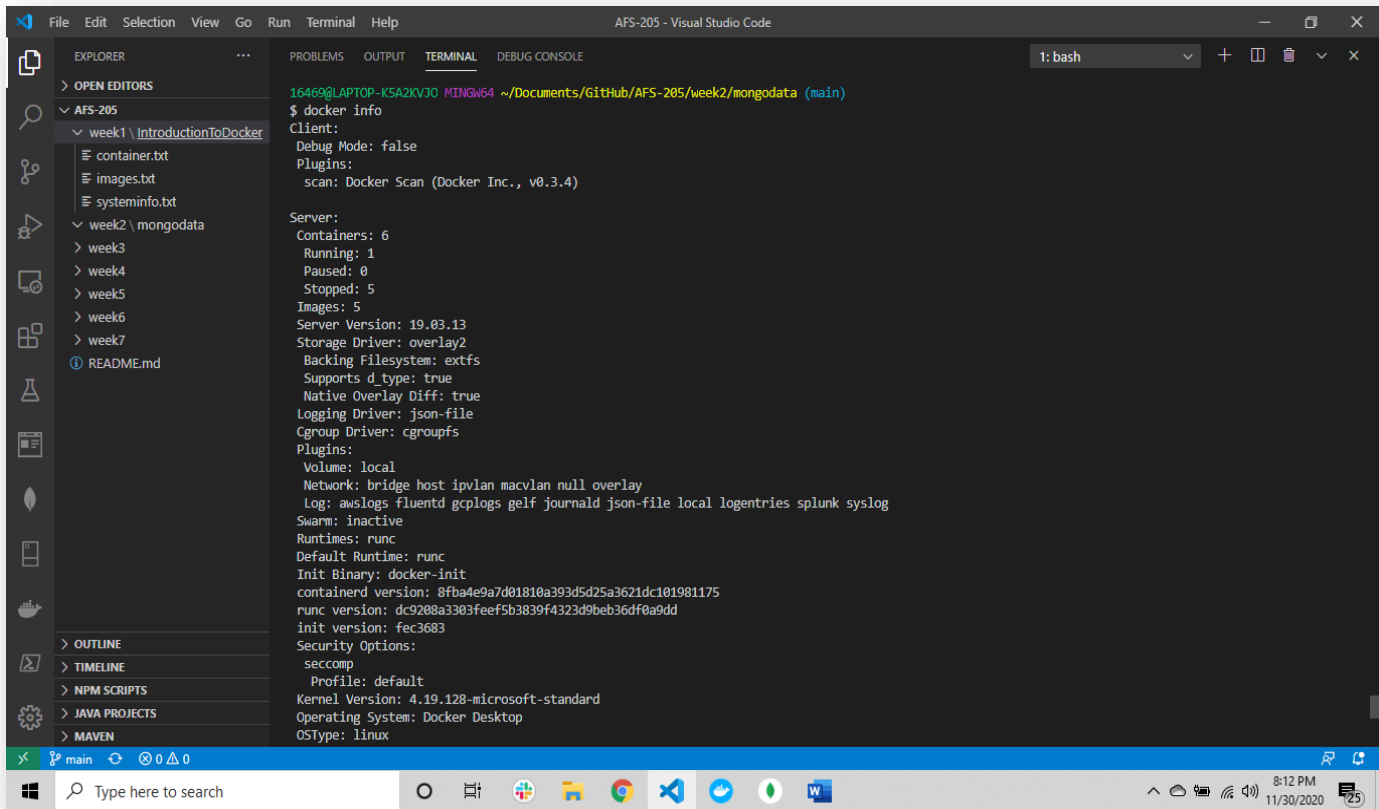


## Week 2 Project: Docker Containers

### Step 1: Download MongoDB Image for Docker

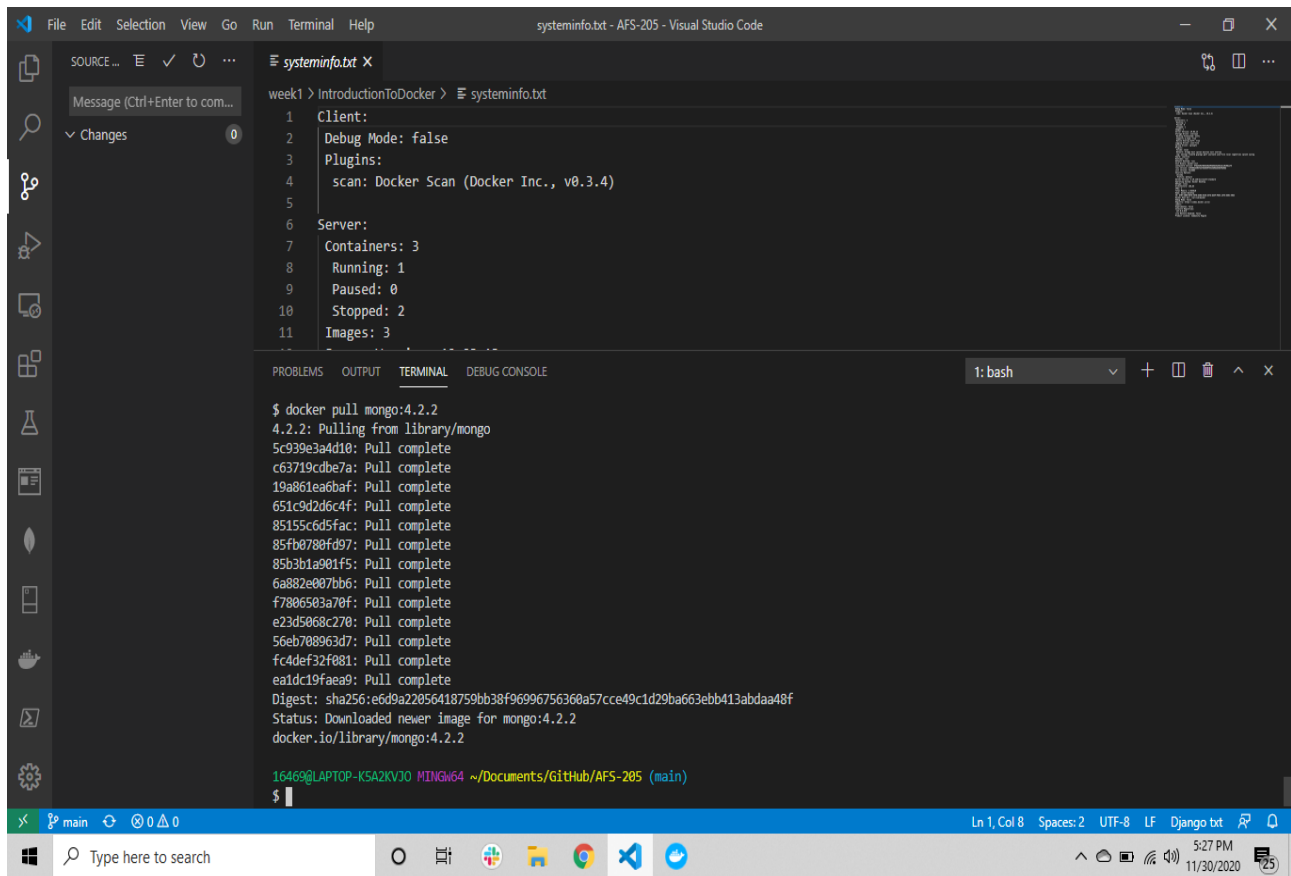
1. Your Docker service needs to be active and running. You can quickly check the current status by entering the following command in your terminal:

A screenshot of the Visual Studio Code interface. The Explorer sidebar on the left shows a project structure with folders 'week1' and 'week2', and files 'container.txt', 'images.txt', 'systeminfo.txt', and 'README.md'. The Terminal panel on the right is active, showing a bash prompt and the output of the 'docker info' command. The output provides detailed information about the Docker client and server, including version, storage driver, and network settings. The Windows taskbar is visible at the bottom with various application icons and a search bar.

