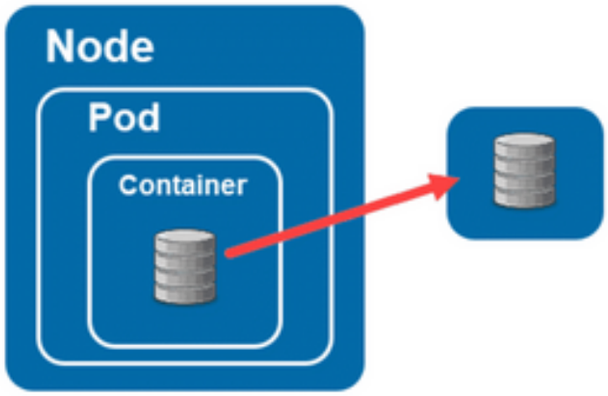
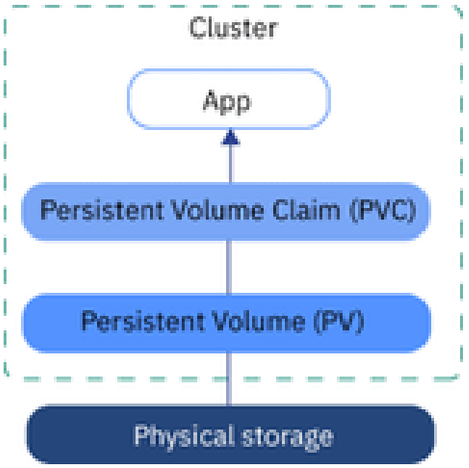
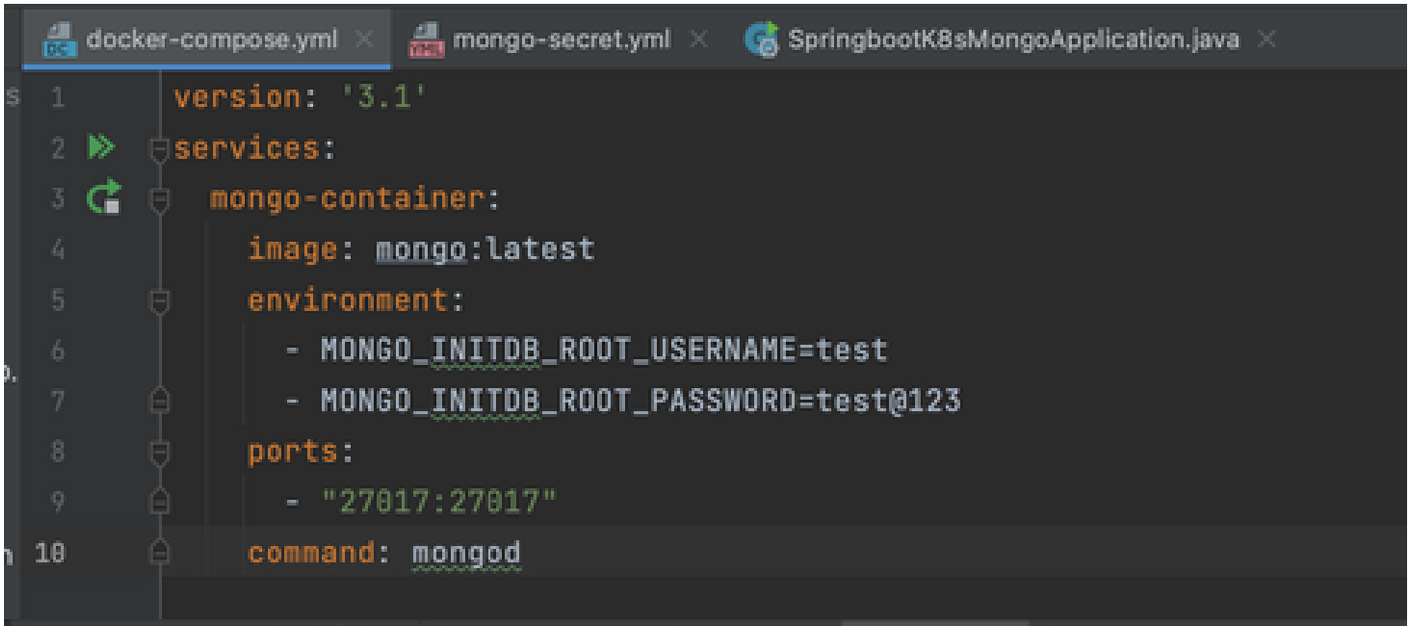
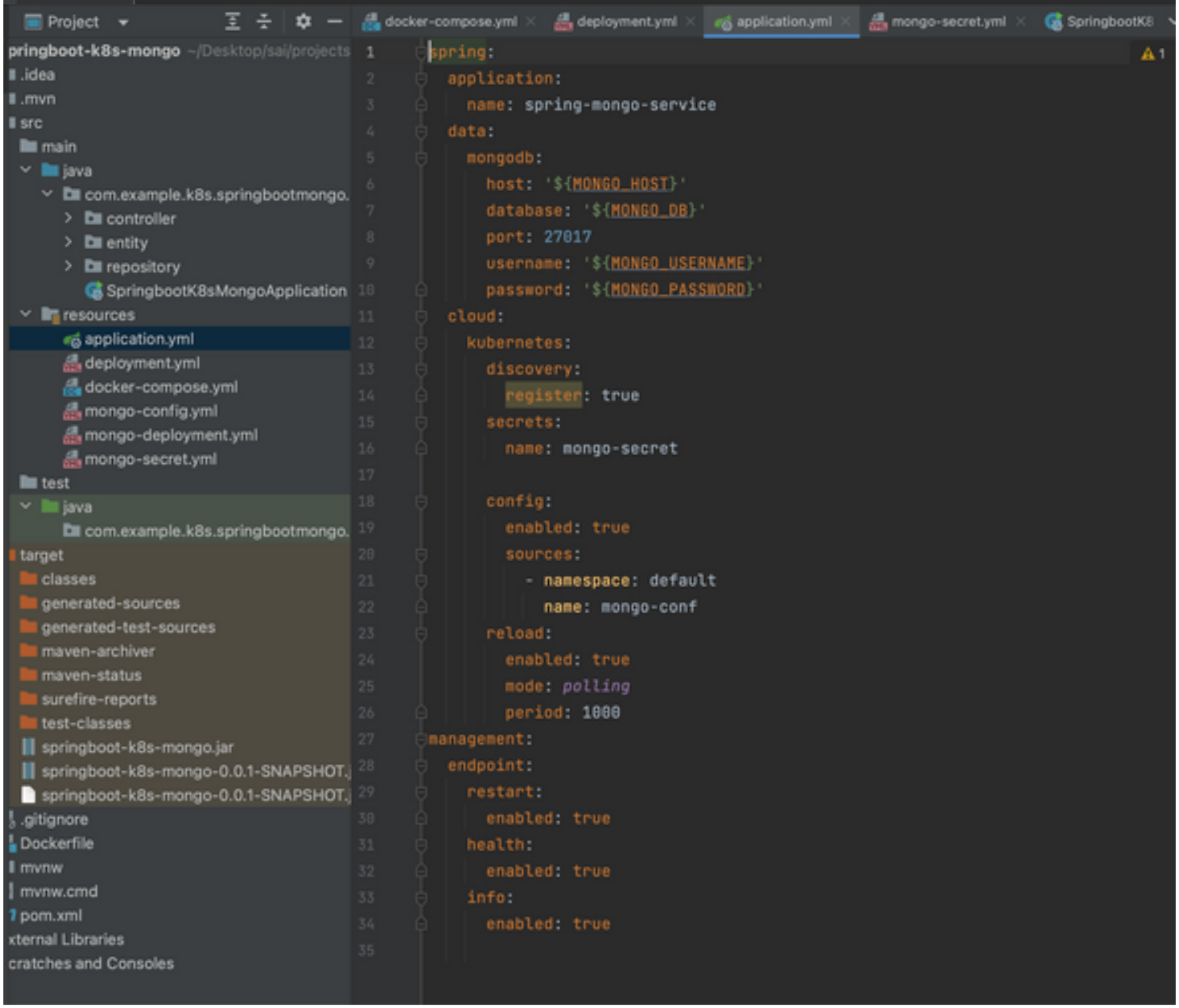




Kubernetes Persistent Volume Claim & MongoDB ConfigMap


docker-compose.yml
mongo-secret.yml
SpringbootK8sMongoApplication.java
version: '3.1'
services:
mongo-container:
image: mongo:latest
environment:
- MONGO_INITDB_ROOT_USERNAME=test
- MONGO_INITDB_ROOT_PASSWORD=test@123
ports:
- "27017:27017"
command: mongod


Project
springboot-k8s-mongo ~/Desktop/sai/projects
.idea
.mvn
src
main
java
com.example.k8s.springbootmongo.
