

03:55:16

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 80 CPU: 0.80%

Memory: 2.67% (106.9MiB / 3.906GiB)

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

**DELETE** **EDITOR**

```
Unable to find image 'docker/getting-started:pwd' locally
pwd: Pulling from docker/getting-started
89d9c30c1d48: Pull complete
24f1c4f0b2f4: Pull complete
16542569a10d: Pull complete
08396939143d: Pull complete
Digest: sha256:9156d395e7e41498d5340e95513d61fc7929db720393448306c5d7263d7f2696
Status: Downloaded newer image for docker/getting-started:pwd
e22c7446a9alc28d067c0e96bb051a5921a09377df7b0690349631a9c714b458
[node1] (local) root@192.168.0.8 ~
$ pwd
/root
[node1] (local) root@192.168.0.8 ~
$
```

03:52:24

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 80 CPU: 0.26%

Memory: 2.69% (107.4MiB / 3.906GiB)

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

**DELETE** **EDITOR**

```
Uploading file(s) 1/1 : app.zip
Unable to find image 'docker/getting-started:pwd' locally
pwd: Pulling from docker/getting-started
89d9c30c1d48: Pull complete
24f1c4f0b2f4: Pull complete
16542569a10d: Pull complete
08396939143d: Pull complete
Digest: sha256:9156d395e7e41498d5340e95513d61fc7929db720393448306c5d7263d7f2696
Status: Downloaded newer image for docker/getting-started:pwd
e22c7446a9alc28d067c0e96bb051a5921a09377df7b0690349631a9c714b458
[node1] (local) root@192.168.0.8 ~
$ pwd
/root
[node1] (local) root@192.168.0.8 ~
$
```

Editor - Google Chrome

labs.play-with-docker.com/sessions/clat16efml8g009gehr0/instances/clat16ef\_clat17...  
mfmfml8g009gehs0

Create or upload files in the session terminal and then refresh

C  
/root  
app.zip

PORT 80

009gehs0

CPU 0.16%

gehr0@direct.labs.play-w

```
Unable to find image 'docker/getting-started:pwd' locally
pwd: Pulling from docker/getting-started
89d9c30c1d48: Pull complete
24f1c4f0b2f4: Pull complete
16542569a10d: Pull complete
08396939143d: Pull complete
Digest: sha256:9156d395e7e41498d340e95513d61fc7929db720393448306c5d7263d7f2696
Status: Downloaded newer image for docker/getting-started:pwd
e22c7446a9alc28d067c9e96bb051a5921a09377df7b0690349631a9c714b458
[node1] (local) root@192.168.0.8 ~
$ pwd
/root
[node1] (local) root@192.168.0.8 ~
$
```

34°C Smoke Search ENG IN 16-11-2023 13:57

03:50:41

CLOSE SESSION

Instances + ADD NEW INSTANCE

192.168.0.8 node1

IP 192.168.0.8 OPEN PORT 80

clat16ef\_clat17mfml8g009gehs0

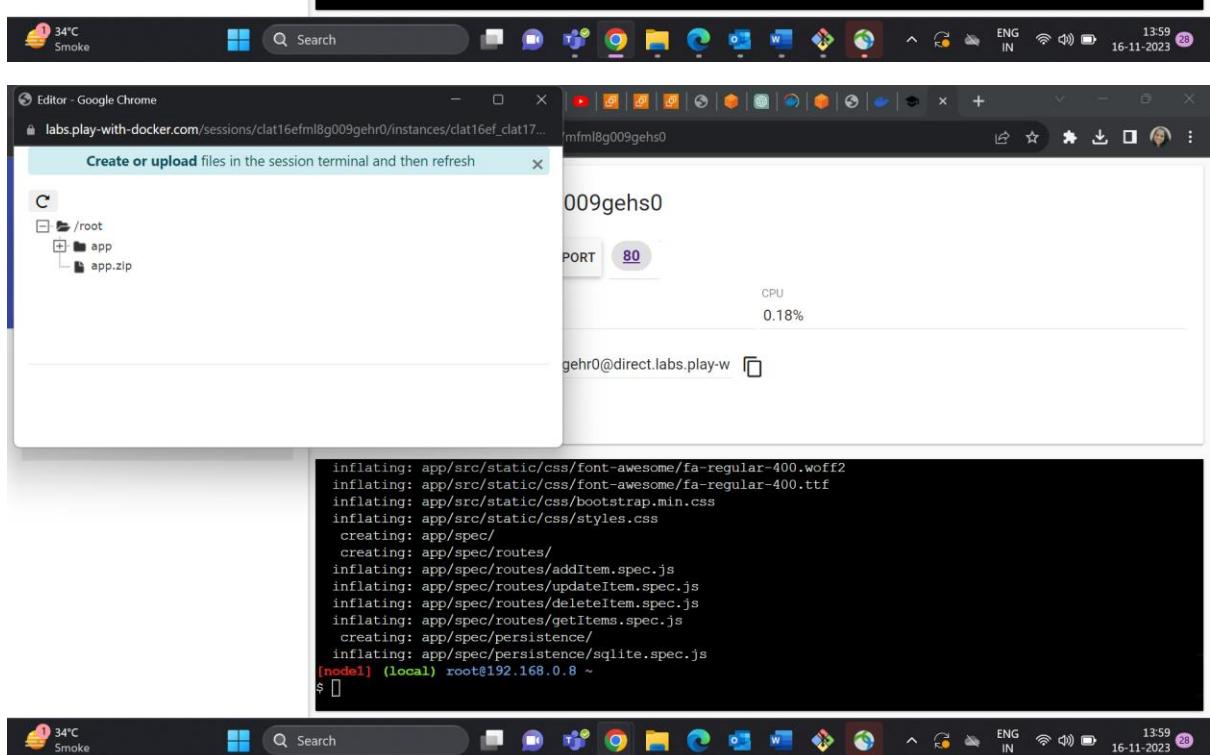
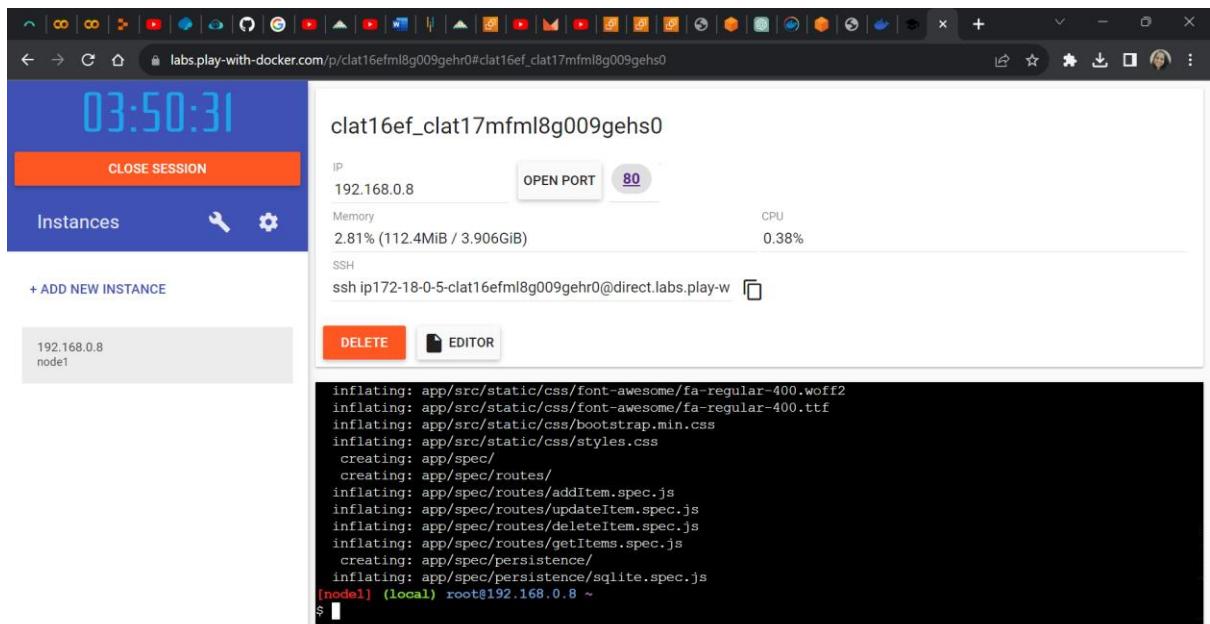
Memory 2.81% (112.3MiB / 3.906GiB) CPU 0.15%