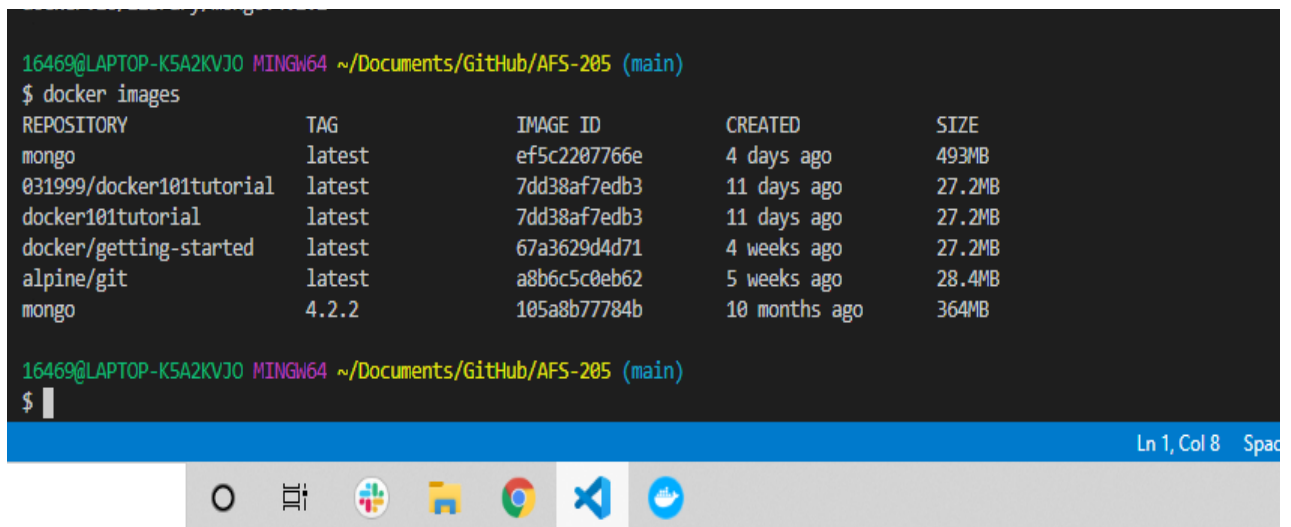
```
16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
$ docker info
Client:
 Debug Mode: false
 Plugins:
  scan: Docker Scan (Docker Inc., v0.3.4)

Server:
 Containers: 6
  Running: 1
  Paused: 0
  Stopped: 5
 Images: 5
 Server Version: 19.03.13
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
 Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: runc
 Default Runtime: runc
 Init Binary: docker-init
 containerd version: 8fba4e9a7d01810a393d5d25a3621dc101981175
 runc version: dc9208a3303feef5b3839f4323d9beb36df0a9dd
 init version: fec3683
 Security Options:
  seccomp
   Profile: default
 Kernel Version: 4.19.128-microsoft-standard
 Operating System: Docker Desktop
 OSType: linux
```

2. Proceed to download the latest official Docker image for the MongoDB database:



3. List the images in your Docker repository with the following command:



4. a- Create a **/mongodata** directory on the host system:  
In Windows OS:

```
16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205 (main)
$ mkdir mongodata
```

5. Start the Docker container with the **run** command using the **mongo** image. The **/data/db** directory in the container is mounted as **/mongodata** on the host. Additionally, this command changes the name of the container to **mongodb**:

```
16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205 (main)
$ docker run -dit -v mongodata:/data/db --name mongodb -d mongo
38129d44f662f0401880bc81b7df0b723aea02d7e7580e151b8c7b26a6c56bf1

16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205 (main)
$ docker run ps
```

