

DOCKER

DOCKER 101

```
docker run -d -p 80:80 dockersamples/101-tutorial
```

- ➔ -d = detach mode
- ➔ -p = port number
- ➔ ImageName

- **docker run -rm -it ubuntu /bin/bash**
--rm : removing the container after its done
- **hostname -- (HostName of the Container)**
- **ps -- (Processes inside the container)**

Namespaces

- What you can see
- Created with syscalls
 - Unix Timesharing System
 - Process IDs
 - Mounts
 - Network
 - User IDs
 - InterProcess Comms
- **Namespaces: Namespaces is where the limit what the process can see (We are able to see the)**
- We setup these Namespaces using **SYSCALLS**
- **docker run <imageName> <CMD> <PARAMetrs>**
- building an app to prove out your MVP (minimum viable product).
-

- cd app/
- ls
- touch Dockerfile.txt
-
-
-

DOCKER

➤ **vi Dockerfile**

```
> FROM node:10-alpine
> WORKDIR /app
> COPY .
> RUN yarn install --production
> CMD ["node", "/app/src/index.js"]
```

➤ **Docker build -t docker-101 .**

➤ **NOW STRATING THE APP CONTAINER**

➤ **Docker run -dp 3000:3000 docker-101-**

➤ You can stop and remove a container in a single command by adding the "force" flag to the **docker rm** command. For example:

```
edff9ff691d5: Preparing
cbe4b9146f86: Preparing
a6524c5b12a6: Waiting
9a5d14f9f550: Waiting
denied: requested access to the resource is denied
[node1] (local) root@192.168.0.18 ~/app
$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username: kalyanjs
Password: [REDACTED]
```

```
Status: Downloaded newer image for dockersamples/101-tutorial:latest
f61225087e8fb39bd4d2933c39892373f9984d265d2d52c611eee6bd73f1baaa
[node1] (local) root@192.168.0.18 ~
$ docker run --rm -it ubuntu /bin/bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e49
6e5c1f05f
Status: Downloaded newer image for ubuntu:latest
root@25ba31d1366a:/# hostname
25ba31d1366a
root@25ba31d1366a:/# hostname container
hostname: you must be root to change the host name
root@25ba31d1366a:/# exit
exit
[node1] (local) root@192.168.0.18 ~
```

DOCKER

OPEN PORT

3000

Memory

4.32% (172.8MiB / 3.906GiB)

CPU

0.48%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v 

DELETE

 EDITOR

```
unzip app.zip
archive: app.zip
  creating: app/
  inflating: app/package.json
  inflating: app/yarn.lock
  creating: app/src/
  creating: app/src/routes/
  inflating: app/src/routes/updateItem.js
  inflating: app/src/routes/getItems.js
  inflating: app/src/routes/addItem.js
  inflating: app/src/routes/deleteItem.js
  inflating: app/src/.DS_Store
  creating: app/src/persistence/
  inflating: app/src/persistence/mysql.js
  inflating: app/src/persistence/sqlite.js
  inflating: app/src/persistence/index.js
  inflating: app/src/index.js
```

DOCKER

4.32% (1/2.8MiB / 3.906GiB)

CPU

0.19%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

DELETE

 EDITOR

```
inflating: app/src/static/css/styles.css
creating: app/spec/
creating: app/spec/routes/
inflating: app/spec/routes/addItem.spec.js
inflating: app/spec/routes/updateItem.spec.js
inflating: app/spec/routes/deleteItem.spec.js
inflating: app/spec/routes/getItems.spec.js
creating: app/spec/persistence/
inflating: app/spec/persistence/sqlite.spec.js
[node1] (local) root@192.168.0.18 ~
$ cd app/
[node1] (local) root@192.168.0.18 ~/app
$ ls
package.json  spec          src          yarn.lock
[node1] (local) root@192.168.0.18 ~/app
$ touch Dockerfile.txt
[node1] (local) root@192.168.0.18 ~/app
```

```
[node1] (local) root@192.168.0.18 ~/app
$ vi Dockerfile.txt
[node1] (local) root@192.168.0.18 ~/app
$ docker build -t docker-101 .
[+] Building 0.2s (2/2) FINISHED
          docker:default
=> [internal] load .dockerignore
               0.1s
=> => transferring context: 2B
               0.0s
=> [internal] load build definition from Dockerfile
               0.1s
=> => transferring dockerfile: 2B
               0.0s
ERROR: failed to solve: failed to read dockerfile: open /var/lib/docker/tmp/buildkit-mount3937489753/Dockerfile: no such file or directory
[node1] (local) root@192.168.0.18 ~/app
```

DOCKER

DELETE

EDITOR

```
$ docker build -t docker-101 .
[+] Building 0.1s (2/2) FINISHED
      docker:default
=> [internal] load build definition from Dockerfile
               0.0s
=> => transferring dockerfile: 2B
               0.0s
=> [internal] load .dockerignore
               0.0s
=> => transferring context: 2B
               0.0s
ERROR: failed to solve: failed to read dockerfile: open /var/lib/docker/tmp/buildkit-mount1497806998/Dockerfile: no such file or directory
[node1] (local) root@192.168.0.18 ~/app
$ touch Dockerfile
[node1] (local) root@192.168.0.18 ~/app
$ vi Dockerfile
```

DOCKER

OPEN PORT

3000

Memory
4.31% (172.3MiB / 3.906GiB)

CPU
0.28%

SSH
ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
$ vi Dockerfile
[node1] (local) root@192.168.0.18 ~/app
$ docker build -t docker-101 .
[+] Building 11.3s (9/9) FINISHED
      docker:default
=> [internal] load .dockerignore
              0.0s
=> => transferring context: 2B
              0.0s
=> [internal] load build definition from Dockerfile
              0.0s
=> => transferring dockerfile: 144B
              0.0s
=> [internal] load metadata for docker.io/library/node:10-alpine
              0.5s
=> [1/4] FROM docker.io/library/node:10-alpine@sha256:dc98dac24efd425
4f75976c40bce46944697a110d06ce7fa47e72 8.8s
```

DOCKER

OPEN PORT

3000

Memory
4.31% (172.3MiB / 3.906GiB)

CPU
0.21%

SSH
ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
2e05bb319d005f7945ff5d      0.0s
=> => naming to docker.io/library/docker-101
                                0.0s
[node1] (local) root@192.168.0.18 ~/app
$ docker run -dp 3000:3000 docker-101
bd46b18d1765f06d4843371d69f44c5aa2550c37731ef37044fd262657e8f127
[node1] (local) root@192.168.0.18 ~/app
$ docker run -dp 3000:3000 docker-101
c3702875ad6836b3cd11cd2a1b4f1e900878aefccae74d6111665f6f8fad9869
[node1] (local) root@192.168.0.18 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND            C
CREATED             STATUS              PORTS
f61225087e8f        dockersamples/101-tutorial   "nginx -g 'daemon off..."  2
9 minutes ago       Up 29 minutes     0.0.0.0:80->80/tcp    gallant_germain
[node1] (local) root@192.168.0.18 ~/app
$ vi Dockerfile
```

DOCKER

OPEN PORT

3000

Memory
4.31% (172.3MiB / 3.906GiB)