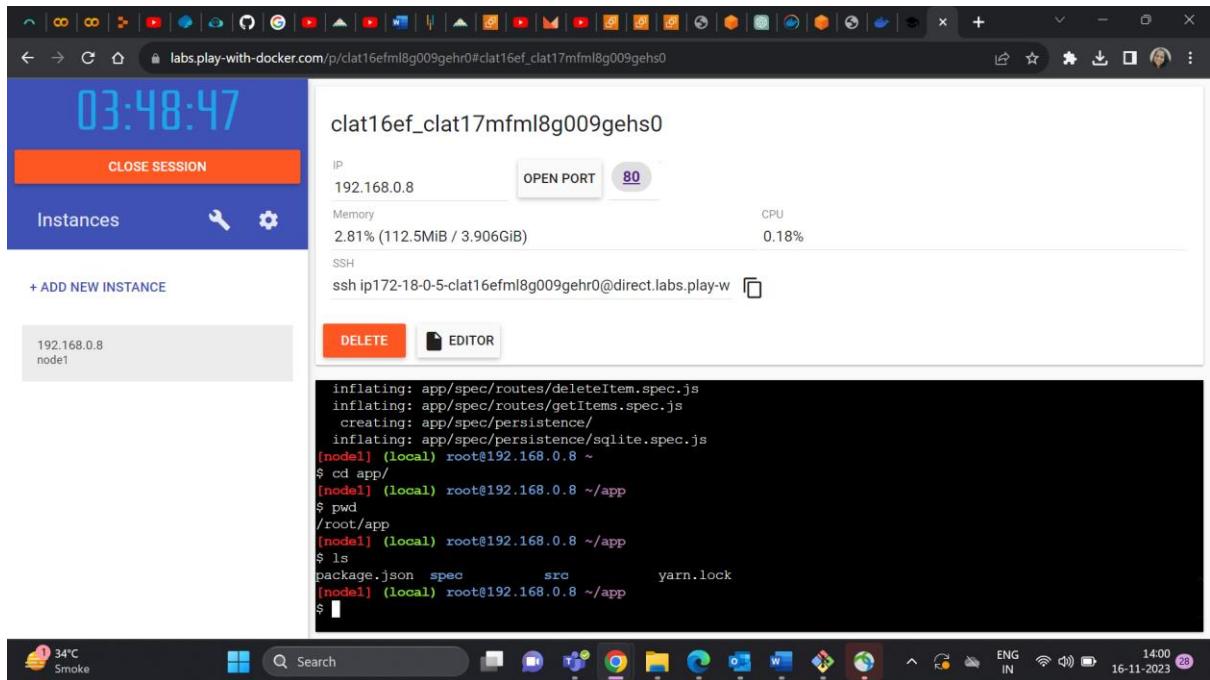
SSH ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

DELETE EDITOR

```
Digest: sha256:9156d395e7e41498d340e95513d61fc7929db720393448306c5d7263d7f2696
Status: Downloaded newer image for docker/getting-started:pwd
e22c7446a9alc28d067c9e96bb051a5921a09377df7b0690349631a9c714b458
[node1] (local) root@192.168.0.8 ~
$ pwd
/root
[node1] (local) root@192.168.0.8 ~
$ unzip app.zip
Archive: app.zip
  creating: app/
  inflating: app/package.json
  inflating: app/yarn.lock
  creating: app/src/
  creating: app/src/routes/
```

34°C Smoke Search ENG IN 16-11-2023 13:58





The screenshot shows a Docker tutorial page titled "Our Application". The left sidebar has a "Docker 101" section with links like "Tutorial", "Getting Started", "Our Application", "Updating our App", "Sharing our App", "Persisting our DB", "Using Bind Mounts", "Multi-Container Apps", "Using Docker Compose", "Image Building Best Practices", "What Next?", and "PWD Tips". The main content area has a "Table of contents" on the right with sections like "Getting our App into PWD", "Building the App's Container Image", "Starting an App Container", and "Recap". The main content shows steps for building a Docker image:

- Create a file named Dockerfile with the following contents:

```
FROM node:10-alpine
WORKDIR /app
COPY .
RUN yarn install --production
CMD ["node", "/app/src/index.js"]
```
- Build the container image using the `docker build` command:

```
docker build -t docker-101 .
```

The text explains that the Dockerfile uses the `node:10-alpine` image, which was not available on the local machine, so it was downloaded. It also mentions that `yarn` was used to install dependencies and that the `CMD` directive specifies the default command to run.