controller
entity
repository
SpringbootK8sMongoApplication
resources
application.yml
deployment.yml
docker-compose.yml
mongo-config.yml
mongo-deployment.yml
mongo-secret.yml
test
java
com.example.k8s.springbootmongo.
target
classes
generated-sources
generated-test-sources
maven-archiver
maven-status
surefire-reports
test-classes
springboot-k8s-mongo.jar
springboot-k8s-mongo-0.0.1-SNAPSHOT.
springboot-k8s-mongo-0.0.1-SNAPSHOT.
.gitignore
Dockerfile
mvnw
mvnw.cmd
pom.xml
xternal Libraries
cratches and Consoles
docker-compose.yml
deployment.yml
application.yml
mongo-secret.yml
spring:
application:
name: spring-mongo-service
data:
mongodb:
host: '${MONGO_HOST}'
database: '${MONGO_DB}'
port: 27017
username: '${MONGO_USERNAME}'
password: '${MONGO_PASSWORD}'
cloud:
kubernetes:
discovery:
register: true
secrets:
name: mongo-secret
config:
enabled: true
sources:
- namespace: default
name: mongo-conf
reload:
enabled: true
mode: polling
period: 1000
management:
endpoint:
restart:
enabled: true
health:
enabled: true
info:
enabled: true

docker-compose.yml | deployment.yml | application.yml | mongo-secret.yml | SpringbootK8sMongoApplication.java

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: spring-mongo-service
spec:
  selector:
    matchLabels:
      app: spring-mongo-service
  replicas: 2
  template:
    metadata:
      labels:
        app: spring-mongo-service
    spec:
      containers:
        - name: spring-mongo-service
          image: spring-mongo-service:1.0
          imagePullPolicy: Never
          ports:
            - containerPort: 8080
          env:
            - name: MONGO_USERNAME
              valueFrom:
                secretKeyRef:
                  name: mongo-secret
                  key: username
            - name: MONGO_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: mongo-secret
                  key: password
            - name: MONGO_DB
              valueFrom:
                configMapKeyRef:
                  name: mongo-conf
                  key: database
            - name: MONGO_HOST
              valueFrom:
                configMapKeyRef:
                  name: mongo-conf
                  key: host
---
kind: Service
apiVersion: v1
metadata:
  name: spring-mongo-service
spec:
  selector:
    app: spring-mongo-service
  ports:
  - protocol: TCP
    port: 8080
    nodePort: 30081
  type: NodePort
```

docker-compose.yml | deployment.yml | application.yml | mongo-secret.yml | SpringbootK8sMongoApplication.java

```yaml
apiVersion: v1
kind: Secret
metadata:
  name: mongo-secret
data:
  username: dGVzdA==
  password: dGVzdEAxMjM=
```

```
echo -n 'test' | base64
dGVzdA==

echo -n 'test@123' | base64
dGVzdEAxMjM=
```

ment.yml | application.yml | mongo-secret.yml | mongo-deployment.yml | mongo-config.yml

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: mongo-conf
data:
  host: mongodb-service
  database: admin
```

mpose.yml | deployment.yml | application.yml | mongo-secret.yml | mongo-deployment.yml | mongo-config.yml | SpringbootK8sMongo

```yaml
apiVersion: v1
kind: Service
metadata:
  labels:
    app: mongo
  name: mongodb-service
spec:
  ports:
  - port: 27017
    targetPort: 27017
  selector:
    app: mongo
  clusterIP: None  # We Use DNS, Thus ClusterIP is not relevant
```

```yaml
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mongo-pv-claim # name of PVC essential for identifying the storage data
  labels:
    app: mongo
    tier: database
spec:
  accessModes:
    - ReadWriteOnce   #This specifies the mode of the claim that we are trying to create.
  resources:
    requests:
      storage: 1Gi    #This will tell kubernetes about the amount of space we are trying to claim.