CPU
0.13%

SSH
ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
=> => transferring dockerfile: 145B 0.0s
=> [internal] load metadata for docker.io/library/node:10-alpine 0.2s
=> CACHED [1/4] FROM docker.io/library/node:10-alpine@sha256:d 0.0s
=> [2/4] WORKDIR /app 0.1s
=> [internal] load build context 0.0s
=> => transferring context: 2.71kB 0.0s
=> [3/4] COPY . . 0.1s
=> [4/4] RUN yarn install --production 25.1s
=> exporting to image 4.5s
=> => exporting layers 4.5s
=> => writing image sha256:c6a039b453e87ed104bcdd2aecbcf544c6d 0.0s
=> => naming to docker.io/library/docker-101 0.0s
[node1] (local) root@192.168.0.18 ~/app
$ docker rm Dockerfile.txt
Error response from daemon: No such container: Dockerfile.txt
[node1] (local) root@192.168.0.18 ~/app
$ rm Dockerfile.txt
```

DOCKER

3000

Memory

4.31% (172.3MiB / 3.906GiB)

CPU

0.24%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
rm Dockerfile.txt
[node1] (local) root@192.168.0.18 ~/app
ls
Dockerfile  package.json  spec      src      yarn.lock
[node1] (local) root@192.168.0.18 ~/app
cat Dockerfile

FROM node:10-alpine
WORKDIR /app
COPY . .
RUN yarn install --production
CMD ["node", "/app/src/index.js"]

[node1] (local) root@192.168.0.18 ~/app
docker ps
CONTAINER ID        IMAGE               COMMAND
CREATED             STATUS              PORTS
NAMES
```

DOCKER

OPEN PORT

3000

Memory

4.31% (172.3MiB / 3.906GiB)

CPU

0.52%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

DELETE

 EDITOR

```
WORKDIR /app
COPY . .
RUN yarn install --production
CMD ["node", "/app/src/index.js"]

[node1] (local) root@192.168.0.18 ~/app
$ docker ps
CONTAINER ID   IMAGE          COMMAND
CREATED        STATUS         PORTS
f61225087e8f   dockersamples/101-tutorial   "nginx -g 'daemon off..." A
bout an hour ago   Up About an hour   0.0.0.0:80->80/tcp gallant_germain
[node1] (local) root@192.168.0.18 ~/app
$ docker stop f6122508
f6122508
[node1] (local) root@192.168.0.18 ~/app
$ docker rm f6122508
```

DOCKER

OPEN PORT

3000

Memory

4.29% (171.8MiB / 3.906GiB)

CPU

0.25%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v 

DELETE

 EDITOR

```
$ docker rm f6122508
f6122508
[node1] (local) root@192.168.0.18 ~/app
$ docker ps
CONTAINER ID   IMAGE      COMMAND   CREATED     STATUS      PORTS     NAMES
[node1] (local) root@192.168.0.18 ~/app
$ docker build -t docker-101
ERROR: "docker buildx build" requires exactly 1 argument.
See 'docker buildx build --help'.

Usage:  docker buildx build [OPTIONS] PATH | URL | -
        Start a build
[node1] (local) root@192.168.0.18 ~/app
$ docker build -t docker-101 .
[+] Building 29.0s (9/9) FINISHED                               docker:default
=> [internal] load build definition from Dockerfile           0.0s
```

DOCKER

OPEN PORT

3000

Memory

4.29% (171.8MiB / 3.906GiB)

CPU

0.22%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

DELETE

 EDITOR

```
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 145B                         0.0s
=> [internal] load .dockerignore                            0.0s
=> => transferring context: 2B                           0.0s
=> [internal] load metadata for docker.io/library/node:10-alpi 0.2s
=> [1/4] FROM docker.io/library/node:10-alpine@sha256:dc98dac2 0.0s
=> [internal] load build context                          0.0s
=> => transferring context: 2.56kB                      0.0s
=> CACHED [2/4] WORKDIR /app                           0.0s
=> [3/4] COPY . .                                     0.1s
=> [4/4] RUN yarn install --production                24.8s
=> exporting to image                                  3.7s
=> => exporting layers                                3.7s
=> => writing image sha256:2ba92969266de4f3ec1cd3dc463aa7a855c 0.0s
=> => naming to docker.io/library/docker-101           0.0s
[node1] (local) root@192.168.0.18 ~/app
$ docker run -dp 3000:3000 docker-101
```

DOCKER

OPEN PORT

3000

Memory
4.29% (171.8MiB / 3.906GiB)

CPU
0.56%

SSH
ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
82abb1255ce64f7df752244ddb4485d49610dcef19b37c586ae784f9e6d7edeb
[node1] (local) root@192.168.0.18 ~/app
$ docker build -t docker-101 .
[+] Building 33.4s (9/9) FINISHED          docker:default
=> [internal] load .dockerignore           0.0s
=> => transferring context: 2B            0.0s
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 145B        0.0s
=> [internal] load metadata for docker.io/library/node:10-alpi 0.3s
=> [1/4] FROM docker.io/library/node:10-alpine@sha256:dc98dac2 0.0s
=> [internal] load build context           0.0s
=> => transferring context: 8.01kB        0.0s
=> CACHED [2/4] WORKDIR /app              0.0s
=> [3/4] COPY . .                         0.2s
=> [4/4] RUN yarn install --production    28.0s
=> exporting to image                     4.5s
=> => exporting layers                   4.5s
```

DOCKER

OPEN PORT

3000

Memory

4.29% (171.8MiB / 3.906GiB)

CPU

0.19%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
a78fb6fa075ac83193cdb6188f1c60dff7cd1): Bind for 0.0.0.0:3000 failed:  
port is already allocated.  
[node1] (local) root@192.168.0.18 ~/app  
$ docker ps  
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES  
82abb1255ce6      2ba92969266d    "docker-entrypoint.s..."   4 minutes ago     Up 4 minutes       0.0.0.0:3000->3000/tcp   boring_swirles  
[node1] (local) root@192.168.0.18 ~/app  
$ docker stop 82abb  
82abb  
[node1] (local) root@192.168.0.18 ~/app  
$ docker rm 82abb  
82abb  
[node1] (local) root@192.168.0.18 ~/app  
$ docker run -dp 3000:3000 docker-101  
82e5eeef52f95eb13ffcbac0bea508167498692e9826a21a03287e4477f0c5328
```

DOCKER

OPEN PORT

3000

Memory
4.30% (171.8MiB / 3.906GiB)

CPU
0.07%

SSH
ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
82e5eeef52f95eb13ffcbac0bea508167498692e9826a21a03287e4477f0c5328
[node1] (local) root@192.168.0.18 ~/app
$ docker push kalyanjs/101-todo-app:tagname
The push refers to repository [docker.io/kalyanjs/101-todo-app]
An image does not exist locally with the tag: kalyanjs/101-todo-app
[node1] (local) root@192.168.0.18 ~/app
$ docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED
SIZE
docker-101          latest   644b44cc351b  5 minutes ago
171MB
<none>              <none>   2ba92969266d  12 minutes ago
171MB
<none>              <none>   c6a039b453e8  22 minutes ago
171MB
<none>              <none>   8ed8bc274e85  53 minutes ago
87.4MB
```

DOCKER

OPEN PORT

3000

Memory

4.29% (171.7MiB / 3.906GiB)

CPU

0.09%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