6. Once the MongoDB server starts running in a container

```
MongoDB Documentation
Suggest a Feature
Report a Bug
Create Free Atlas Cluster

16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
$ docker ps
CONTAINER ID    IMAGE    COMMAND                  CREATED          STATUS          PORTS                               NAMES
1117488d18b3    mongo    "docker-entrypoint.s..." 23 minutes ago  Up 38 seconds  0.0.0.0:27017->27017/tcp            mongodb

16469@LAPTOP-K5A2KVJO MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
$
```

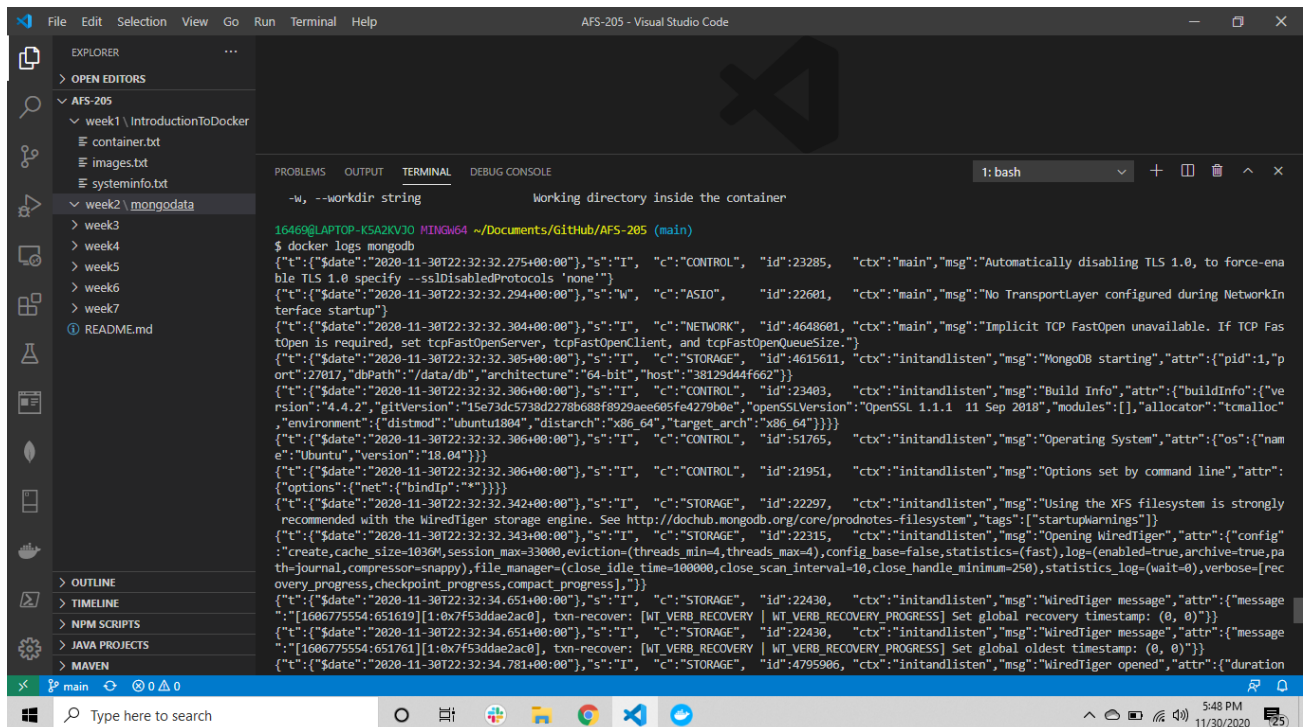
7. Optionally you can specify the MongoDB port explicitly:  
The default port number is **27017**, as can be seen in the output.

```
bash: !: command not found

16469@LAPTOP-K5A2KVJ0 MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
$ docker run -it -v mongodata:/data/db -p 27017:27017 --name mongodb -d mongo
1117488d18b3d4f8ea374acea31303585ba7e665d081bdfb16d0ce138e65cd53

16469@LAPTOP-K5A2KVJ0 MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
```