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: mongo
  labels:
    app: mongo
spec:
  selector:
    matchLabels:
      app: mongo
  replicas: 1
  template:
    metadata:
      labels:
        app: mongo
      name: mongodb-service
    spec:
      containers:
      - image: mongo:latest
        name: mongo
```

```yaml
    env:
      - name: MONGO_INITDB_ROOT_USERNAME
        valueFrom:
          secretKeyRef:
            name: mongo-secret
            key: username
      - name: MONGO_INITDB_ROOT_PASSWORD
        valueFrom:
          secretKeyRef:
            name: mongo-secret
            key: password
    ports:
    - containerPort: 27017
      name: mongo
    volumeMounts:
    - name: mongo-persistent-storage
      mountPath: /data/db #This is the path in the container on which the mounting will take place.
  volumes:
  - name: mongo-persistent-storage # Obtaining 'vloume' from PVC
    persistentVolumeClaim:
      claimName: mongo-pv-claim
```

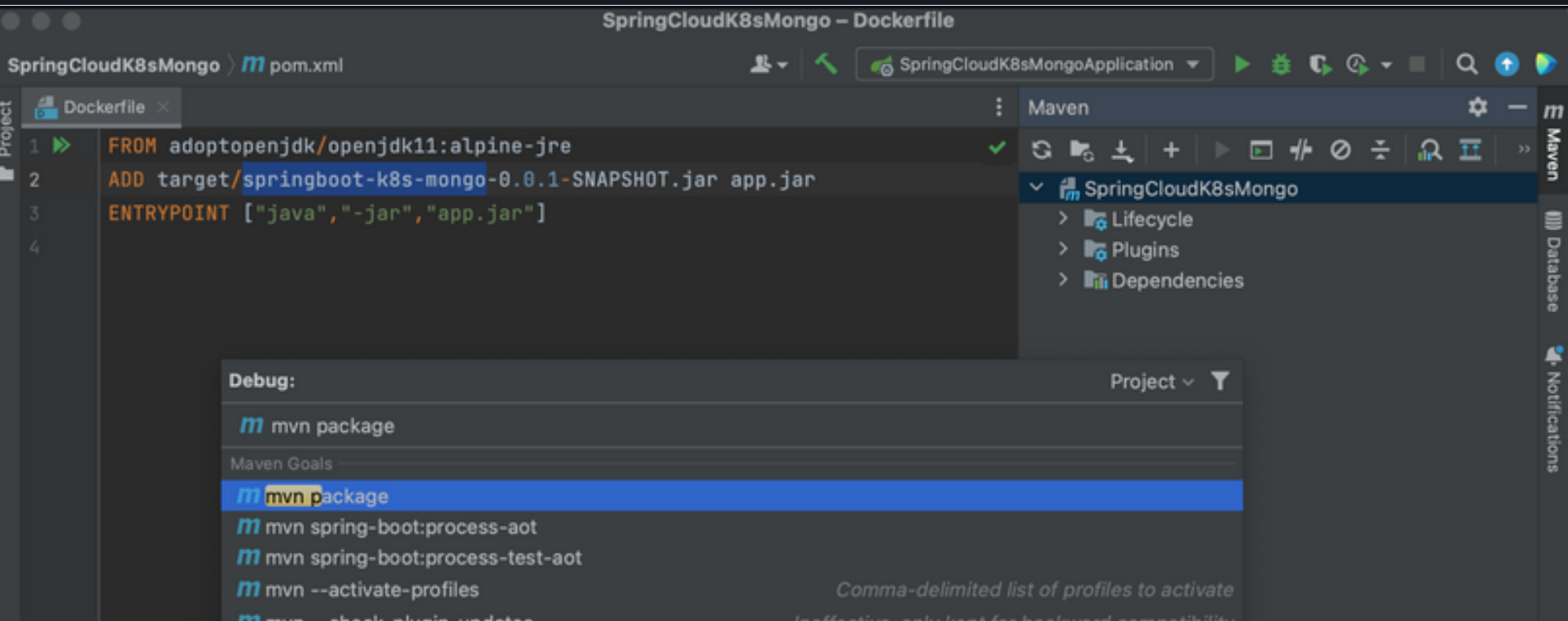
```java
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

1 usage
@SpringBootApplication
@EnableDiscoveryClient
public class SpringbootK8sMongoApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringbootK8sMongoApplication.class, args);
    }

}
```

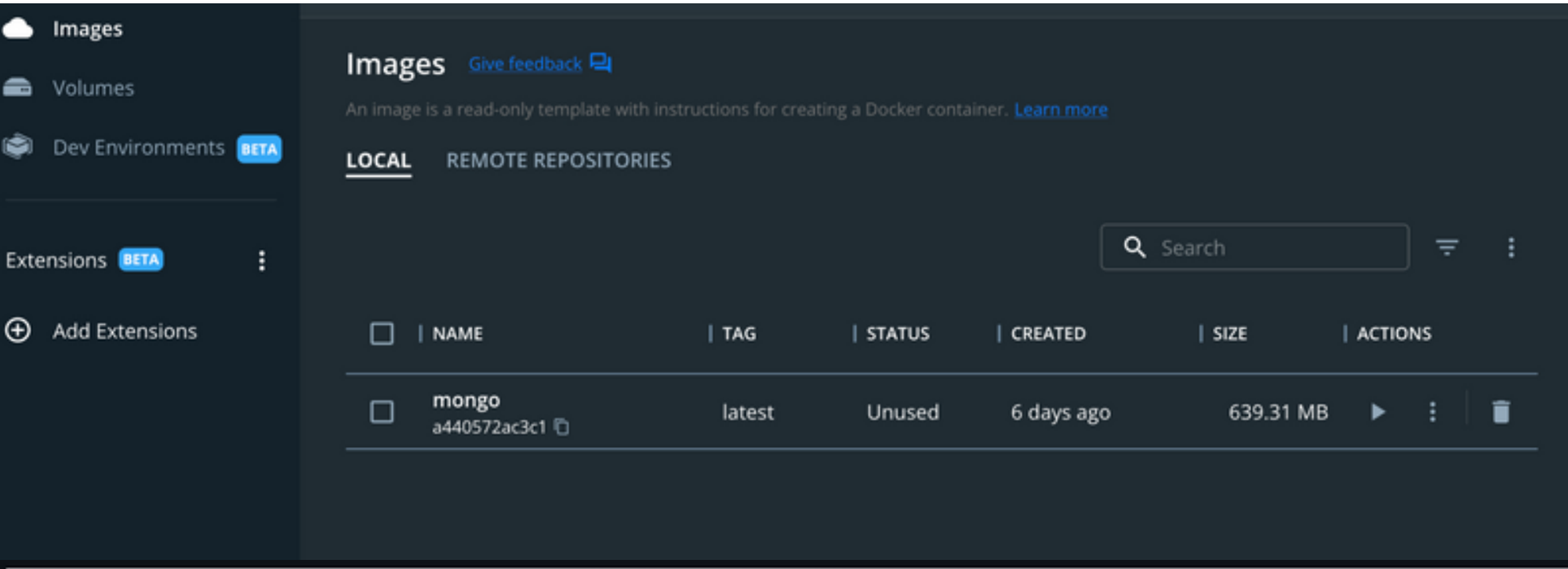
```
saiashish@MBP-C02F5ASXMD6M resources % docker compose up
[+] Running 10/10
 ⠿ mongo-container Pulled                                                                        41.0s
   ⠿ 10ac4908093d Pull complete                                                                   6.0s
   ⠿ 685504465d09 Pull complete                                                                   6.1s
   ⠿ ebd36404f329 Pull complete                                                                   6.4s
   ⠿ 3abd9b25affb Pull complete                                                                   6.5s
   ⠿ 2d7fde532eae Pull complete                                                                   6.5s
   ⠿ 24fc70e4c7d7 Pull complete                                                                   6.6s
   ⠿ ffc2353072f7 Pull complete                                                                   6.6s
   ⠿ 560de8e3a6c7 Pull complete                                                                  36.4s
   ⠿ 0748cd1d792c Pull complete                                                                  36.4s
[+] Running 2/2
 ⠿ Network resources_default              Created                                                 0.1s
 ⠿ Container resources-mongo-container-1  Created                                                 1.2s
Attaching to resources-mongo-container-1
resources-mongo-container-1  | about to fork child process, waiting until server is ready for connections.