```
[node1] (local) root@192.168.0.18 ~/app
$ docker tag docker-101 kalyanjs/101-todo-app
[node1] (local) root@192.168.0.18 ~/app
$ docker push kalyanjs/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/kalyanjs/101-todo-app]
eb2034ee0308: Preparing
a180041d99cb: Preparing
c9a6802e6cec: Preparing
edff9ff691d5: Preparing
cbe4b9146f86: Preparing
a6524c5b12a6: Waiting
9a5d14f9f550: Waiting
denied: requested access to the resource is denied
[node1] (local) root@192.168.0.18 ~/app
$ docker login
Log in with your Docker ID or email address to push and pull images fr
```

DOCKER

```
edff9ff691d5: Preparing
cbe4b9146f86: Preparing
a6524c5b12a6: Waiting
9a5d14f9f550: Waiting
denied: requested access to the resource is denied
[node1] (local) root@192.168.0.18 ~/app
$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: kalyanjs
Password:
```

DOCKER

OPEN PORT

3000

Memory

4.29% (171.8MiB / 3.906GiB)

CPU

0.08%

SSH

ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v

DELETE

 EDITOR

```
Error response from daemon: Get "https://registry-1.docker.io/v2/": un
authorized: incorrect username or password
[node1] (local) root@192.168.0.18 ~/app
$
[node1] (local) root@192.168.0.18 ~/app
$ docker login kalyanjs
Username:
Error: Non-null Username Required
[node1] (local) root@192.168.0.18 ~/app
$
[node1] (local) root@192.168.0.18 ~/app
$ docker login -u kalyanjs
Password:
Error response from daemon: Get "https://registry-1.docker.io/v2/": un
authorized: incorrect username or password
[node1] (local) root@192.168.0.18 ~/app
$ docker login[]
```

DOCKER

clasrdgg_claum86fml8g009genh0

IP

192.168.0.17

OPEN PORT

Memory

0.99% (39.45MiB / 3.906GiB)

CPU

0.11%

SSH

ssh ip172-18-0-67-clasrdggftqg00bver6g@direct.labs.play-wi 

DELETE

 EDITOR

```
# is HIGHLY! discouraged. Any consequences of doing so are      #
# completely the user's responsibilities.                      #
#
# The PWD team.                                              #
#####
[node2] (local) root@192.168.0.17 ~
$ docker run -dp 3000:3000 kalyanjs/101-todo-app
Unable to find image 'kalyanjs/101-todo-app:latest' locally
latest: Pulling from kalyanjs/101-todo-app
ddad3d7c1e96: Pull complete
de915e575d22: Extracting  21.79MB/22.21MB
7150aa69525b: Download complete
```

DOCKER

4. Now try your push command again. If you're copying the value from Docker Hub, you can drop the `tagname` portion, as we didn't add a tag to the image name.

```
docker push YOUR-USER-NAME/101-todo-app
```

Running our Image on a New Instance

Now that our image has been built and pushed into a registry, let's try running our app on a brand instance that has never seen this container!

1. Back in PWD, click on **Add New Instance** to create a new instance.
2. In the new instance, start your freshly pushed app.

```
docker run -dp 3000:3000 YOUR-USER-NAME/101-todo-app
```

You should see the image get pulled down and eventually start up!

3. Click on the 3000 badge when it comes up and you should see the app with your modifications! Hooray!

Recap

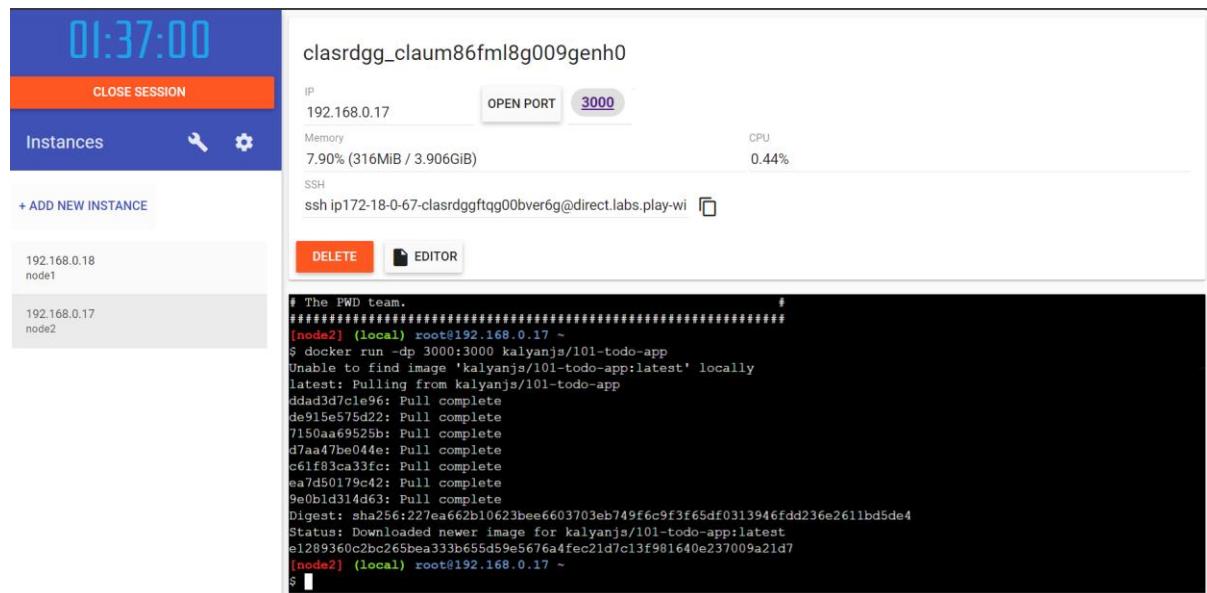
DOCKER

New Item

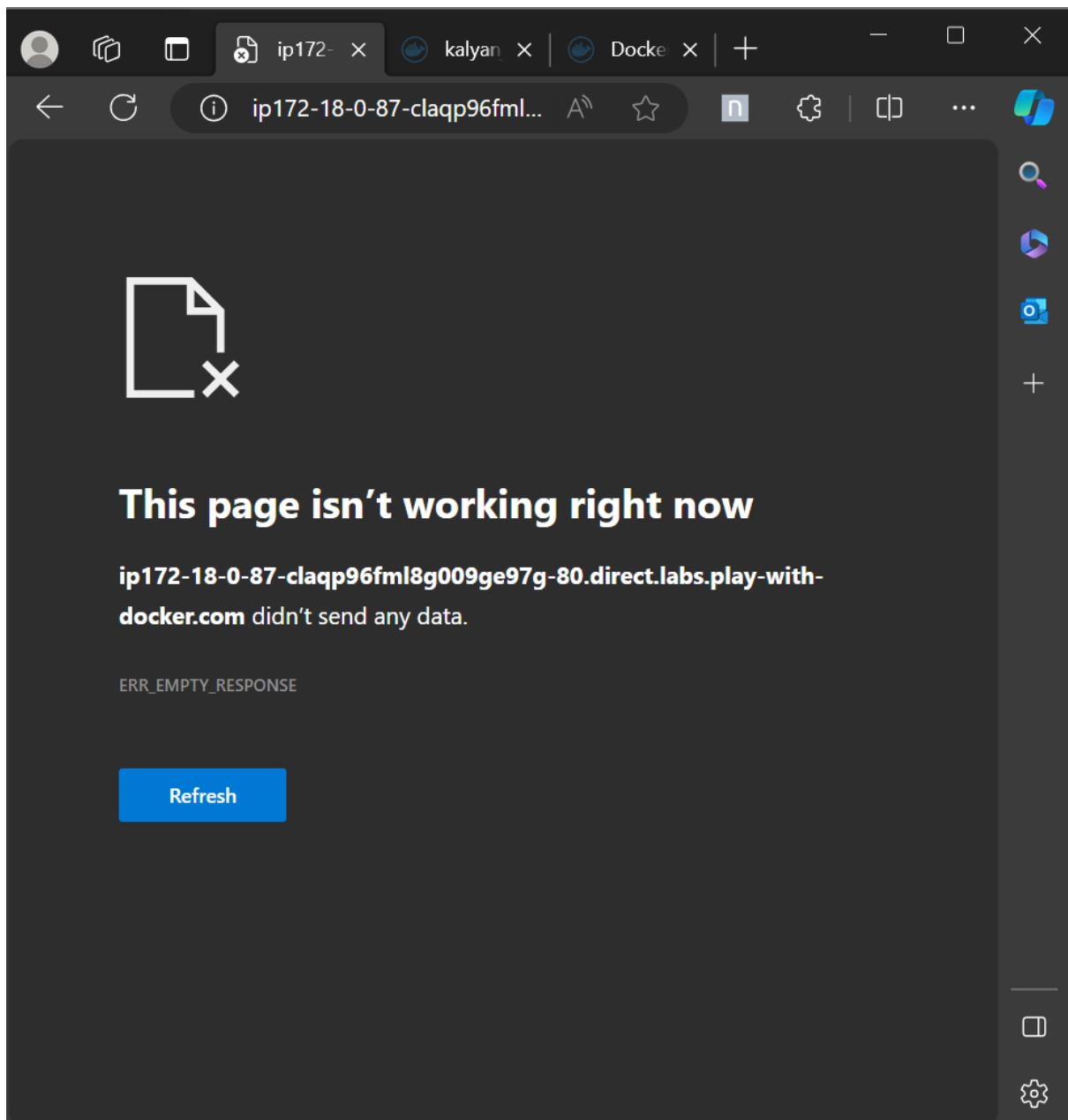
Add Item

You have no todo items yet! Add one above!

DOCKER



DOCKER



DOCKER

The screenshot shows a Docker container management interface. At the top, the IP address 192.168.0.18 is displayed. Below it, there is a button labeled "OPEN PORT" and a highlighted port number "3000". Under the "Memory" section, it shows 3.05% (121.9MiB / 3.906GiB) used. In the "CPU" section, the usage is listed as 0.36%. The "SSH" section contains the command "ssh ip172-18-0-106-clasrdggftqg00bver6g@direct.labs.play-v" followed by a copy icon. At the bottom, there are two buttons: "DELETE" and "EDITOR". A large black box at the bottom displays a terminal session history:

