

Assignment 4

Delyar Tabatabai

Jira and burndown:

The screenshot shows a Jira software interface for the project "CS 561 - Delyar Tabatabai". The board view displays three columns: TO DO, IN PROGRESS, and DONE 8 ISSUES. The DONE column contains two items, each with a screenshot of a terminal window. Item 3 shows a terminal with Docker commands, and item 1 shows a terminal with AWS training information. The Jira sidebar on the left includes sections for Planning, Development, and Reports, along with a "Board" section which is currently selected.

Projects / CS 561 - Delyar Tabatabai

DT Sprint 3 - Assignment 4

finish assignment 4!

TO DO IN PROGRESS DONE 8 ISSUES

3-Do all four combinations of the Docker client-server

ASSIGNMENT 4

DT-35

1-Do the online AWS training for this week; see the Syllabus

ASSIGNMENT 4

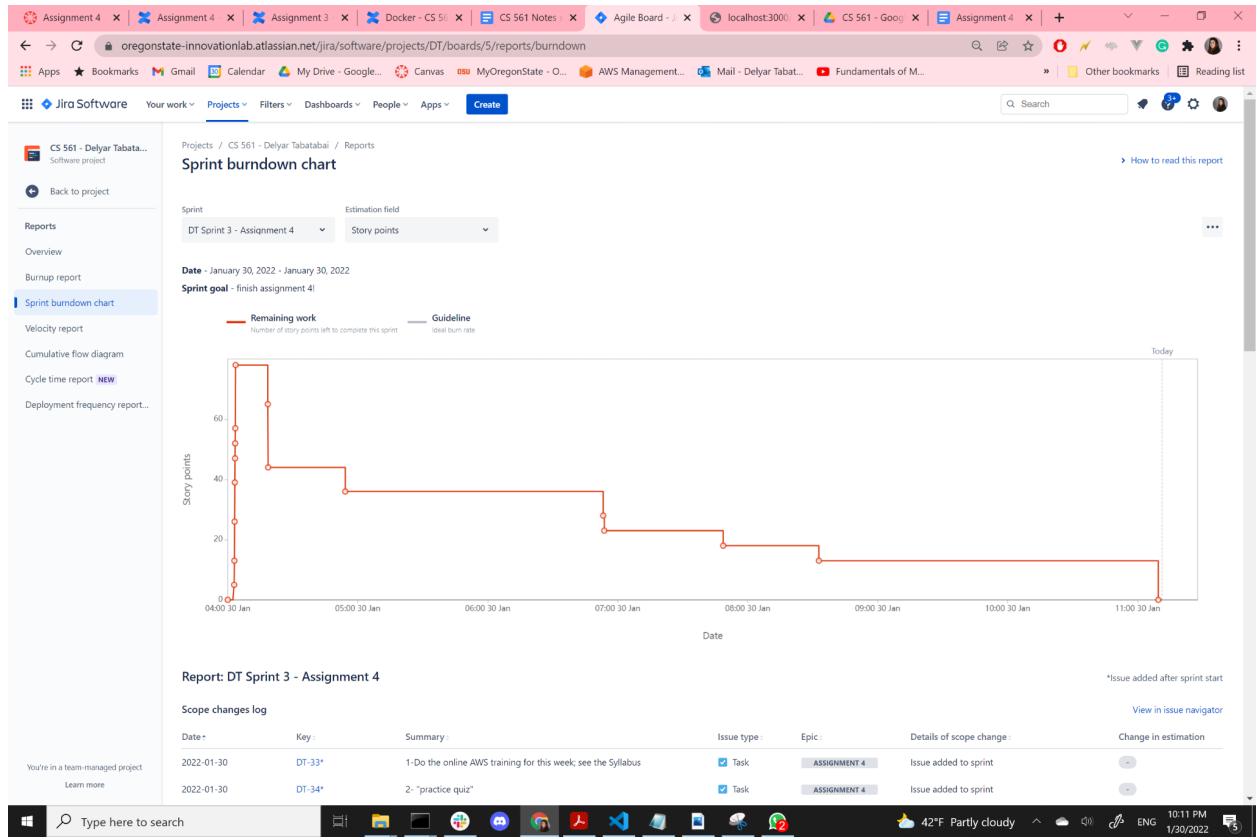
DT-33

7- five blog posts

You're in a team-managed project

Learn more

Quickstart



AWS Course:

DigitalCloud Training | Training | Pricing | Free Resources | Knowledge Hub | About | Challenge Labs | Delyar | Take Notes

Section Content: 100% Complete | 15/15 Steps

- Section 10—Introduction
- Decker-Containers and Microservices
- Amazon Elastic Container Service (ECS)
- Amazon ECS Launch Types
- [HOL] Launch Task on AWS Fargate
- Amazon ECS and IAM Roles
- Scaling Amazon ECS
- Amazon ECS with ALB
- [HOL] Deploy ECS Cluster (EC2 Launch Type)
- [HOL] Use ALB with Fargate Cluster
- Amazon Elastic Kubernetes Service (EKS)
- Architecture Patterns – EKS
- Exam Cram – EKS

Completed Sections:

- Section 8: DNS, Caching... 20 Lessons
- Section 9: Block and File... 22 Lessons
- Section 10: Docker Cont... 15 Lessons
- Section 10—Introduction
- Decker-Containers and Microservi...
- Amazon Elastic Container Service...
- Amazon ECR Launch Types
- [HOL] Launch Task on AWS Fargate
- Amazon ECS and IAM Roles
- Scaling Amazon ECS
- Amazon ECS with ALB
- [HOL] Deploy ECS Cluster (EC2 L...
- [HOL] Use ALB with Fargate Glus...
- Amazon Elastic Kubernetes Servi...
- Architecture Patterns—EKS
- Exam Cram—EKS
- Quiz—Amazon EKS
- Cheat Sheets—Amazon EKS

Join our Newsletter and get a 10% discount for your next purchase

Server and client in the same docker container:

The screenshot shows a Windows desktop environment with a terminal window open in the background and a video player window in the foreground.

The terminal window (Select Command Prompt - docker-compose run --rm dev) displays the following output:

```
at Object.Module._extensions..js (node:internal/modules/cjs/loader:1149:10)
at Module.load (node:internal/modules/cjs/loader:975:32)
at Function.Module._load (node:internal/modules/cjs/loader:822:12)
at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:77:12) {
  code: 'MODULE_NOT_FOUND',
  requireStack: [ '/code/app.js' ]
}

Node.js v17.4.0
root@70d040af8706:/code# npm install express.js
npm WARN old lockfile
npm WARN old lockfile The package-lock.json file was created with an old version of npm.
npm WARN old lockfile so supplemental metadata must be fetched from the registry.npm WARN old lockfile
npm WARN old lockfile This is a one-time fix-up, please be patient...
npm WARN old lockfile

added 50 packages, removed 2 packages, and audited 52 packages in 4s

2 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

npm notice
npm notice New minor version of npm available! 8.3.1 -> 8.4.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v8.4.0
npm notice Run npm install -g npm@8.4.0 to update!
npm notice

root@70d040af8706:/code# node app.js
Node.js Express server is running on port 3000..
```

The video player window (Select Command Prompt - docker-compose run --rm dev) shows a black screen with a progress bar at the bottom. It has buttons for "Invite", "Mute Me", and "Select Encoding".

The taskbar at the bottom shows several pinned icons and a search bar labeled "Type here to search". The system tray indicates it's 45°F, 8:50 PM, and 1/30/2022.

Server in docker, client in another docker container:

The screenshot shows a Windows desktop environment with two command prompt windows open and a taskbar at the bottom.