resources-mongo-container-1  | forked process: 28
resources-mongo-container-1  |
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.828+00:00"},"s":"I",  "c":"CONTROL",  "id":20698,   "ctx":"-","msg":"***** SERVER RESTARTED *****"}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.832+00:00"},"s":"I",  "c":"NETWORK",  "id":4915701, "ctx":"-","msg":"Initialized wire specification","attr":{"spec":{"incomin
rsion":0,"maxWireVersion":17},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":17},"outgoing":{"minWireVersion":6,"maxWireVersion":17},"isInternalClient":true}}}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.833+00:00"},"s":"I",  "c":"CONTROL",  "id":23285,   "ctx":"-","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0
ols 'none'"}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.834+00:00"},"s":"I",  "c":"NETWORK",  "id":4648601, "ctx":"main","msg":"Implicit TCP FastOpen unavailable. If TCP FastOpen is
rver, tcpFastOpenClient, and tcpFastOpenQueueSize."}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.835+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"
orService","namespace":"config.tenantMigrationDonors"}}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.835+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"
ipientService","namespace":"config.tenantMigrationRecipients"}}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.835+00:00"},"s":"I",  "c":"REPL",     "id":5123008, "ctx":"main","msg":"Successfully registered PrimaryOnlyService","attr":{"
vice","namespace":"config.tenantSplitDonors"}}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.836+00:00"},"s":"I",  "c":"CONTROL",  "id":5945603, "ctx":"main","msg":"Multi threading initialized"}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.837+00:00"},"s":"I",  "c":"CONTROL",  "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":28,"port":27
hitecture":"64-bit","host":"f15215b6e8df"}}
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.837+00:00"},"s":"I",  "c":"CONTROL",  "id":23403,   "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"
61c1353638a71e710f385a566bcd2f547","openSSLVersion":"OpenSSL 3.0.2 15 Mar 2022","modules":[],"allocator":"tcmalloc","environment":{"distmod":"ubuntu2204","distarch":"x86_64","target_arch":"x8
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.837+00:00"},"s":"I",  "c":"CONTROL",  "id":51765,   "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Ubun