```
[node1] (local) root@192.168.0.18 ~/app
$ docker tag docker-101 kalyanjs/101-todo-app
[node1] (local) root@192.168.0.18 ~/app
$ docker push kalyanjs/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/kalyanjs/101-todo-app]
eb2034ee0308: Pushed
a180041d99cb: Pushed
c9a6802e6cec: Pushed
edff9ff691d5: Mounted from library/node
cbe4b9146f86: Mounted from library/node
a6524c5b12a6: Mounted from library/node
9a5d14f9f550: Mounted from library/node
latest: digest: sha256:227ea662b10623bee6603703eb749f6c9f3f65df0313946
ffd236e2611bd5de4 size: 1787
```

DOCKER



Sharing our App



```
docker push YOUR-USER-NAME/101-todo-app
```

Running our Image on a New Instance

Now that our image has been built and pushed into a registry, let's try running our app on a brand instance that has never seen this container!

1. Back in PWD, click on **Add New Instance** to create a new instance.
2. In the new instance, start your freshly pushed app.

```
docker run -dp 3000:3000 YOUR-USER-NAME/101-todo-app
```



You should see the image get pulled down and eventually start up!

3. Click on the 3000 badge when it comes up and you should see the app with your modifications! Hooray!

Recap

In this section, we learned how to share our images by pushing them to a registry. We then went to a brand new instance and were able to run the

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 kalyanjs/101-todo-app
Unable to find image 'kalyanjs/101-todo-app:latest' locally
latest: Pulling from kalyanjs/101-todo-app
ddad3d7c1e96: Already exists
de915e575d22: Already exists
7150aa69525b: Already exists
d7aa47be044e: Already exists
c61f83ca33fc: Pull complete
ea7d50179c42: Pull complete
9e0b1d314d63: Pull complete
Digest: sha256:227ea662b10623bee6603703eb749f6c9f3f65df0313946fdd236e2
611bd5de4
Status: Downloaded newer image for kalyanjs/101-todo-app:latest
4a4f71f699e2bbb64f1f2fbdeca9efc02745222075e5bf176cf1c4efa3830fb5
[node1] (local) root@192.168.0.28 ~/app
$
```

The Container's Filesystem

When a container runs, it uses the various layers from an image for its filesystem. Each container also gets its own "scratch space" to create/update/remove files. Any changes won't be seen in another container, *even if* they are using the same image.

Start a ubuntu container that will create a file named /data.txt with a random number between 1 and 10000.

```
docker run -d ubuntu bash -c "shuf -i 1-10000 -n 1 -o /data.txt && tail -f /dev/null"
```

DOCKER

In case you're wondering about the command, we're starting a bash shell and invoking two commands (why we have the `&&`). The first portion picks a single random number and writes it to `/data.txt`. The second command is simply watching a file to keep the container running.

```
192.168.0.28
OPEN PORT

Memory

CPU

SSH
ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi
 

DELETE EDITOR

$ docker run -d ubuntu bash -c "shuf -i 1-10000 -n 1 -o /data.txt && tail -f /dev/null
>
> docker ps
> ^C
[node1] (local) root@192.168.0.28 ~/app
$ docker run -d ubuntu bash -c "shuf -i 1-10000 -n 1 -o /data.txt && tail -f /dev/null"
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e49
6e5c1f05f
Status: Downloaded newer image for ubuntu:latest
e9ed8e21c9fabb837982b1b69e079cb9ab514324c7ca59e0e4e66b372a6ed3b8
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

```
$ docker ps
CONTAINER ID        IMAGE               COMMAND
CREATED             STATUS              PORTS
e9ed8e21c9fa      ubuntu              "bash -c 'shuf -i 1-..."   3
4 seconds ago      Up 33 seconds
anujan
4a4f71f699e2      kalyanjs/101-todo-app    "docker-entrypoint.s..."  7
4 minutes ago      Up 7 minutes       0.0.0.0:3000->3000/tcp    priceless_lumiere
92e8af2fb46f      docker/getting-started:pwd  "nginx -g 'daemon of..."  2
4 minutes ago      Up 24 minutes      0.0.0.0:80->80/tcp      quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$ docker exec e9ed8 cat /data.txt
2644
[node1] (local) root@192.168.0.28 ~/app
$ [
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
4 seconds ago  Up 33 seconds          quizzical_ram
anujan
4a4f71f699e2  kalyanjs/101-todo-app  "docker-entrypoint.s..."  7
 minutes ago   Up 7 minutes      0.0.0.0:3000->3000/tcp  priceless_lum
iere
92e8af2fb46f  docker/getting-started:pwd  "nginx -g 'daemon of..."  2
4 minutes ago   Up 24 minutes     0.0.0.0:80->80/tcp       quizzical_nob
el
[node1] (local) root@192.168.0.28 ~/app
$ docker exec e9ed8 cat /data.txt
2644
[node1] (local) root@192.168.0.28 ~/app
$ docker run -it ubuntu ls /
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot etc  lib  lib64  media  opt  root  sbin  sys  usr
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