Left Command Prompt (Rooted):

```
root@5abc848b0284:/code
>Selecting previously unselected package nodejs-doc.
>Preparing to unpack .../nodejs-doc_10.19.0~dfsg-3ubuntu1_all.deb ...
>Unpacking nodejs-doc (10.19.0~dfsg-3ubuntu1) ...
>Selecting previously unselected package nodejs.
>Preparing to unpack .../nodejs_10.19.0~dfsg-3ubuntu1_amd64.deb ...
>Unpacking nodejs (10.19.0~dfsg-3ubuntu1) ...
>Setting up libc-ares2:amd64 (1.15.0-1ubuntu0.1) ...
>Setting up libuv1:amd64 (1.34.2-1ubuntu1.3) ...
>Setting up libnode64:amd64 (10.19.0~dfsg-3ubuntu1) ...
>Setting up nodejs-doc (10.19.0~dfsg-3ubuntu1) ...
>Setting up nodejs (10.19.0~dfsg-3ubuntu1) ...
>update-alternatives: using /usr/bin/nodejs to provide /usr/bin/js (js) in auto mode
>update-alternatives: warning: skip creation of /usr/share/man/man1/js.1.gz because associated file /usr/share/man/man1/nodejs.1.gz (of link group js) doesn't exist
>Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
root@5abc848b0284:/code# node app.js
Node.js Express server is running on port 3000..
```

Right Command Prompt (User delyar):

```
C:\Users\delyar>docker ps
CONTAINER ID IMAGE COMMAND
EATED STATUS PORTS
NAMES
5abc848b0284 swiftlang/swift:nightly-focal "bash" 25
seconds ago Up 23 seconds
myserver
02611690013f1 rustlang/rust:nightly "bash -c 'curl -o h..'" 27
minutes ago Up 27 minutes 0.0.0.0:3000->3000/tcp, ::3000->3000/tcp
lab_dev_run_c5d291dc0d5f

C:\Users\delyar>docker inspect myserver | grep IPAddress
'grep' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\delyar>docker inspect --format '{{ .NetworkSettings.IPAddress }}' myserver
Template parsing error: template: :1: unclosed action

C:\Users\delyar>docker inspect -f '{{range.NetworkSettings.Networks}}{{.IPAddress}}{{end}}' myserver
'172.18.0.3'

C:\Users\delyar>docker-compose run --rm --name myclient client
ERROR:
      Can't find a suitable configuration file in this directory or any parent. Are you in the right directory?
      Supported filenames: docker-compose.yml, docker-compose.yaml, compose.yml, compose.yaml

C:\Users\delyar>docker inspect -f '{{range.NetworkSettings.Networks}}{{.IPAddress}}{{end}}' myclient
'172.18.0.4'

C:\Users\delyar>
```

Taskbar:

- Type here to search
- File Explorer icon
- Start button icon
- Icons for File Explorer, File History, Task View, Taskbar settings, and a red notification badge.
- System tray icons: battery (42°F Partly cloudy), signal strength, volume, and date/time (10:07 PM, 1/30/2022).

Server in docker, client on host:

```
c:\ Command Prompt - docker-compose run --rm --service-ports dev
root@0261169013f1:/# npm install express
npm ERR! Tracker "idealTree" already exists

npm ERR! A complete log of this run can be found in:
npm ERR!   /root/.npm/_logs/2022-01-31T05_18_25_180Z-debug-0.log
root@0261169013f1:/# ls
bin  code  etc  lib  media  opt  root  sbin  sys  usr
boot  dev  home  lib64  mnt  proc  run  srv  tmp  var
root@0261169013f1:/# cd code
root@0261169013f1:/code# ls
app.js  cs561-swift  docker-compose.yml  node_modules  package.json
cs561-rust  delayarlib  mock-weather-server  package-lock.json
root@0261169013f1:/code# apt-get install npm
Reading package lists...
Building dependency tree
Reading state information... Done
npm is already the newest version (5.8.0+ds6-4+deb10u2).
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
root@0261169013f1:/code# ls
app.js  cs561-swift  docker-compose.yml  node_modules  package.json
cs561-rust  delayarlib  mock-weather-server  package-lock.json
root@0261169013f1:/code# apt-get install npm
Reading package lists...
Building dependency tree
Reading state information... Done
npm is already the newest version (5.8.0+ds6-4+deb10u2).
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
root@0261169013f1:/code# npm install express
up to date, audited 52 packages in 1s

2 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New minor version of npm available! 8.3.1 => 8.4.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v8.4.0
npm notice Run npm install -g npm@8.4.0 to update!
npm notice
root@0261169013f1:/code# node app.js
Node.js Express server is running on port 3000..
```