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.837+00:00"},"s":"I",  "c":"CONTROL",  "id":21951,   "ctx":"initandlisten","msg":"Options set by command line","attr":{"option
.1","port":27017,"tls":{"mode":"disabled"}},"processManagement":{"fork":true,"pidFilePath":"/tmp/docker-entrypoint-temp-mongod.pid"},"systemLog":{"destination":"file","logAppend":true,"path":
resources-mongo-container-1  | {"t":{"$date":"2023-02-14T19:39:18.838+00:00"},"s":"I",  "c":"STORAGE",  "id":22297,   "ctx":"initandlisten","msg":"Using the XFS filesystem is strongly recomme
rage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem","tags":["startupWarnings"]}}
```

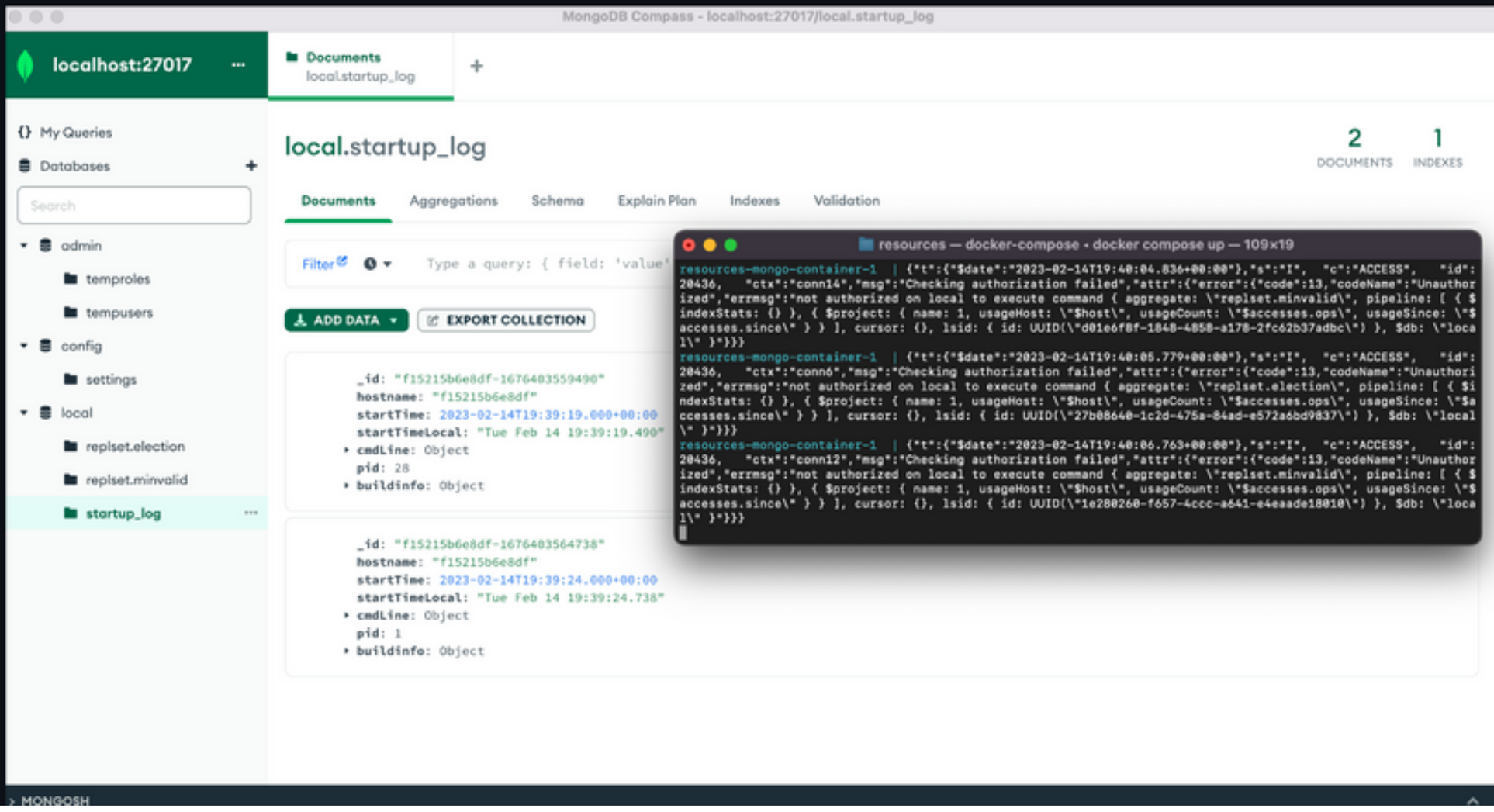
```
docker rm $(docker ps -a -q) -f
docker rmi $(docker images -a -q) -f

docker-compose build
docker-compose up

mongodb://test:test@123@localhost:27017/?authSource=admin&readPreference=primary&appname=MongoDB%20Compass&ssl=false
```

```
kubectl apply -f mongo-config.yml
configmap/mongo-conf created

kubectl apply -f mongo-secret.yml
secret/mongo-secret created

kubectl apply -f mongo-deployment.yml
service/mongodb-service created
persistentvolumeclaim/mongo-pv-claim created
deployment.apps/mongo created

kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
mongo-849c7745db-jj9ps   1/1     Running   0          41s
```

```
kubectl apply -f deployment.yml
service/spring-mongo-service created
deployment.apps/spring-mongo-service created

kubectl get pods
NAME                                  READY   STATUS              RESTARTS   AGE
mongo-849c7745db-47qm8                1/1     Running             0          5m33s
spring-mongo-service-6c9f688b8f-dqsfl   0/1     ContainerCreating   0          3s
spring-mongo-service-6c9f688b8f-kmfsf   0/1     ContainerCreating   0          3s

kubectl logs -f mongo-849c7745db-47qm8
kubectl logs -f spring-mongo-service-844c79fd84-bknxs

minikube service springboot-k8s

POST /addProduct

{
  "id" : "1",
  "productId" : "1234",
  "description" : "sai",
  "price" : "1234"
}

db.product.find().pretty();
```




Images
Volumes
Dev Environments BETA
Extensions BETA
Add Extensions
Images
Give feedback
An image is a read-only template with instructions for creating a Docker container. Learn more
LOCAL
REMOTE REPOSITORIES
Search
NAME
TAG
STATUS
CREATED
SIZE
ACTIONS
mongo
a440572ac3c1
latest
Unused
6 days ago
639.31 MB


MongoDB Compass - localhost:27017/local.startup_log