DELETE EDITOR

```
iere
92e8af2fb46f    docker/getting-started:pwd    "nginx -g 'daemon of..."    2
8 minutes ago    Up 28 minutes    0.0.0.0:80->80/tcp        quizzical_nob
el
[node1] (local) root@192.168.0.28 ~/app
$ docker rm -f
"docker rm" requires at least 1 argument.
See 'docker rm --help'.

Usage: docker rm [OPTIONS] CONTAINER [CONTAINER...]

Remove one or more containers
[node1] (local) root@192.168.0.28 ~/app
$ docker rm -f e9ed8
e9ed8
[node1] (local) root@192.168.0.28 ~/app
$
```

[Volumes](#) provide the ability to connect specific filesystem paths of the container back to the host machine. If a directory in the container is mounted, changes in that directory are also seen on the host machine.

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
4 minutes ago   Up 24 minutes   0.0.0.0:80->80/tcp      quizzical_nob
el
[node1] (local) root@192.168.0.28 ~/app
$ docker exec e9ed8 cat /data.txt
2644
[node1] (local) root@192.168.0.28 ~/app
$ docker run -it ubuntu ls /
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot etc  lib  lib64  media  opt  root  sbin  sys  usr
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND
REATED             STATUS              PORTS
e9ed8e21c9fa        ubuntu              "bash -c 'shuf -i 1-...
minutes ago        Up 4 minutes
anujan
4a4f71f699e2        kalyanjs/101-todo-app    "docker-entrypoint.s..."  1
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
See 'docker rm --help'.
```

```
Usage: docker rm [OPTIONS] CONTAINER [CONTAINER...]
```

```
Remove one or more containers
```

```
[node1] (local) root@192.168.0.28 ~/app
$ docker rm -f e9ed8
e9ed8
[node1] (local) root@192.168.0.28 ~/app
$ docker volume create todo-db
todo-db
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
docker: Error response from daemon: invalid volume specification: 'todo-db:/etc/todos': invalid mount config for type "volume": invalid mount path: '/etc/todos' mount path must be absolute.
See 'docker run --help'.
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit

DELETE

 EDITOR

```
See 'docker run --help'.
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
0d8a203fa71e18712a9a69d1521cd4a382ffdba65939f54d36fee9844b3fae26
docker: Error response from daemon: driver failed programming external
connectivity on endpoint brave_carson (3d8a770280d226c1e7b5bdc056b08f
8bb2f272b29fc282003d31ed592c2c7287): Bind for 0.0.0.0:3000 failed: por
t is already allocated.
[node1] (local) root@192.168.0.28 ~/app
$ docker volume ls
DRIVER      VOLUME NAME
local      todo-db
[node1] (local) root@192.168.0.28 ~/app
$ docker volume rm todo-db
Error response from daemon: remove todo-db: volume is in use - [0d8a20
3fa71e18712a9a69d1521cd4a382ffdba65939f54d36fee9844b3fae26]
[node1] (local) root@192.168.0.28 ~/app
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
[node1] (local) root@192.168.0.28 ~/app
$ docker volume ls
DRIVER      VOLUME NAME
local      todo-db
[node1] (local) root@192.168.0.28 ~/app
$ docker volume rm todo-db
Error response from daemon: remove todo-db: volume is in use - [0d8a20
3fa71e18712a9a69d1521cd4a382ffdba65939f54d36fee9844b3fae26]
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
2896c113fd67a7bfbbbe827a92ee95ef837d46763f7fb447ba5b647c20b8dec4
docker: Error response from daemon: driver failed programming external
  connectivity on endpoint priceless_visvesvaraya (fb38b83105e10de912d7
2828abcd3f1cee3b509d6e06747c6ad3e8aa09fecdd): Bind for 0.0.0.0:3000 f
ailed: port is already allocated.
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit



DELETE

EDITOR

```
$ docker exec 46f3 cat /data.txt
7315
[node1] (local) root@192.168.0.28 ~/app
$ docker run -it ubuntu ls /
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr
[node1] (local) root@192.168.0.28 ~/app
$ docker rm -f 46f3cd
46f3cd
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND
CREATED             STATUS              PORTS
92e8af2fb46f      docker/getting-started:pwd   "nginx -g 'daemon off..."   4
8 minutes ago       Up 48 minutes       0.0.0.0:80->80/tcp    quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

```
[node1] (local) root@192.168.0.28 ~/app
$ docker rm -f 46f3cd
46f3cd
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND
REATED             STATUS              PORTS
92e8af2fb46f      docker/getting-started:pwd   "nginx -g 'daemon off..."  4
8 minutes ago      Up 48 minutes       0.0.0.0:80->80/tcp    quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$ docker volume create todo-db
todo-db
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
509c8d2b3aeb955a3b47af616230ea75a55fb8c73226166b7ffff07a44d3768a
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
todo-db
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
509c8d2b3aeb955a3b47af616230ea75a55fb8c73226166b7ffff07a44d3768a
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND            C
CREATED             STATUS              PORTS              NAMES
92e8af2fb46f      docker/getting-started:pwd   "nginx -g 'daemon off..."   5
3 minutes ago      Up 53 minutes       0.0.0.0:80->80/tcp   quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
02034246d6c6982f7720f3c1b9407059a884932dc0010c62f8dc8fa833066d5
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

DELETE

EDITOR

```
CONTAINER ID  IMAGE                      COMMAND          C
REATED        STATUS         PORTS          NAMES
92e8af2fb46f  docker/getting-started:pwd  "nginx -g 'daemon of..."  5
3 minutes ago  Up 53 minutes   0.0.0.0:80->80/tcp  quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$ [node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 -v todo-db:/etc/todos docker-101
02034246d6c6982f7720f3c1b9407059a884932dc0010c62f8dc8fa833066d5
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID  IMAGE                      COMMAND          C
REATED        STATUS         PORTS          NAMES
92e8af2fb46f  docker/getting-started:pwd  "nginx -g 'daemon of..."  5
4 minutes ago  Up 54 minutes   0.0.0.0:80->80/tcp  quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

New Item

Add Item

finish the workshop



DO amazing things



Be awesome



DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit 

DELETE

 EDITOR

```
$ docker rm -f 0204
Error response from daemon: No such container: 0204
[node1] (local) root@192.168.0.28 ~/app
$ docker volume inspect todo-db
[
    {
        "CreatedAt": "2023-11-16T12:17:05Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/todo-db/_data",
        "Name": "todo-db",
        "Options": null,
        "Scope": "local"
    }
]
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

The screenshot shows a web-based Docker management interface. At the top, there's a header bar with a back arrow, a refresh button, a URL field containing <https://labs.play-wit...>, a search icon, a font size icon, a star icon, a user icon, a gear icon, a copy icon, and a more options icon.

The main area displays a container configuration for IP 192.168.0.28. It includes sections for "OPEN PORT", "Memory", "CPU", and "SSH". The SSH section shows a command line:

```
ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit
```

Below the container configuration are two buttons: "DELETE" (orange) and "EDITOR" (with a file icon).

The bottom half of the screen is a terminal window showing the following commands and their output:

```
$ docker rm -f 0204
Error response from daemon: No such container: 0204
[node1] (local) root@192.168.0.28 ~/app
$ docker volume inspect todo-db
[
    {
        "CreatedAt": "2023-11-16T12:17:05Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/todo-db/_data",
        "Name": "todo-db",
        "Options": null,
        "Scope": "local"
    }
]
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

To run our container to support a development workflow, we will do the following:

- Mount our source code into the container
- Install all dependencies, including the "dev" dependencies
- Start nodemon to watch for filesystem changes

So, let's do it!