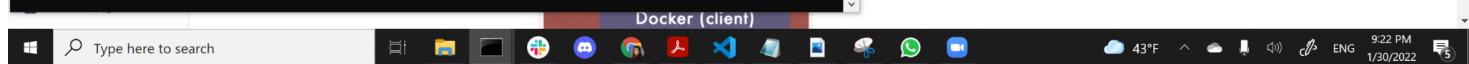
```
c:\ Command Prompt
Microsoft Windows [Version 10.0.19042.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\delyar>curl "localhost:3000/data/2.5/weather"
curl: (52) Empty reply from server

C:\Users\delyar>curl "localhost:3000/data/2.5/weather"
curl: (52) Empty reply from server

C:\Users\delyar>curl http://localhost:3000/data/2.5/weather
curl: (52) Empty reply from server

C:\Users\delyar>curl http://localhost:3000/data/2.5/weather
{"coord":{"lon":-123.262,"lat":44.5646}, "weather":[{"id":800,"main":"Clear","description":"clear sky","icon":"01n"}], "base":"stations", "main":{"temp":282.61,"feels_like":282.61,"temp_min":280.58,"temp_max":285.29,"pressure":1018,"humidity":84}, "visibility":10000, "wind":{"speed":0.89,"deg":225,"gust":0.89}, "clouds":{"all":0}, "dt":1642038331, "sys":{"type":2,"id":2040223,"country":"US","sunrise":1642002454,"sunset":1642035291}, "timezone":-28800, "id":5720727, "name":"Corvallis", "cod":200}
C:\Users\delyar>
```



Server on host, client in docker:

```
root@47190f610f3e:/code
root@47190f610f3e:/code# cd ..
root@47190f610f3e:/code# ls
app.js  Dockerfile  docker-compose.yml  node_modules  package.json
css561-rust  delyarlib  dock-weather-server  package-lock.json
root@47190f610f3e:/code# apt-get update
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [889 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal/main amd64 Packages [1275 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [1470 kB]
Get:8 http://archive.ubuntu.com/ubuntu focal/universe amd64 Packages [11.3 MB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [839 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 Packages [30.1 kB]
Get:11 http://archive.ubuntu.com/ubuntu focal/restricted amd64 Packages [33.4 kB]
Get:12 http://archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [177 kB]
Get:13 http://archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [952 kB]
Get:14 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [1121 kB]
Get:15 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1895 kB]
Get:16 http://archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 Packages [33.7 kB]
Get:17 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 Packages [23.8 kB]
Get:18 http://archive.ubuntu.com/ubuntu focal-backports/main amd64 Packages [50.8 kB]
Fetched 20.7 MB in 55s (375 kB/s)
Reading package lists... Done
root@47190f610f3e:/code# apt-get install curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
curl
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 161 kB of archives.
After this operation, 412 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.7 [161 kB]
Fetched 161 kB in 11s (14.7 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package curl.
(Reading database ... 16673 files and directories currently installed.)
Preparing to unpack .../curl_7.68.0-1ubuntu2.7_amd64.deb ...
Unpacking curl (7.68.0-1ubuntu2.7) ...
Setting up curl (7.68.0-1ubuntu2.7) ...
root@47190f610f3e:/code# curl host.docker.internal:3000/data/2.5/weather
{"coord":{"lon": -123.262,"lat": 44.5646}, "weather": [{"id": 800, "main": "Clear", "description": "clear sky", "icon": "01n"}], "base": "stations", "main": {"temp": 282.61, "feels_like": 282.61, "temp_min": 280.58, "temp_max": 285.29, "pressure": 1018, "humidity": 84}, "visibility": 10000, "wind": {"speed": 0.89, "deg": 225, "gust": 0.89}, "clouds": {"all": 0}, "dt": 1642038}
root@47190f610f3e:/code#
```



Bullet 4:

The screenshot shows a browser window with the Swagger Editor interface. On the left, there's a code editor displaying a JSON API definition. On the right, there's a detailed view of an endpoint. The endpoint '/weather' is listed under 'Responses'. The endpoint '/hello' is selected, showing its parameters, responses, and a successful execution example.

