

Section 4 - Command line structure - System Verification

1 - First Docker commands

Verify the Docker Install (1)

- Use the `docker version` command to display all version information
- This is the first command to execute on a new system to verify the Docker installation

Verify the Docker Install (2)

```
# docker version
```

```
Client:
```

```
Version:      18.03.0-ce  
API version:  1.37  
Go version:   go1.9.4  
Git commit:   0520e24302  
Built: Fri Mar 23 08:31:36 2018  
OS/Arch:     windows/amd64  
Experimental: false  
Orchestrator: swarm
```

```
Server:
```

```
Engine:
```

```
Version:      18.05.0-ce  
API version:  1.37 (minimum version 1.12)  
Go version:   go1.10.1  
Git commit:   f150324  
Built: Wed May 9 22:20:42 2018  
OS/Arch:     linux/amd64
```



Verify the Docker Install (3)

Notes:

- From the output of the `docker version` command we can see the client and the server version.
- Remember that Docker is a client server application.
- Ideally client and server versions should be the same but they don't have to be.
- We refer to Docker server also as Docker engine or Docker daemon.
- The fact that I did get returned information from the server validates that I can talk to the server and that it's working properly.

docker info

- Use the `docker info` command to display system-wide information

docker info (2)

```
# docker info
Containers: 5
  Running: 0
  Paused: 0
  Stopped: 5
Images: 24
Server Version: 18.05.0-ce
...
Swarm: inactive
... output truncated
```

Notes:

From the command output we can see:

- Number of containers (Running, Paused and Stopped)
- Number of images stored
- The Swarm state (active inactive)

Complete list of Docker commands (1)

- To get the complete list of Docker commands type `docker` and hit `enter`
- The Docker COMMAND has 3 main sections:
 1. Options
 2. Management Commands
 3. Commands

Complete list of Docker commands (2)

```
$ docker
```

```
Usage:  docker COMMAND
```

```
Options:
```

```
...
```

```
Management Commands:
```

```
  container    Manage containers
```

```
  image        Manage images
```

```
  network      Manage networks
```

```
...
```

```
Commands:
```

```
  attach        Attach local standard input, output, and error streams to a
```

```
  build          Build an image from a Dockerfile
```

```
...
```


Docker command format

- New format:

```
docker <mng-command> <command> [options]
```

- Old format (still working):

```
docker <command> [options]
```

Examples

new command format

```
# docker container run  
# docker container ps
```

old command format

```
# docker run  
# docker ps
```

Notes: Docker is really focused on backwards compatibility. So the docker run will probably work forever; but new commands we get will use this docker command value.

Commands Summary

```
# docker version
# docker info
# docker <Enter> => CLI documentation
# docker <mng-command> <command>
# docker container run
# docker run
```

Exercise

- Ref:
- `D_S4_L1_First_Docker_commands_ex.md`