1. Make sure you don't have any previous `docker-101` containers running.
2. Run the following command. We'll explain what's going on afterwards:

```
docker run -dp 3000:3000 \
-w /app -v $PWD:/app \
node:10-alpine \
sh -c "yarn install && yarn run dev"
```

- `-dp 3000:3000` - same as before. Run in detached (background) mode and create a port mapping
- `-w /app` - sets the "working directory" or the current directory that the command will run from

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

DELETE EDITOR

```
[node1] (local) root@192.168.0.28 ~/app
$ docker run -dp 3000:3000 \
> -w /app -v $PWD:/app \
> node:10-alpine \
> sh -c "yarn install && yarn run dev"
Unable to find image 'node:10-alpine' locally
10-alpine: Pulling from library/node
ddad3d7c1e96: Already exists
de915e575d22: Already exists
7150aa69525b: Already exists
d7aa47be044e: Already exists
Digest: sha256:dc98dac24efd4254f75976c40bce46944697a110d06ce7fa47e7268
470cf2e28
Status: Downloaded newer image for node:10-alpine
ab8b77c9559ec23cb003b78834c9801126d9730e3dbad90a0760f45cc89a1e83
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

DELETE

```
470cf2e28
Status: Downloaded newer image for node:10-alpine
ab8b77c9559ec23cb003b78834c9801126d9730e3dbad90a0760f45cc89a1e83
[node1] (local) root@192.168.0.28 ~ /app
$ docker logs ab8
yarn install v1.22.5
[1/4] Resolving packages...
[2/4] Fetching packages...
info fsevents@1.2.9: The platform "linux" is incompatible with this module.
info "fsevents@1.2.9" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
Done in 61.56s.
yarn run v1.22.5
$ nodemon src/index.js
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE  **EDITOR**

```
info fsevents@1.2.9: The platform "linux" is incompatible with this module.
info "fsevents@1.2.9" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
Done in 61.56s.
yarn run v1.22.5
$ nodemon src/index.js
[nodemon] 1.19.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching dir(s): ***!
[nodemon] starting `node src/index.js`
Using sqlite database at /etc/todos/todo.db
Listening on port 3000
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

The screenshot shows a Docker-based development environment. At the top, a Microsoft Edge browser window displays a session from 'labs.play-with-docker.com'. The file tree shows a directory structure with files like 'react-dom.production.min.js', 'react.production.min.js', 'yarn.lock', and 'app.zip'. Below the file tree is an editor window for 'app.js' with code lines 103 to 114 visible. Buttons for 'Save' and 'Reload' are at the top of the editor. A terminal window below the editor shows command-line output for a 'yarn' run:

```
info "fsevents@1.2.9" is an optional dependency and failed compatibility check. Excluding it from installation.
[3/4] Linking dependencies...
[4/4] Building fresh packages...
Done in 61.56s.
yarn run v1.22.5
$ nodemon src/index.js
[nodemon] 1.19.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching dir(s): ***!
[nodemon] starting `node src/index.js`
Using sqlite database at /etc/todos/todo.db
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
[nodemon] watching dir(s): ***!
[nodemon] starting `node src/index.js`
Using sqlite database at /etc/todos/todo.db
Listening on port 3000
[node1] (local) root@192.168.0.28 ~/app
$ ^C
[node1] (local) root@192.168.0.28 ~/app
$ docker ps
CONTAINER ID        IMAGE               COMMAND
REATED             STATUS              PORTS
ab8b77c9559e      node:10-alpine      "docker-entrypoint.s..."   8
     minutes ago    Up 8 minutes       0.0.0.0:3000->3000/tcp   affectionate_el
lis
92e8af2fb46f      docker/getting-started:pwd  "nginx -g 'daemon off;"  2
     hours ago     Up 2 hours        0.0.0.0:80->80/tcp      quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$ docker ps -a
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
Usage: docker buildx build [OPTIONS] PATH | URL | -  
  
Start a build  
[node1] (local) root@192.168.0.28 ~/app  
$ docker ps  
CONTAINER ID   IMAGE          COMMAND      C  
REATED         STATUS        PORTS          NAMES  
ab8b77c9559e   node:10-alpine "docker-entrypoint.s..."  1  
1 minutes ago  Up 11 minutes  0.0.0.0:3000->3000/tcp  affectionate_ellis  
92e8af2fb46f   docker/getting-started:pwd  "nginx -g 'daemon off..."  2  
hours ago      Up 2 hours    0.0.0.0:80->80/tcp     quizzical_nobel  
[node1] (local) root@192.168.0.28 ~/app  
$ docker build -t docker-101 .  
[+] Building 139.4s (9/9) FINISHED          docker:default  
=> [internal] load build definition from Dockerfile 0.0s
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
1 minutes ago   Up 11 minutes   0.0.0.0:3000->3000/tcp   affectionate_ellis
92e8af2fb46f   docker/getting-started:pwd   "nginx -g 'daemon off..."  2
hours ago       Up 2 hours      0.0.0.0:80->80/tcp      quizzical_nobel
[node1] (local) root@192.168.0.28 ~/app
$ docker build -t docker-101 .
[+] Building 139.4s (9/9) FINISHED                               docker:default
=> [internal] load build definition from Dockerfile           0.0s
=> => transferring dockerfile: 146B                           0.0s
=> [internal] load .dockerignore                            0.0s
=> => transferring context: 2B                             0.0s
=> [internal] load metadata for docker.io/library/node:10-alpine 0.0s
=> [1/4] FROM docker.io/library/node:10-alpine               0.0s
=> [internal] load build context                           17.2s
=> => transferring context: 52.92MB                         17.1s
=> CACHED [2/4] WORKDIR /app                                0.0s
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit



DELETE

EDITOR

```
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 146B                         0.0s
=> [internal] load .dockerignore                            0.0s
=> => transferring context: 2B                           0.0s
=> [internal] load metadata for docker.io/library/node:10-alpi 0.0s
=> [1/4] FROM docker.io/library/node:10-alpine             0.0s
=> [internal] load build context                          17.2s
=> => transferring context: 52.92MB                      17.1s
=> CACHED [2/4] WORKDIR /app                            0.0s
=> [3/4] COPY . . .                                     5.0s
=> [4/4] RUN yarn install --production                98.4s
=> exporting to image                                    18.5s
=> => exporting layers                                 18.5s
=> => writing image sha256:ca0b12013f8aba37d57104d35ea8ec44b78 0.0s
=> => naming to docker.io/library/docker-101            0.0s
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit



DELETE

EDITOR