03:39:47

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 80

Memory: 2.76% (110.4MiB / 3.906GiB) CPU: 0.30%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

DELETE EDITOR

```
[node1] (local) root@192.168.0.8 ~/app
$ pwd
/root/app
[node1] (local) root@192.168.0.8 ~/app
$ ls
package.json spec      src      yarn.lock
[node1] (local) root@192.168.0.8 ~/app
$ vim Dockerfile
[node1] (local) root@192.168.0.8 ~/app
$ cat Dockerfile
FROM node:10-alpine
WORKDIR /app
COPY . .
RUN yarn install --production
```

03:37:43

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 80

Memory: 9.14% (365.6MiB / 3.906GiB) CPU: 0.15%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

DELETE EDITOR

```
COPY . .
RUN yarn install --production
CMD ["node", "/app/src/index.js"]
[node1] (local) root@192.168.0.8 ~/app
$ docker build -t docker-101 .
[+] Building 25.9s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 143B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/node:10-alpine
=> [1/4] FROM docker.io/library/node:10-alpine@sha256:dc98dac24efd4254f75976c40bce46944697a110d06ce7
=> => resolve docker.io/library/node:10-alpine@sha256:dc98dac24efd4254f75976c40bce46944697a110d06ce7
=> => sha256:ddad3d7cle96adf9153f8921a7c9790f880a390163df453be1566e9ef0d546e0 2.82MB / 2.82MB
```

The screenshot shows a Docker session interface. At the top, there's a header with icons and a URL: `labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clat17mfml8g009gehs0`. Below the header, a digital clock displays `03:37:32`. A red button labeled `CLOSE SESSION` is visible. The main area has tabs for `Instances` and `Editor`, with the `Instances` tab selected. It shows an instance with IP `192.168.0.8`, Memory usage `9.14% (365.6MiB / 3.906GiB)`, and CPU usage `0.26%`. An `OPEN PORT` button is set to port `80`. An SSH connection is established, with the command prompt `ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w` followed by a terminal window showing a Docker build log:

```

$ docker build -t docker-101 .
=> >> extracting sha256:de915e575d22c7e33c03fddaf7aeec0672e5d6a67e590a26fe0b30c7022f53cd
=> >> extracting sha256:7150aa69525b5f82b3df6a61a002f82362bf3ea8ce51b9000b965f7476a5cc
=> >> extracting sha256:d7aa47be044e5a988e3e7f204e2e28cb9f070daa32ed081072ad6d5bf6c095d1
=> [internal] load build context
=> => transferring context: 4.64MB
=> [2/4] WORKDIR /app
=> [3/4] COPY . .
=> [4/4] RUN yarn install --production
=> exporting to image
=> => exporting layers
=> => writing image sha256:8164f35bfea4d0c0c71861375d76199869d110401e8alc82370eaa526c574af
=> => naming to docker.io/library/docker-101
[node1] (local) root@192.168.0.8 ~/app
$ 

```

The screenshot shows a Docker tutorial page titled "Our Application". The left sidebar has a "Docker 101" menu with items like "Getting Started", "Our Application", "Updating our App", etc. The main content area has a heading "Starting an App Container". It says: "Now that we have an image, let's run the application! To do so, we will use the `docker run` command (remember that from earlier?).". Below this, step 1 is listed: "1. Start your container using the `docker run` command:" followed by a terminal window showing the command `docker run -dp 3000:3000 docker-101`. Step 2 is: "2. Open the application by clicking on the "3000" badge at the top of the PWD interface. Once open, you should have an empty todo list!". Below this, there's a screenshot of a browser showing a todo list with a single item "New Item" and a green "Add Item" button. The status bar at the bottom shows "3. Go ahead and add an item or two and see that it works as you expect. You can mark items" and the date "16-11-2023".

