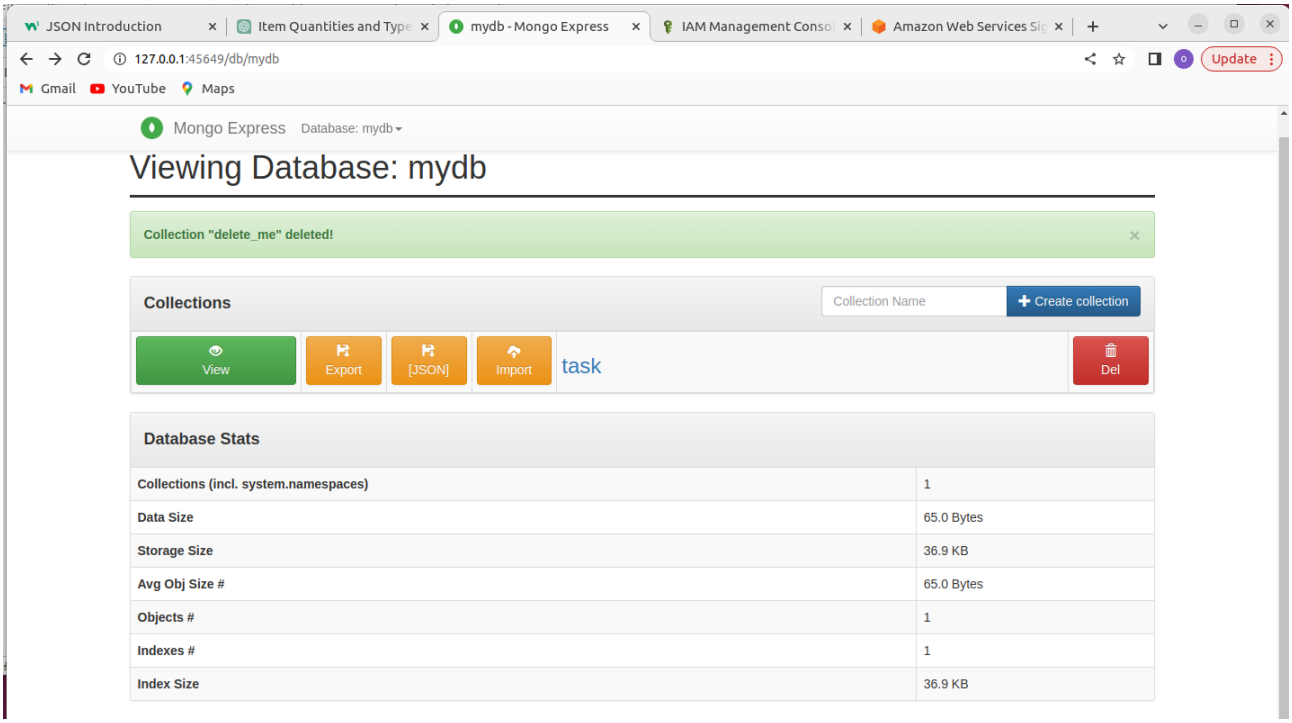
```
osamaabdulrazzak@all-MS-7D35:~/Desktop/new dataengineerign/data_engineering_bootcamp_2303/tasks/4_microservices_development/day_3_kubernetes/hands-on$ minikube service  mongo-express-service
|-----------|-----------------------|-------------|---------------------------|
| NAMESPACE |         NAME          | TARGET PORT |            URL            |
|-----------|-----------------------|-------------|---------------------------|
| default   | mongo-express-service |        8080 | http://192.168.49.2:30001 |
|-----------|-----------------------|-------------|---------------------------|
🏃  Starting tunnel for service mongo-express-service.
|-----------|-----------------------|-------------|------------------------|
| NAMESPACE |         NAME          | TARGET PORT |          URL           |
|-----------|-----------------------|-------------|------------------------|
| default   | mongo-express-service |             | http://127.0.0.1:45649 |
|-----------|-----------------------|-------------|------------------------|
🎉  Opening service default/mongo-express-service in default browser...
/snap/core20/current/lib/x86_64-linux-gnu/libstdc++.so.6: version `GLIBCXX_3.4.29' not found (required by /lib/x86_64-linux-gnu/libproxy.so.1)
Failed to load module: /home/osamaabdulrazzak/snap/code/common/.cache/gio-modules/libgiolibproxy.so
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
Opening in existing browser session.
MESA-INTEL: warning: Performance support disabled, consider sysctl dev.i915.perf_stream_paranoid=0
```

and then we navigate to mongo service url
and create the new database ‘mydb’