API Definition (Left Panel):

```
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Selected Endpoint: /hello

GET /weather Current Temperature

GET /hello return a random greeting

Parameters

No parameters

Execute Clear

Responses Response content type application/json

Curl

```
curl -X 'GET' \
'http://localhost:3000/v1/hello' \
-H 'accept: application/json'
```

Request URL

```
http://localhost:3000/v1/hello
```

Server response

Code Details

200 Response body

```
Hi!
```

Download

Response headers

```
content-length: 3
content-type: text/html; charset=utf-8
```

Responses

Bullet 5:

The screenshot shows the Hopscotch interface for a REST API. The URL is `http://localhost:3000/v1/weather`. The response status is `200 • OK`, time taken is `14 ms`, and size is `488 B`. The response body is a JSON object containing weather data for a specific location. The JSON output is as follows:

```
1 + {  
2 +   "coord": {  
3 +     "lon": -123.262,  
4 +     "lat": 44.5646  
5 +   },  
6 +   "weather": [  
7 +     {  
8 +       "id": 800,  
9 +       "main": "Clear",  
10 +      "description": "clear sky",  
11 +      "icon": "01n"  
12 +    }  
13 +  ],  
14 +  "base": "stations",  
15 +  "main": {  
16 +    "temp": 282.61,  
17 +    "feels_like": 282.61,  
18 +    "temp_min": 280.58,  
19 +    "temp_max": 285.29,  
20 +    "pressure": 1018,  
21 +    "humidity": 84  
22 +  },  
23 +  "visibility": 10000,  
24 +  "wind": {  
25 +    "speed": 0.89,  
26 +    "deg": 225,  
27 +    "gust": 0.89  
28 +  },  
29 +  "clouds": {  
30 +    "all": 0  
31 +  },  
32 +}  
33 +
```

The screenshot shows the Hopscotch interface for a REST API. The URL is `http://localhost:3000/v1/hello`. The response status is `200 • OK`, time taken is `30 ms`, and size is `3 B`. The response body is a simple string: `Hi!`

HOPPSOTCH Star 36,070

REST

GraphQL

Realtime

Docs

Settings

POST http://localhost:3000/v1/auth

Send Save

Parameters 2 Body Headers Authorization Pre-request Script Tests

Query Parameters

username Delyar

password Tabatabai

Parameter 3 Value 3

Status: 200 • OK Time: 32 ms Size: 34 B

JSON Raw Headers Test Results

Response Body

```
1  {
2    "Success": "this is your token..."
3 }
```

Search Just now

POST http://localhost:3000/v1/auth 33 seconds ago

POST http://localhost:3000/v1/auth 1 minute ago

GET http://localhost:3000/v1/hello 3 minutes ago

GET http://localhost:3000/v1/weather 3 weeks ago

POST https://api.openweathermap.org/data/2.5... 1 month ago

POST api.openweathermap.org/data/2.5/weath... 1 month ago

GET api.openweathermap.org/data/2.5/weath... 1 month ago



File Edit Selection View Go Run Terminal Help

app.js - Visual Studio Code

swagger.yaml app.js C:\...\mock-weather-server JS app.js C:\...\hw4

C: > Users > delyar > Documents > Lab > mock-weather-server > JS app.js > get_weather > "weather" > "icon"

32
33
34
35
36 app.get('/data/2.5/weather',get_weather)
37 function get_weather(request, response) {
38 | response.json({ "coord":{ "lon":-123.262,"lat":44.5646}, "weather": [{ "id":800, "main": "Clear", "description": "clear sky", "icon": "01n" }], "t
39 }
40
41
42 app.get("/v1/weather", get_weather2)
43 function get_weather2(request, response) {
44 | console.log('get_weather2 received req for v1 weather')
45 | response.json({ "coord":{ "lon":-123.262,"lat":44.5646}, "weather": [{ "id":800, "main": "Clear", "description": "clear sky", "icon": "01n" }], "t
46 }
47
48 app.get("/v1/hello", get_hello)
49 function get_hello(request, response) {
50 | response.send('Hi!')
51 }
52
53 app.post('/v1/auth',post_auth)
54 function post_auth(req,res){
55 | var username = req.body.username
56 | var pwd = req.body.password
57
58 | console.log(username,pwd)
59 | res.json({ "Sucess" : "this is your token..." })
60
61 }
62 }

bodyParser Aa ab * 2 of 2 ↑ ↓ ≡ ×

Command Prompt