03:16:27

CLOSE SESSION

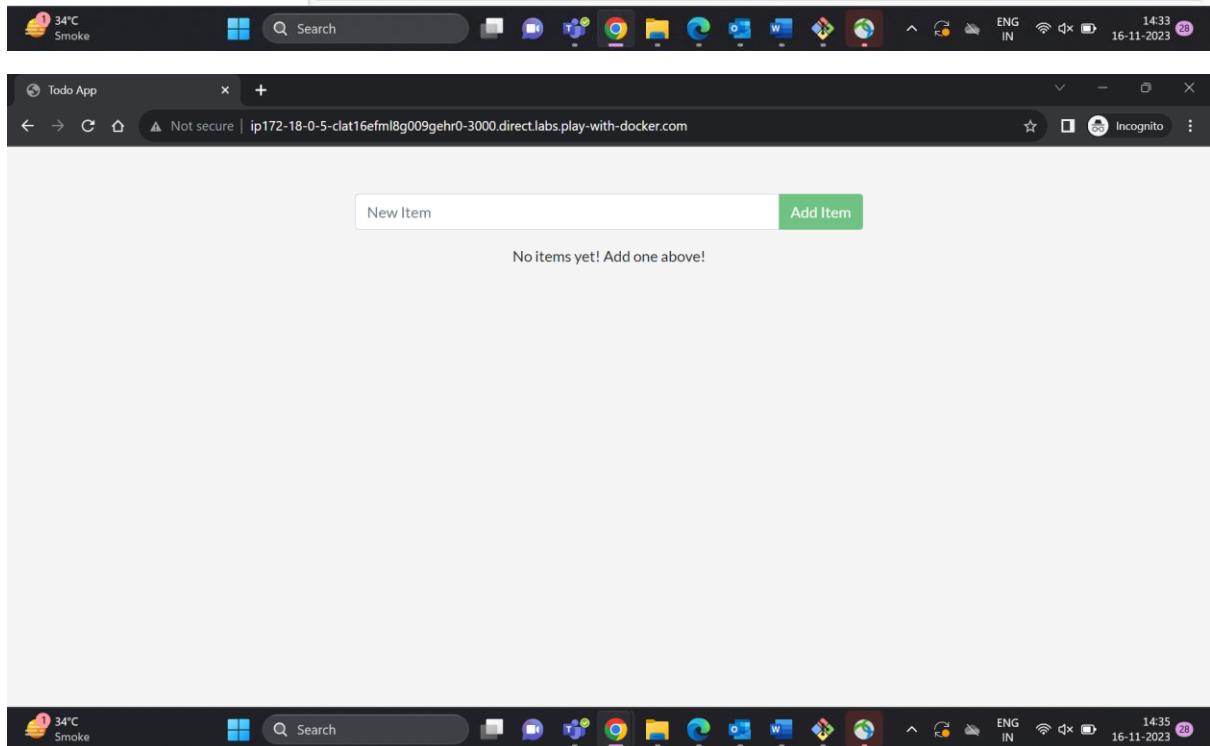
Instances

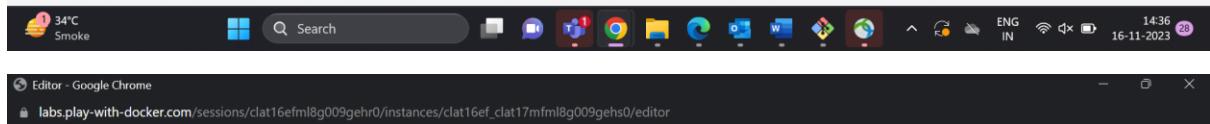
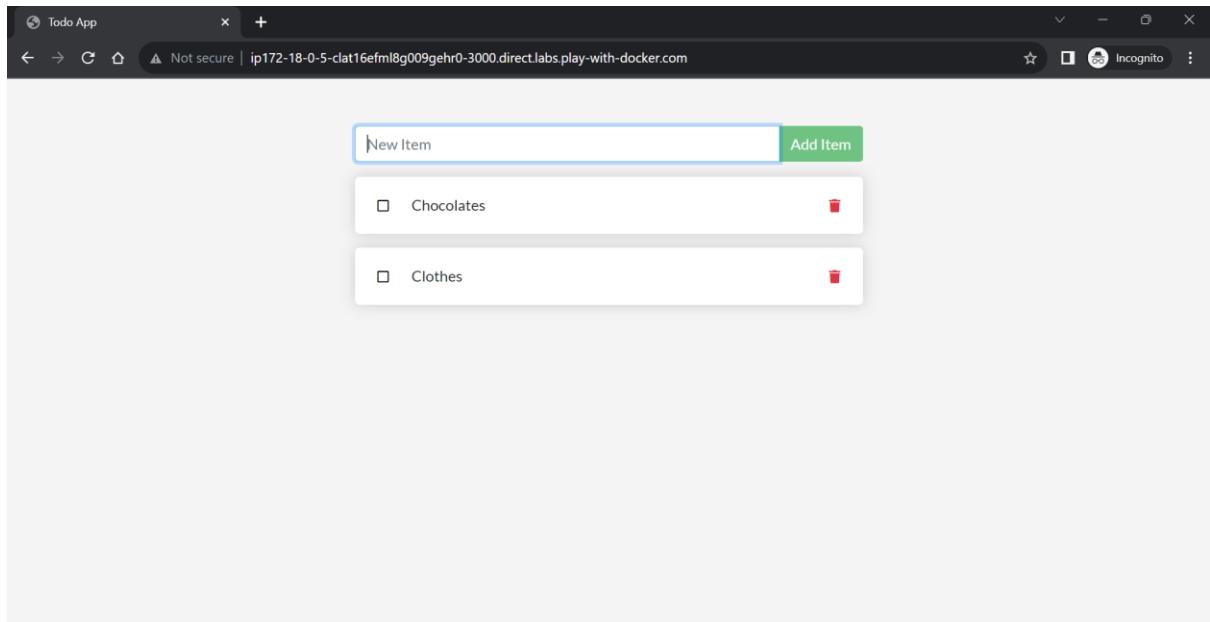
IP: 192.168.0.8 OPEN PORT: 3000 80

Memory: 9.75% (389.9MiB / 3.906GiB) CPU: 0.31%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w

```
→ naming to docker.io/library/docker-101
[node1] (local) root@192.168.0.8 ~/app
$ docker run -dp 3000:3000 docker-101
0829f0af05f41031f157e5620076d82cf1f2b018ef1cb6ebcb417d41d10129cc
[node1] (local) root@192.168.0.8 ~/app
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
NAMES
0829f0af05f4 docker-101 "docker-entrypoint.s..." 14 seconds ago Up 12 seconds 0.0.0.
0:3000->3000/tcp flamboyant_ptolemy
e22c7446a9a1 docker/getting-started:pwd "nginx -g 'daemon of..." 23 minutes ago Up 23 minutes 0.0.0.
0:80->80/tcp nice_mccarthy
[node1] (local) root@192.168.0.8 ~/app
$
```