```
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/node:10-alpi 0.0s
=> [1/4] FROM docker.io/library/node:10-alpine 0.0s
=> [internal] load build context 17.2s
=> => transferring context: 52.92MB 17.1s
=> CACHED [2/4] WORKDIR /app 0.0s
=> [3/4] COPY . . 5.0s
=> [4/4] RUN yarn install --production 98.4s
=> exporting to image 18.5s
=> => exporting layers 18.5s
=> => writing image sha256:ca0b12013f8aba37d57104d35ea8ec44b78 0.0s
=> => naming to docker.io/library/docker-101 0.0s
[node1] (local) root@192.168.0.28 ~/app
$ docker network create todo-app
49bf3ede8e462668db9d69cb12ec17ab931a4dafb3447928b6fbf4497ddc09cc
[node1] (local) root@192.168.0.28 ~/app
$
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

 EDITOR

```
See 'docker run --help'.
[node1] (local) root@192.168.0.28 ~/app
$ docker run -d --network todo-app --network-alias mysql \
-v todo-mysq
ql-data:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=secret -e MYSQL_DATABASE=todos mysql:5.7
docker: invalid reference format.
See 'docker run --help'.
[node1] (local) root@192.168.0.28 ~/app
$ docker run -d \
> --network todo-app --network-alias mysql \
> -v todo-mysql-data:/var/lib/mysql \
> -e MYSQL_ROOT_PASSWORD=secret \
> -e MYSQL_DATABASE=todos \
> mysql:5.7
Unable to find image 'mysql:5.7' locally
5.7: Pulling from library/mysql
11a38aecbcb7a: Pull complete
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
11a38aebcb7a: Pull complete
91ab01309bd6: Pull complete
6c91fabb88c2: Pull complete
8f46e806ab5c: Pull complete
29f5af1d1661: Pull complete
62aca7179a54: Pull complete
85023e6de3be: Pull complete
6d5934a87cbb: Pull complete
c878502d3f70: Pull complete
4756467c684a: Pull complete
ee9043dd2677: Pull complete
Digest: sha256:f566819f2eee3a60cf5ea6c8b7d1bfc9de62e34268bf62dc34870c4
fca8a85d1
Status: Downloaded newer image for mysql:5.7
91668f26c7cd08202be88a29112559629f2cc0f7f9d4dbe5be4de636e070391
[node1] (local) root@192.168.0.28 ~/app
$ 
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit

DELETE

 EDITOR

```
[node1] (local) root@192.168.0.28 ~/app
$ docker exec -it 91668 mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.44 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi

DELETE **EDITOR**

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
| todos          |
+-----+
5 rows in set (0.00 sec)

mysql>
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE  **EDITOR**

```
+-----+
| Database           |
+-----+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| todos              |
+-----+
5 rows in set (0.00 sec)

mysql> docker run -it --network todo-app nicolaka/netshoot
      -> dig mysql
      ->
      -> ^C
mysql> ^C
mysql> █
```

DOCKER

The screenshot shows a Docker container interface with the following elements:

- Address Bar:** https://labs.play-wit... (partially visible)
- Container IP:** 192.168.0.28
- Actions:** OPEN PORT (button)
- Metrics:** Memory, CPU
- SSH:** ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wit (with a copy icon)
- Buttons:** DELETE (orange), EDITOR (with a file icon)
- Terminal History:** A scrollable list of commands and outputs:

```
-> ^C
mysql> exit
Bye
[node1] (local) root@192.168.0.28 ~/app
$ docker run -it --network todo-app nicolaka/netshoot
Unable to find image 'nicolaka/netshoot:latest' locally
latest: Pulling from nicolaka/netshoot
8a49fdb3b6a5: Pull complete
f08cc7654b42: Pull complete
bacdb080ad6d: Pull complete
df75a2676b1d: Pull complete
d30ac41fb6a9: Pull complete
3f3eebe79603: Pull complete
086410b5650d: Pull complete
4f4fb700ef54: Pull complete
5a7fe97d184f: Pull complete
a6d1b2d7a50e: Pull complete
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
082c4d4fd986 ~ 
082c4d4fd986 ~ dig mysql
; <>> DiG 9.18.13 <>> mysql
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50563
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;mysql.           IN      A
;; ANSWER SECTION:
mysql.          600     IN      A      172.19.0.2
```

DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
; Got answer:  
; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50563  
; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0  
  
; QUESTION SECTION:  
;mysql. IN A  
  
; ANSWER SECTION:  
mysql. 600 IN A 172.19.0.2  
  
; Query time: 0 msec  
; SERVER: 127.0.0.11#53(127.0.0.11) (UDP)  
; WHEN: Thu Nov 16 15:27:39 UTC 2023  
; MSG SIZE rcvd: 44
```

082c4d4fd986



DOCKER

192.168.0.28

OPEN PORT

Memory

CPU

SSH

ssh ip172-18-0-45-clavusggftqg00bi71jg@direct.labs.play-wi



DELETE

EDITOR

```
[node1] (local) root@192.168.0.28 ~/app
$ docker exec -it 91668 mysql -p todos
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.44 MySQL Community Server (GPL)

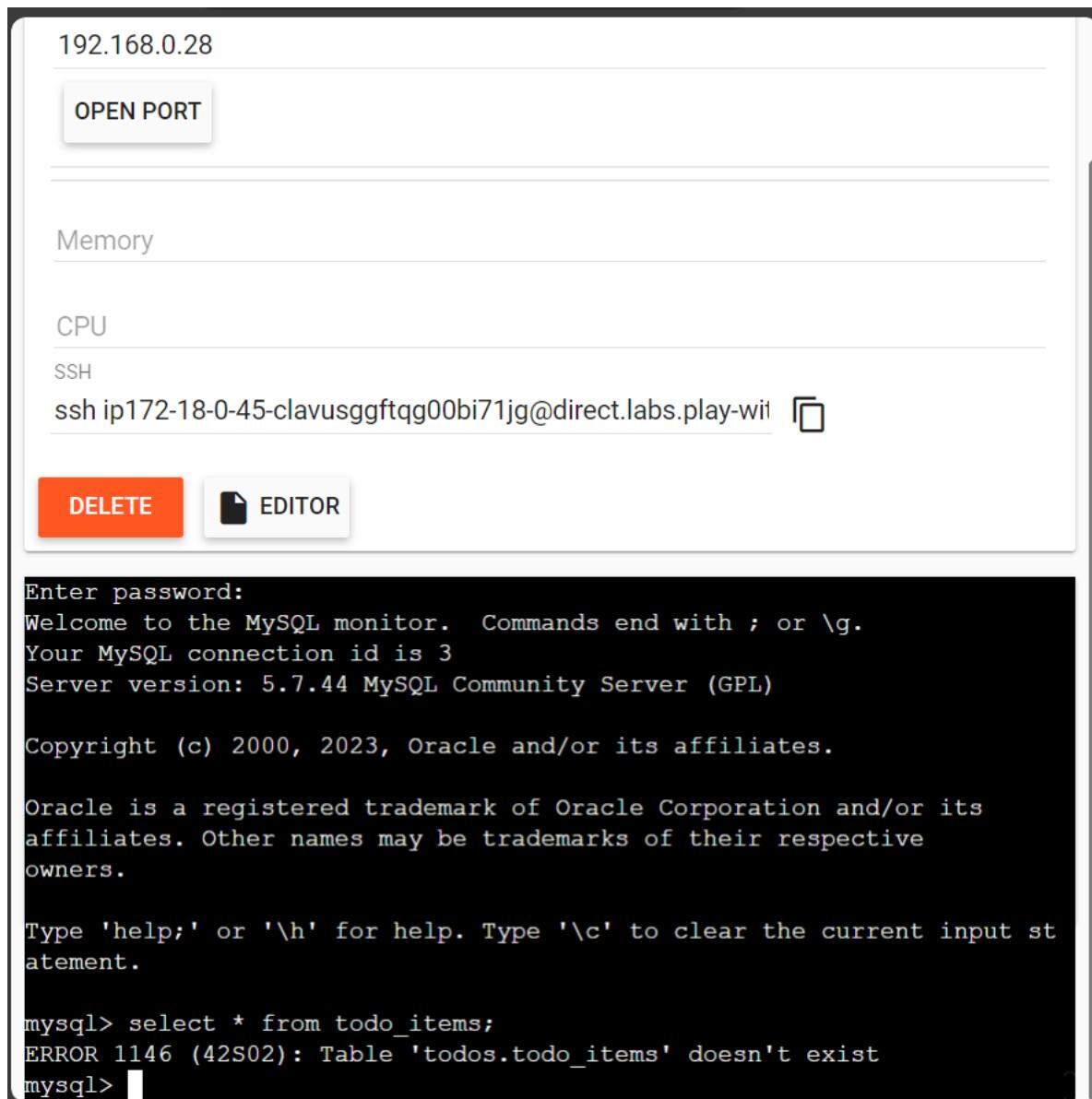
Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input st
atement.