localhost:27017
Documents
local.startup_log
My Queries
Databases
Search
admin
temproles
tempusers
config
settings
local
replset.election
replset.minvalid
startup_log
local.startup_log
2
DOCUMENTS
1
INDEXES
Documents
Aggregations
Schema
Explain Plan
Indexes
Validation
Filter
Type a query: { field: 'value'
ADD DATA
EXPORT COLLECTION
_id: "f15215b6e8df-1676403559490"
hostname: "f15215b6e8df"
startTime: 2023-02-14T19:39:19.000+00:00
startTimeLocal: "Tue Feb 14 19:39:19.490"
cmdLine: Object
pid: 28
buildinfo: Object
_id: "f15215b6e8df-1676403564738"
hostname: "f15215b6e8df"
startTime: 2023-02-14T19:39:24.000+00:00
startTimeLocal: "Tue Feb 14 19:39:24.738"
cmdLine: Object
pid: 1
buildinfo: Object
resources — docker-compose • docker compose up — 109×19
resources-mongo-container-1 | {"t":{"$date":"2023-02-14T19:40:04.836+00:00"},"s":"I", "c":"ACCESS", "id":20436, "ctx":"conn14","msg":"Checking authorization failed","attr":{"error":{"code":13,"codeName":"Unauthorized","errmsg":"not authorized on local to execute command { aggregate: \"replset.minvalid\", pipeline: [{ $indexStats: {} }, { $project: { name: 1, usageHost: \"$host\", usageCount: \"$accesses.ops\", usageSince: \"$accesses.since\" } }], cursor: {}, lsid: { id: UUID(\"d01e6f8f-184b-4858-a178-2fc62b37adbc\") }, $db: \"local\" }"}}}
resources-mongo-container-1 | {"t":{"$date":"2023-02-14T19:40:05.779+00:00"},"s":"I", "c":"ACCESS", "id":20436, "ctx":"conn6","msg":"Checking authorization failed","attr":{"error":{"code":13,"codeName":"Unauthorized","errmsg":"not authorized on local to execute command { aggregate: \"replset.election\", pipeline: [{ $indexStats: {} }, { $project: { name: 1, usageHost: \"$host\", usageCount: \"$accesses.ops\", usageSince: \"$accesses.since\" } }], cursor: {}, lsid: { id: UUID(\"27b08640-1c2d-475a-84ad-e572a6bd9837\") }, $db: \"local\" }"}}}
resources-mongo-container-1 | {"t":{"$date":"2023-02-14T19:40:06.763+00:00"},"s":"I", "c":"ACCESS", "id":20436, "ctx":"conn12","msg":"Checking authorization failed","attr":{"error":{"code":13,"codeName":"Unauthorized","errmsg":"not authorized on local to execute command { aggregate: \"replset.minvalid\", pipeline: [{ $indexStats: {} }, { $project: { name: 1, usageHost: \"$host\", usageCount: \"$accesses.ops\", usageSince: \"$accesses.since\" } }], cursor: {}, lsid: { id: UUID(\"1e280260-f657-4ccc-a641-e4eaade18010\") }, $db: \"local\" }"}}}
MONGOSH