A screenshot of a code editor window titled "app.js". The code is a React component that handles todo items. It includes logic for updating the state, rendering a loading message if no items are found, and displaying each item with its own delete button. A sidebar on the left shows the project structure with files like Dockerfile, package.json, index.js, routes, static, and js. A terminal tab is also visible with the instruction "Create or upload files in the session terminal and then refresh".

```
item => {
    const index = items.findIndex(i => i.id === item.id);
    setItems([...items.slice(0, index), ...items.slice(index + 1)]);
},
[items],
);

if (items === null) return 'Loading...';

return (
    <React.Fragment>
        <AddItemForm onNewItem={onNewItem} />
        {items.length === 0 && (
            <p className="text-center">You have no todo items yet! Add one above!</p>
        )}
        {items.map(item => (
            <ItemDisplay
                item={item}
                key={item.id}
                onItemUpdate={onItemUpdate}
                onItemRemoval={onItemRemoval}
            />
        ))}
    </React.Fragment>
);
}

function AddItemForm({ onNewItem }) {
    const { Form, InputGroup, Button } = ReactBootstrap;

    const [newItem, setNewItem] = React.useState('');
    const [submitting, setSubmitting] = React.useState(false);

    const submitNewItem = e => {
        e.preventDefault();
        setSubmitting(true);
    };
}
```



The screenshot shows a web browser window with the URL [ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/updating-our-app/](http://ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/updating-our-app/). The page title is "Updating our App". On the left, there's a sidebar titled "Docker 101" with various links like "Getting Started", "Our Application", "Sharing our App", etc. The main content area has a section titled "Updating our Source Code" with three numbered steps:

- In the `~/app/src/static/js/app.js` file, update line 56 to use the new empty text. ([Editing files in PWD tips here](#))  
Code snippet:

```
<p className="text-center">No items yet! Add one above!</p>
<p className="text-center">You have no todo items yet! Add one above!</p>
```
- Let's build our updated version of the image, using the same command we used before.  
Command:

```
docker build -t docker-101 .
```
- Let's start a new container using the updated code.  
Command:

```
docker run -dp 3000:3000 docker-101
```

A note says "Uh oh! You probably saw an error like this (the IDs will be different):" followed by a command output:

```
docker: Error response from daemon: driver failed programming external connectivity (bb242b2ca4d67eba76e79474fb36bb5125708ebdabd7f45c8eaf16caaabde9dd): Bind for 0.0.0.0.
```

The screenshot shows a web browser window with the URL [labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef\\_clat17mfml8g009gehs0](http://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clat17mfml8g009gehs0). The page title is "clat16ef\_clat17mfml8g009gehs0". It displays session details for an instance at IP 192.168.0.8, port 3000, and port 80. The session status is "OPEN PORT". The "EDITOR" tab is selected, showing a terminal session with the following commands and output:

```
=> [3/4] COPY .
=> [4/4] RUN yarn install --production
=> exporting to image
=> => exporting layers
=> => writing image sha256:dc18d76ea6cf740a3781f5bfa07542ff144d73abf616c7c4fd61745ad0b4eab1
=> => naming to docker.io/library/docker-101
[node1] (local) root@192.168.0.8 ~
$ docker run -dp 3000:3000 docker-101
$26091d9af55d08e16702cc88b3b45d0f88c9fa0679b4c71a436fe1318a9c3b3
docker: Error response from daemon: driver failed programming external connectivity on endpoint pedantic_gull (91b018d5cf2723bd598d08c399dd6bb604075858f40ac554f897b477fb879c46): Bind for 0.0.0.0:3000 failed: port is already allocated.
[node1] (local) root@192.168.0.8 ~
$
```

The screenshot shows a web browser window with the URL <http://ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/updating-our-app/>. The page title is "Updating our App" under "Docker 101". The main content is titled "Replacing our Old Container". It provides step-by-step instructions with terminal command examples:

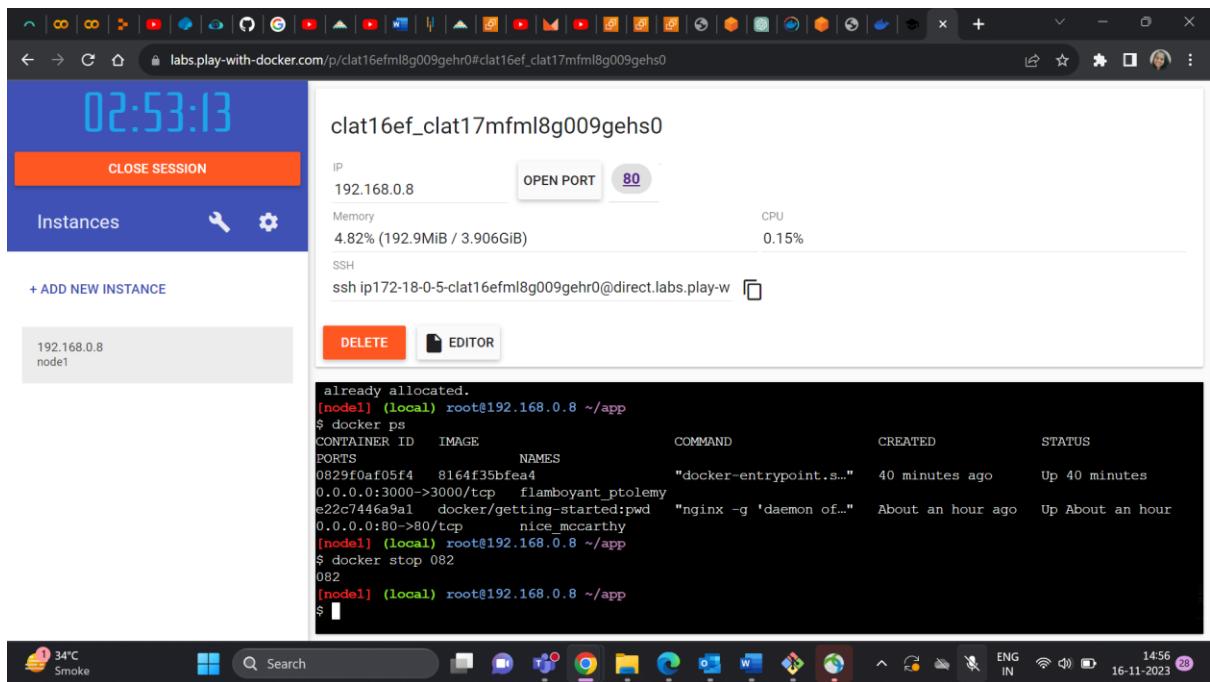
- Get the ID of the container by using the `docker ps` command.
- Use the `docker stop` command to stop the container.
- Once the container has stopped, you can remove it by using the `docker rm` command.
- Now, start your updated app.
- Open the app and you should see your updated help text!