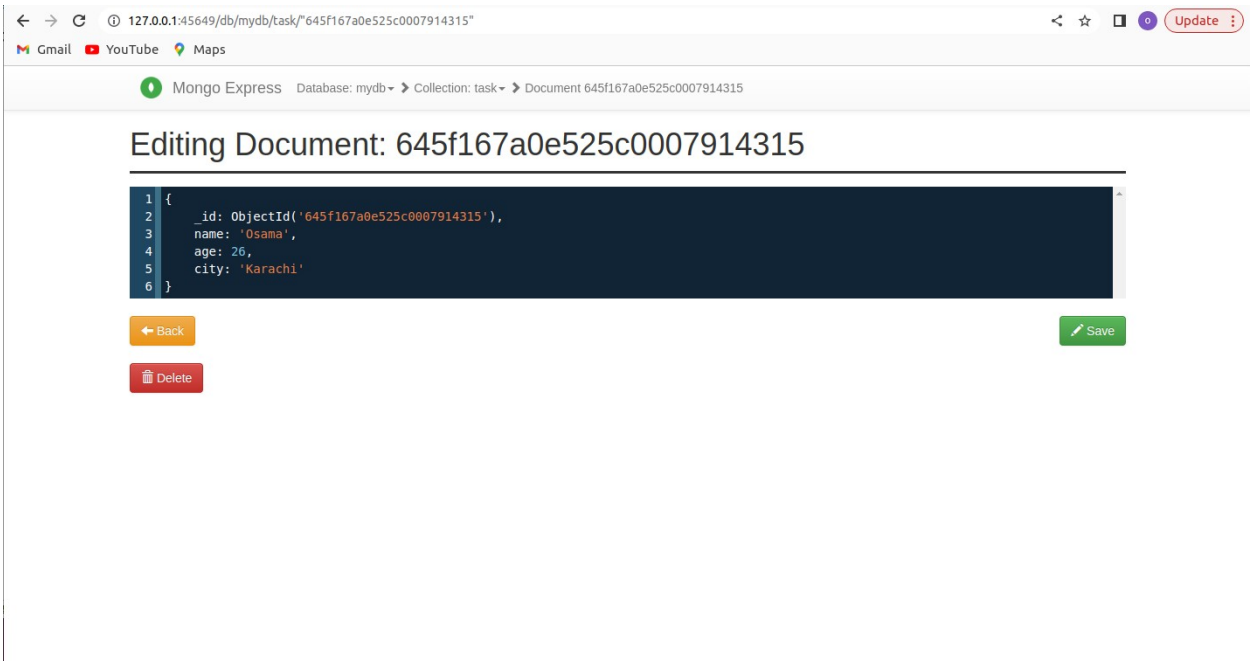


*Name: Osama Abdul Razzak(2303.KHI.DEG.029)*
*Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)*

and create collection ‘task_1’ in it

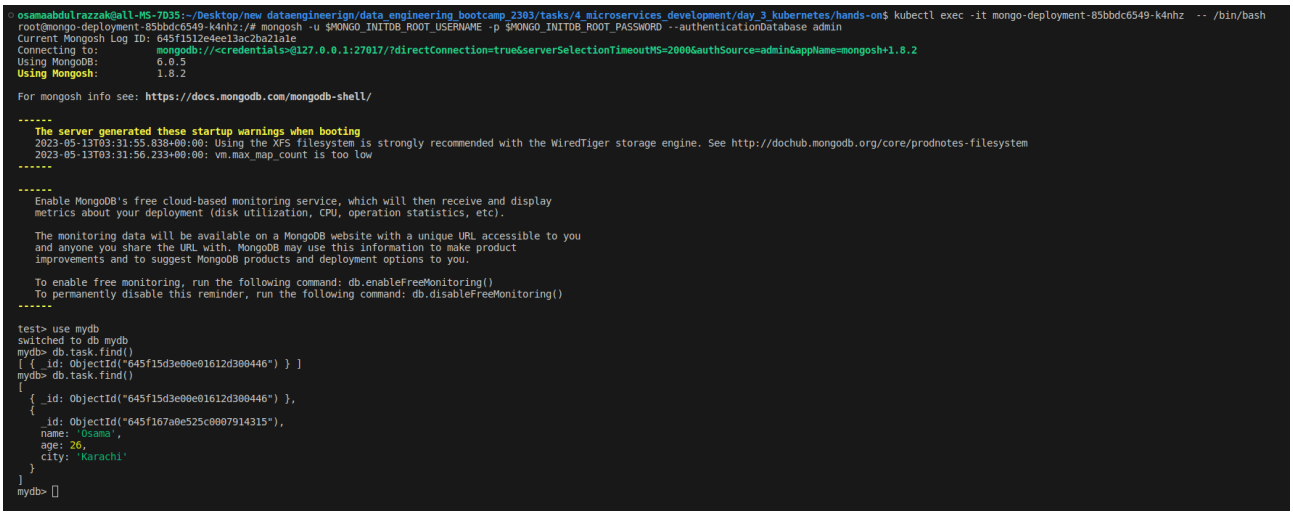



Edit the document



***Name: Osama Abdul Razzak(2303.KHI.DEG.029)***
***Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)***

then enter into the pod and move into mongo shell environment by typing Mongosh and write no sql query for accessing collection




we successfully access the collection