8. Always check the Docker log to see the chain of events after making changes.



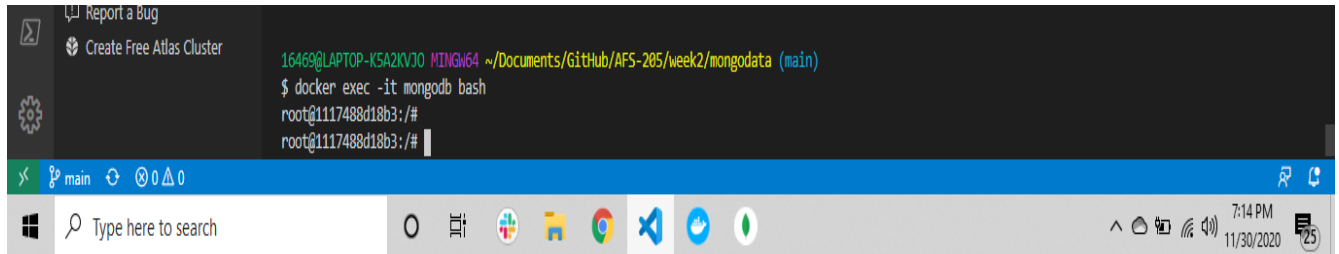
```
File Edit Selection View Go Run Terminal Help AFS-205 - Visual Studio Code

EXPLORER
> OPEN EDITORS
AFS-205
  week1 \ IntroductionToDocker
    container.txt
    images.txt
    systeminfo.txt
  week2 \ mongodata
  week3
  week4
  week5
  week6
  week7
  README.md
  OUTLINE
  TIMELINE
  NPM SCRIPTS
  JAVA PROJECTS
  MAVEN

TERMINAL
1: bash
-w, --workdir string Working directory inside the container

16469@LAPTOP-K5A2KVJ0 MINGW64 ~/Documents/GitHub/AFS-205 (main)
$ docker logs mongodb
{"t":{"$date":"2020-11-30T22:32:32.275+00:00"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"main","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'" }
{"t":{"$date":"2020-11-30T22:32:32.294+00:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup" }
{"t":{"$date":"2020-11-30T22:32:32.384+00:00"},"s":"I", "c":"NETWORK", "id":4648601, "ctx":"main","msg":"Implicit TCP FastOpen unavailable. If TCP FastOpen is required, set tcpFastOpenServer, tcpFastOpenClient, and tcpFastOpenQueueSize." }
{"t":{"$date":"2020-11-30T22:32:32.385+00:00"},"s":"I", "c":"STORAGE", "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting", "attr":{"pid":1, "port":27017, "dbPath":"/data/db", "architecture":"64-bit", "host":"38129d44f662"}}
{"t":{"$date":"2020-11-30T22:32:32.386+00:00"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"initandlisten","msg":"Build Info", "attr":{"buildInfo":{"version":"4.4.2", "gitVersion":"15e73dc5738d2278b688f8929aee605fe4279b0e", "openSSLVersion":"OpenSSL 1.1.1 11 Sep 2018", "modules":[], "allocator":"tcmalloc", "environment":{"distmod":"ubuntu1804", "distarch":"x86_64", "target_arch":"x86_64"}}}}
{"t":{"$date":"2020-11-30T22:32:32.386+00:00"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"initandlisten","msg":"Operating System", "attr":{"os":{"name":"Ubuntu", "version":"18.04"}}}
{"t":{"$date":"2020-11-30T22:32:32.386+00:00"},"s":"I", "c":"CONTROL", "id":21951, "ctx":"initandlisten","msg":"Options set by command line", "attr":{"options":{"net":{"bindIp":"*"}}}}
{"t":{"$date":"2020-11-30T22:32:32.342+00:00"},"s":"I", "c":"STORAGE", "id":22297, "ctx":"initandlisten","msg":"Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem", "tags":["startupWarnings"]}
{"t":{"$date":"2020-11-30T22:32:32.343+00:00"},"s":"I", "c":"STORAGE", "id":22315, "ctx":"initandlisten","msg":"Opening WiredTiger", "attr":{"config":{"create,cache_size=10369,session_max=33000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file_manager=(close_idle_time=100000,close_scan_interval=10,close_handle_minimum=250),statistics_log=(wait=0),verbose=[recovery_progress,checkpoint_progress,compact_progress]}}}
{"t":{"$date":"2020-11-30T22:32:34.651+00:00"},"s":"I", "c":"STORAGE", "id":22430, "ctx":"initandlisten","msg":"WiredTiger message", "attr":{"message":{"[1606775554:651619][1:0x7f53ddae2ac0], txn-recover: [WT_VERB_RECOVERY | WT_VERB_RECOVERY_PROGRESS] Set global recovery timestamp: (0, 0)}}}}
{"t":{"$date":"2020-11-30T22:32:34.651+00:00"},"s":"I", "c":"STORAGE", "id":22430, "ctx":"initandlisten","msg":"WiredTiger message", "attr":{"message":{"[1606775554:651761][1:0x7f53ddae2ac0], txn-recover: [WT_VERB_RECOVERY | WT_VERB_RECOVERY_PROGRESS] Set global oldest timestamp: (0, 0)}}}}
{"t":{"$date":"2020-11-30T22:32:34.781+00:00"},"s":"I", "c":"STORAGE", "id":4795906, "ctx":"initandlisten","msg":"WiredTiger opened", "attr":{"duration":0}}
```