At the bottom of the browser window, the taskbar shows various pinned icons and the date/time as 16-11-2023 14:54.

The screenshot shows a web browser window with the URL [http://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef\\_clat17mfml8g009gehs0](http://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clat17mfml8g009gehs0). The interface displays a session titled "clat16ef\_clat17mfml8g009gehs0" with the IP address 192.168.0.8. It shows memory usage (5.22% / 208.8MiB / 3.906GiB) and CPU usage (0.12%). An SSH terminal window is open, showing the following terminal session:

```
already allocated.
[node1] (local) root@192.168.0.8 ~/app
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
0829f0af05f4 816af35bfea4 "docker-entrypoint.s..." 40 minutes ago Up 40 minutes
0.0.0.0:3000->3000/tcp flamboyant_ptolemy
e22c7446a9al docker/getting-started:pwd "nginx -g 'daemon of..." About an hour ago Up About an hour
0.0.0.0:80->80/tcp nice_mccarthy
[node1] (local) root@192.168.0.8 ~/app
$ docker stop 082
082
[node1] (local) root@192.168.0.8 ~/app
$
```

At the bottom of the browser window, the taskbar shows various pinned icons and the date/time as 16-11-2023 14:56.



02:53:13

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8 node1

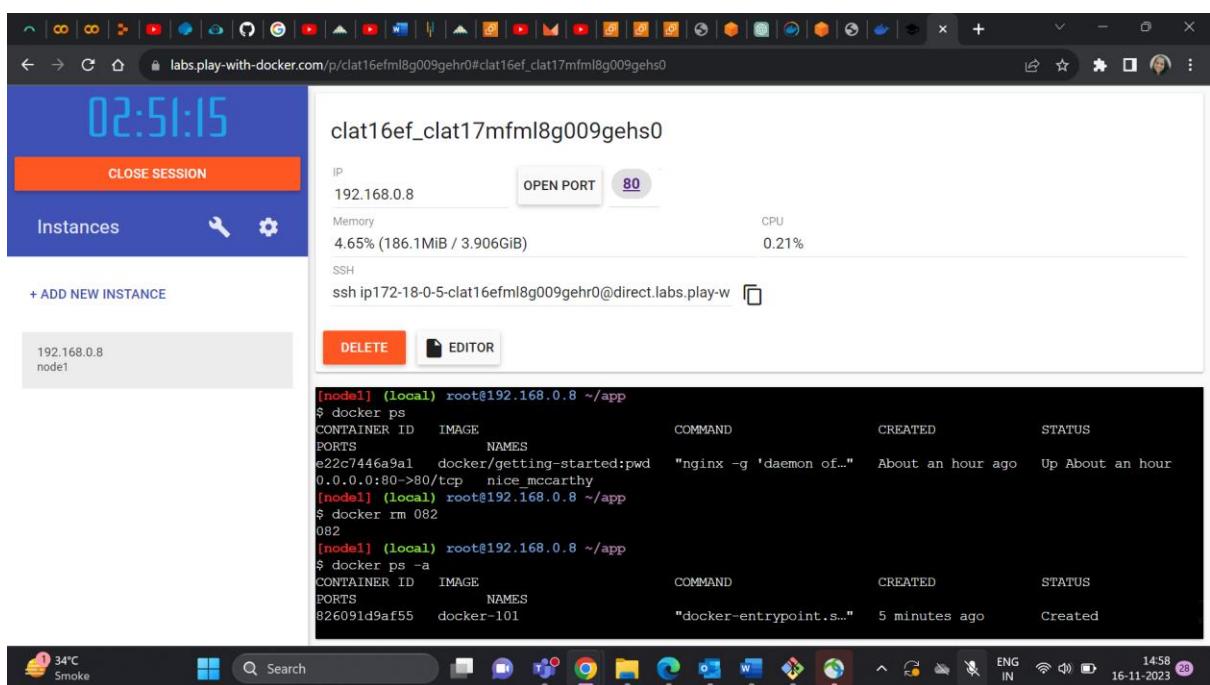
IP: 192.168.0.8 OPEN PORT: 80

Memory: 4.82% (192.9MiB / 3.906GiB) CPU: 0.15%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

DELETE EDITOR

```
already allocated.
[node1] (local) root@192.168.0.8 ~/app
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
0829f0af05f4 816af35bfea4 "docker-entrypoint.s..." 40 minutes ago Up 40 minutes
0.0.0.0:3000->3000/tcp flamboyant_ptolemy
e22c7446a9a1 docker/getting-started:pwd "nginx -g 'daemon of..." About an hour ago Up About an hour
0.0.0.0:80->80/tcp nice_mccarthy
[node1] (local) root@192.168.0.8 ~/app
$ docker stop 082
082
[node1] (local) root@192.168.0.8 ~/app
$
```



02:51:15

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 80

Memory: 4.65% (186.1MiB / 3.906GiB) CPU: 0.21%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

DELETE EDITOR

```
[node1] (local) root@192.168.0.8 ~/app
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
e22c7446a9a1 docker/getting-started:pwd "nginx -g 'daemon of..." About an hour ago Up About an hour
0.0.0.0:80->80/tcp nice_mccarthy
[node1] (local) root@192.168.0.8 ~/app
$ docker rm 082
082
[node1] (local) root@192.168.0.8 ~/app
$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
826091d9af55 docker-101 "docker-entrypoint.s..." 5 minutes ago Created
```

The screenshot shows a web-based interface for managing Docker sessions. At the top, there's a header with a close button, a refresh button, and a URL bar containing `labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clat17mfml8g009gehs0`. Below the header is a digital clock displaying **02:51:00**. A red button labeled **CLOSE SESSION** is visible. On the left, there's a sidebar titled **Instances** with a search icon and a gear icon. Below the sidebar, a link to **+ ADD NEW INSTANCE** is shown. The main area displays a terminal window with the following content:

```
0.0.0.0:80->80/tcp nice mccarthy
[node1] (local) root@192.168.0.8 ~/app
$ docker rm 082
082
[node1] (local) root@192.168.0.8 ~/app
$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
826091d9af55 docker-101 "docker-entrypoint.s..." 5 minutes ago Created
e22c7446a9a1 docker/getting-started:pwd "nginx -g 'daemon of..." About an hour ago Up About an hour
0.0.0.0:80->80/tcp nice mccarthy
[node1] (local) root@192.168.0.8 ~/app
$
```

This screenshot shows a browser window titled **Todo App**. The address bar indicates the URL is `ip172-18-0-5-clat16efml8g009gehr0-3000/direct.labs.play-with-docker.com`. The page content includes a text input field with placeholder text **New Item** and a green **Add Item** button. Below the input field, a message reads **You have no todo items yet! Add one above!**



To push an image, we first need to create a repo on Docker Hub.