```
C:\Users\delyar>curl http://localhost:3000/v1/weather
{"coord":{"lon":-123.262,"lat":44.5646}, "weather": [ {"id":800, "main": "Clear", "description": "clear sky", "icon": "01n"} ], "base": "stations", "main": {"temp": 282.61, "feels_like": 282.61, "temp_min": 280.58, "temp_max": 285.29, "pressure": 1018, "humidity": 84}, "visibility": 10000, "wind": {"speed": 0.89, "deg": 225, "gust": 0.89}, "clouds": {"all": 0}, "dt": 1642038331, "sys": {"type": 2, "id": 2040223, "country": "US", "sunrise": 1642002454, "sunset": 1642035291}, "timezone": -28800, "id": 5720727, "name": "Corvallis", "cod": 200}
C:\Users\delyar>curl http://localhost:3000/v1/hello
Hi!
C:\Users\delyar>curl -d "username=Delyar&password=123" -X POST http://localhost:3000/v1/auth
{"Sucess": "this is your token..."}
C:\Users\delyar>
```

Ln 38, Col 123 Spaces: 5 UTF-8 CRLF {} JavaScript ⚙ Rain to stop ⌂ ⌃ ⌄ ENG 6:41 PM 1/30/2022

Bullet 6:

```
root@f777d58777e3:/# ls
bin boot code dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@f777d58777e3:/# ls -a
. .dockerenv bin code dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp var
.. .swift_tag boot dev home lib32 libx32 mnt proc run srv tmp var
root@f777d58777e3:/# ls -l code
total 4
drwxr-xr-x 1 root root 4096 Jan 24 01:22 cs561-rust
drwxrwxrwx 1 root root 4096 Jan 24 02:40 CSSSI_swift
drwxr-xr-x 1 root root 4096 Jan 9 23:20 delyarlib
-rw-rw-rwx 1 root root 485 Jan 24 02:45 docker-compose.yml
drwxrwxrwx 1 root root 4096 Jan 24 01:45 mock-weather-server
drwxrwxrwx 1 root root 4096 Jan 31 03:24 node_modules
-rw-rw-rwx 1 root root 757 Jan 31 03:24 package-lock.json
root@f777d58777e3:/# .
bash: .: filename argument required
.. usage: . filename [arguments]
root@f777d58777e3:/# . code
bash: .. code: is a directory
root@f777d58777e3:/# . package-lock.json
bash: package-lock.json: No such file or directory
root@f777d58777e3:/# . docker-compose.yml
bash: docker-compose.yml: No such file or directory
root@f777d58777e3:/# pwd
/
root@f777d58777e3:/# pwd
/
root@f777d58777e3:/# ls -a
. .dockerenv bin code dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
.. .swift_tag boot dev home lib32 libx32 mnt proc run srv tmp var
root@f777d58777e3:/# ls
bin boot code dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@f777d58777e3:/# ps
  PID TTY          TIME CMD
    1 pts/0    00:00:00 bash
   20 pts/0    00:00:00 ps
root@f777d58777e3:/# history | grep docker
 7 . docker-compose.yml
 12 history | grep docker
root@f777d58777e3:/# history > temp.log
root@f777d58777e3:/# open .code
bash: open: command not found
root@f777d58777e3:/# whoami
root
root@f777d58777e3:/#
```