9. The container is currently running in a **detached mode**. Connect to the container using the interactive terminal instead:



```
16469@LAPTOP-K5A2KV30 MINGW64 ~/Documents/GitHub/AFS-205/week2/mongodata (main)
$ docker exec -it mongodb bash
root@1117488d18b3:/#
root@1117488d18b3:/#
```

## 10. Default Database

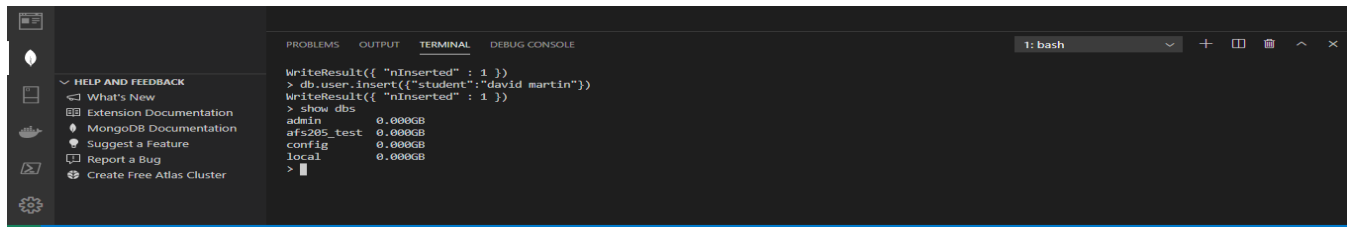
Show dbs

## 11. Create Database



```
WriteResult({ "nInserted" : 1 })
> db.user.insert({"student":"david martin"})
WriteResult({ "nInserted" : 1 })
>
```

## 12. Insert Database



```
WriteResult({ "nInserted" : 1 })
> db.user.insert({"student":"david martin"})
WriteResult({ "nInserted" : 1 })
> show dbs
admin            0.000GB
afs205_test      0.000GB
config           0.000GB
local            0.000GB
>
```