1. Go to [Docker Hub](#) and log in if you need to.
2. Click the **Create Repository** button.
3. For the repo name, use `101-todo-app`. Make sure the Visibility is **Public**.
4. Click the **Create** button!

If you look on the right-side of the page, you'll see a section named **Docker commands**. This gives an example command that you will need to run to push to this repo.

**Docker commands**

[Public View](#)

```
docker push dockersamples/101-todo-app:tagname
```

Add a short description for this repository

The short description is used to index your content on Docker Hub and in search engines. It's visible to users in search results.

[Update](#)

**ruchikamoon15 / 101-todo-app**

Description

This repository does not have a description

Last pushed: a few seconds ago

**Docker commands**

To push a new tag to this repository:

```
docker push ruchikamoon15/101-todo-app:tagname
```

**Tags**

This repository is empty. Push some images to it to see them appear here.

**Automated Builds**

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#)

The screenshot shows a web browser window with the URL [ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/sharing-our-app/](http://ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/sharing-our-app/). The page title is "Sharing our App" under "Docker 101". The main content is titled "Pushing our Image". It includes a sidebar with "Docker 101" navigation and a "Table of contents" on the right. A code block shows an attempt to push an image:

```
$ docker push dockersamples/101-todo-app
The push refers to repository [docker.io/dockersamples/101-todo-app]
An image does not exist locally with the tag: dockersamples/101-todo-app
```

The text explains that the push failed because the image did not exist locally. It then guides the user through tagging the image and pushing it again.

The screenshot shows a web browser window with the URL [labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef\\_clat17mfml8g009gehs0](http://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clat17mfml8g009gehs0). The interface displays session information for "clat16ef\_clat17mfml8g009gehs0". It shows an IP of 192.168.0.8, ports 3000 and 80, and resource usage (Memory: 4.88%, CPU: 0.17%). An SSH terminal window is open, showing a successful login and a command-line session:

```
ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w
$ docker push ruchikamoon15/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/ruchikamoon15/101-todo-app]
An image does not exist locally with the tag: ruchikamoon15/101-todo-app
[node1] (local) root@192.168.0.8 ~./app
$ docker login -u ruchikamoon15
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[node1] (local) root@192.168.0.8 ~./app
$
```

02:30:28

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 3000 80

Memory: 8.06% (322.2MiB / 3.906GiB) CPU: 0.45%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

**DELETE** **EDITOR**

```
5970ac15404cb3170837fe26f26ca05f450a364125d71bd039c443f04c05ee39
[node1] (local) root@192.168.0.8 ~app
$ docker push ruchikamoon15/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/ruchikamoon15/101-todo-app]
An image does not exist locally with the tag: ruchikamoon15/101-todo-app
[node1] (local) root@192.168.0.8 ~app
$ docker login -u ruchikamoon15
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

02:30:10

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8 node1

IP: 192.168.0.8 OPEN PORT: 3000 80

Memory: 8.06% (322.2MiB / 3.906GiB) CPU: 0.46%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

**DELETE** **EDITOR**

```
Using default tag: latest
The push refers to repository [docker.io/ruchikamoon15/101-todo-app]
An image does not exist locally with the tag: ruchikamoon15/101-todo-app
[node1] (local) root@192.168.0.8 ~app
$ docker login -u ruchikamoon15
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[node1] (local) root@192.168.0.8 ~app
$ docker tag docker-101 ruchikamoon15/101-todo-app
[node1] (local) root@192.168.0.8 ~app
```

02:29:59

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

IP: 192.168.0.8 OPEN PORT: 3000 80

Memory: 8.06% (322.3MiB / 3.906GiB) CPU: 0.23%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

**DELETE** **EDITOR**

```
Login Succeeded
[node1] (local) root@192.168.0.8 ~/app
$ docker tag docker-101 ruchikamoon15/101-todo-app
[node1] (local) root@192.168.0.8 ~/app
$ docker push ruchikamoon15/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/ruchikamoon15/101-todo-app]
0f58267bbc6a: Pushed
4fd9017cb8bb: Pushed
453f7937ee0b: Pushed
edff9ff691d5: Mounted from library/node
cbe4b9146f86: Mounted from library/node
a6524c5b12a6: Mounted from library/node
9a5d14f9f550: Mounted from library/node
```

36°C Smoke Search ENG IN 15:19 16-11-2023

02:29:51

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

IP: 192.168.0.8 OPEN PORT: 3000 80

Memory: 8.06% (322.3MiB / 3.906GiB) CPU: 0.40%

SSH: ssh ip172-18-0-5-clat16efml8g009gehr0@direct.labs.play-w 

**DELETE** **EDITOR**