mysql> 
```

DOCKER



o confirm we have the database up and running, connect to the database and verify it connects.

```
docker exec -it <mysql-container-id> mysql -p
```

DOCKER

```
1 version: "3.7"
2 services:
3 app:
4   image: node-10-alpine
5   command: sh -c "yarn install && yarn run dev"
6   ports:
7     - 3000:3000
8   working_dir: /app
9   volumes:
10    - ./:/app
11 environment:
12   MYSQL_HOST: mysql
13   MYSQL_USER: root
14   MYSQL_PASSWORD: secret
15   MYSQL_DB: todos
16
-- INSERT --
```

8,1 All

```
12   MYSQL_HOST: mysql
13   MYSQL_USER: root
14   MYSQL_PASSWORD: secret
15   MYSQL_DB: todos
16
17
18 mysql:
19   image: mysql:5.7
20   volumes:
21     - todo-mysql-data:/var/lib/mysql
22   environment:
23     MYSQL_ROOT_PASSWORD: secret
24     MYSQL_DATABASE: todos
25
26 volumes:
27 todo-mysql-data:
```

27,16 Bot

DOCKER

```
2023-11-16T16:40:57.254246Z 3 [Note] Unknown database 'todos'  
2023-11-16T16:41:38.341173Z 4 [Note] Unknown database 'todos'  
[node2] (local) root@192.168.0.17 ~/app  
$ docker-compose version  
Docker Compose version v2.23.0  
[node2] (local) root@192.168.0.17 ~/app  
$ version: "3.7"  
bash: version:: command not found  
[node2] (local) root@192.168.0.17 ~/app  
$ docker-compose.yml  
bash: docker-compose.yml: command not found  
[node2] (local) root@192.168.0.17 ~/app  
$ touch docker-compose.yml  
[node2] (local) root@192.168.0.17 ~/app  
$ vi docker-compose.yml  
[node2] (local) root@192.168.0.17 ~/app  
$ docker-compose up -d
```

SSH

ssh ip172-18-0-60-clb175efml8g009gev3g@direct.labs.play-v 

DELETE

 EDITOR

```
"docker image history" requires exactly 1 argument.  
See 'docker image history --help'.
```

```
Usage: docker image history [OPTIONS] IMAGE
```

```
Show the history of an image
```

```
[node2] (local) root@192.168.0.17 ~/app  
$ docker image history --no-trunc docker-101  
IMAGE  
CREATED          CREATED BY
```

DOCKER

```
0B          buildkit.  
dockerfile.v0  
<missing>  
2 years ago      /bin/sh -c #(nop)  CMD ["node"]
```

DOCKER

OPEN PORT

3000

Memory

8.05% (322.1MiB / 3.906GiB)

CPU

0.72%

SSH

ssh ip172-18-0-24-clb4qd6fml8g00emk28g@direct.labs.play- 

DELETE

 EDITOR

```
de915e575d22: Pull complete
7150aa69525b: Pull complete
d7aa47be044e: Pull complete
c61f83ca33fc: Pull complete
ea7d50179c42: Pull complete
9e0b1d314d63: Pull complete
Digest: sha256:227ea662b10623bee6603703eb749f6c9f3f65df0313946fdd236e2
511bd5de4
Status: Downloaded newer image for kalyanjs/101-todo-app:latest
02db9795842080e2a8366e144c933eae8b06d6c02844ed267c01085757d0ca3e
[node1] (local) root@192.168.0.18 ~
$ touch package.json
[node1] (local) root@192.168.0.18 ~
$ vi package.json
[node1] (local) root@192.168.0.18 ~
$
```

DOCKER

The screenshot shows a Docker container interface with the following details:

- OPEN PORT**: A button to open a port.
- 3000**: The port number is set to 3000.
- Memory**: Usage is 8.16% (326.3MiB / 3.906GiB).
- CPU**: Usage is 0.62%.
- SSH**: An SSH session is active, connected to ip172-18-0-24-clb4qd6fml8g00emk28g@direct.labs.play.
- DELETE**: A red button to delete the container.
- EDITOR**: A button to edit the Dockerfile.

The terminal window displays the following Dockerfile content:

```
1 FROM node:10-alpine
2 WORKDIR /app
3 COPY package.json yarn.lock ./ 
4 RUN yarn install --production
5 COPY . .
6 CMD ["node", "/app/src/index.js"]
7
8
```

File statistics: "package.json" 8L, 139B. The terminal shows the output "1,1 All".

DOCKER

The screenshot shows a web-based Docker interface. At the top, the URL is https://labs.play-wit... . The main content displays a container with the ID clb4qd6f_clb4qeefml8g00emk29g. Key information includes:

- IP:** 192.168.0.18
- OPEN PORT:** 3000
- Memory:** 8.06% (322.2MiB / 3.906GiB)
- CPU:** 0.66%
- SSH:** ssh ip172-18-0-24-clb4qd6fml8g00emk28g@direct.labs.play-wit...

Below these details are two buttons: **DELETE** (orange) and **EDITOR** (with a file icon).

A terminal window at the bottom shows the following command history and build output:

```
$ touch package.json
[node1] (local) root@192.168.0.18 ~
$ vi package.json
[node1] (local) root@192.168.0.18 ~
$ vi package.json
[node1] (local) root@192.168.0.18 ~
$ docker build -t docker-101 .
[+] Building 0.1s (2/2) FINISHED          docker:default
=> [internal] load build definition from Dockerfile      0.0s
=> => transferring dockerfile: 2B      0.0s
=> [internal] load .dockerignore      0.0s
```