```
[node1] (local) root@192.168.0.8 ~/app
$ docker push ruchikamoon15/101-todo-app
Using default tag: latest
The push refers to repository [docker.io/ruchikamoon15/101-todo-app]
0f58267bbc6a: Pushed
4fd9017cb8bb: Pushed
453f7937ee0b: Pushed
edff9ff691d5: Mounted from library/node
cbe4b9146f86: Mounted from library/node
a6524c5b12a6: Mounted from library/node
9a5d14f9f550: Mounted from library/node
latest: digest: sha256:6b574495840bddcaad9db6e49edd9a13a56b363a4a8112e56a474059ff8f9c4f size: 1787
[node1] (local) root@192.168.0.8 ~/app
$
```

36°C Smoke Search ENG IN 15:19 16-11-2023

**Docker 101**

Tutorial ^

- Getting Started
- Our Application
- Updating our App
- Sharing our App
- Persisting our DB
- Using Bind Mounts
- Multi-Container Apps
- Using Docker Compose
- Image Building Best Practices
- What Next?
- PWD Tips

## 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 freshly pushed image. This is quite common in CI pipelines, where the pipeline will create the image and push it to a registry and then the production environment can use the latest version of the image.

02:27:02

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

### clat16ef\_clauc1ggftqg00bi6sq0

IP: 192.168.0.7 | OPEN PORT: 3000

Memory: 8.12% (325MiB / 3.906GiB) | CPU: 0.45%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-with-docker.com

```
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 ruchikamoon15/101-todo-app
Unable to find image 'ruchikamoon15/101-todo-app:latest' locally
latest: Pulling from ruchikamoon15/101-todo-app
ddad3d7c1e96: Pull complete
de915e575d22: Pull complete
7150aa9525b: Pull complete
d7aa47be044e: Pull complete
d50970ea009b: Pull complete
a086e2e25eaa: Pull complete
9bd357f942e4: Pull complete
Digest: sha256:6b574495840bddcaad9db6e49edd9a13a56b363a4a8112e56a474059ff8f9c4f
Status: Downloaded newer image for ruchikamoon15/101-todo-app:latest
66a86c055c2d5ed4d967eb37d3423b3f395df40a4f1d90e5802306bd39bfbe2f
```

The screenshot shows a web browser window with the URL [ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/persisting-our-data/](http://ip172-18-0-5-clat16efml8g009gehr0-80.direct.labs.play-with-docker.com/tutorial/persisting-our-data/). The page title is "Persisting our DB". On the left, there's a sidebar titled "Docker 101" with links like "Getting Started", "Our Application", "Updating our App", etc. The main content area has a section titled "Seeing this in Practice". It explains that two containers will be created to demonstrate file persistence. Step 1 shows a command to run an Ubuntu container with a random number file: 

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

. Step 2 shows how to exec into the container and cat the file: 

```
docker exec <container-id> cat /data.txt
```

. Step 3 indicates that starting another container won't have the same file. A "Table of contents" sidebar on the right lists topics such as "The Container's Filesystem", "Container Volumes", and "Recap".

The screenshot shows a web browser window with the URL [labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef\\_clauc1ggftqg00bi6sq0](http://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clauc1ggftqg00bi6sq0). The interface displays a session titled "clat16ef\_clauc1ggftqg00bi6sq0". It shows two instances: "node1" at IP 192.168.0.8 and "node2" at IP 192.168.0.7. The "node2" terminal window shows a terminal session where a new Ubuntu container is being run and a file is being created and watched. The terminal output includes commands like "shuf -i 1-10000 -n 1 > /data.txt && tail -f /dev/null". The interface also shows CPU and Memory usage statistics.

The screenshot shows a web-based interface for managing Docker sessions. At the top, there's a header bar with various icons and a URL: [https://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef\\_clauc1ggftqg00bi6sq0](https://labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clauc1ggftqg00bi6sq0). Below the header is a digital clock showing 02:19:40.

The main interface has a blue header with the session ID "clat16ef\_clauc1ggftqg00bi6sq0". It includes a "CLOSE SESSION" button and a search/filter icon. The session details show:

- IP: 192.168.0.7
- OPEN PORT: 3000
- Memory: 10.10% (403.9MiB / 3.906GiB)
- CPU: 0.50%

The "SSH" section contains a terminal window with the following content:

```
ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-lab.lan
[66a96c055c2d5ed4d967eb37d3423b3f395df40a4fld90e5802306bd39bfbe2f
[node2] (local) root@192.168.0.7 ~
$ docker run -d ubuntu bash -c "shuf -i 1-10000 -n 1 > /data.txt && tail -f /dev/null"
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e496e5c1f05f
Status: Downloaded newer image for ubuntu:latest
3fd348061ab8b603187398efd9c66b382b182e486a9b940f83befad1af701f269
[node2] (local) root@192.168.0.7 ~
$ docker exec 3fd cat /data.txt
9003
[node2] (local) root@192.168.0.7 ~
$ docker run -it ubuntu ls]
```

Below the terminal window, there are "DELETE" and "EDITOR" buttons. The session list shows two instances:

- 192.168.0.8 node1 (highlighted in grey)
- 192.168.0.7 node2

The bottom of the screen shows a Windows taskbar with various pinned icons and system status indicators like battery level (36°C), network, and date/time (16-11-2023 15:29).

This screenshot shows the same interface after a session switch. Now, "node1" (IP 192.168.0.8) is active. The terminal window displays the contents of the "/data.txt" file created earlier:

```
bin dev home lib32 libx32 mnt proc run srv tmp var
boot etc lib lib64 media opt root sbin sys usr
[node2] (local) root@192.168.0.7 ~
```

The rest of the interface remains the same, including the session details, memory/CPU usage, and the list of instances.

The bottom taskbar shows the date/time as 16-11-2023 15:30.

02:15:07

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT: 3000

Memory: 9.87% (394.7MiB / 3.906GiB) CPU: 0.85%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

DELETE EDITOR

```
bin dev home lib32 libx32 mnt proc run srv tmp var
boot etc lib lib64 media opt root sbin sys usr
[node2] (local) root@192.168.0.7 ~
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
3fd348061a8b ubuntu "bash -c 'shuf -i 1-..." 3 minutes ago Up 3 minutes
66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 8 minutes ago Up 8 minutes 0.0.0.0:3000->3000/tcp boring_agnesi
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 3fd
3fd
[node2] (local) root@192.168.0.7 ~
```

02:14:59

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT: 3000

Memory: 9.87% (394.7MiB / 3.906GiB) CPU: 0.29%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

DELETE EDITOR

```
eloquent_mcclintock
66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 8 minutes ago Up 8 minutes 0.0.0.0:3000->3000/tcp boring_agnesi
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 3fd
3fd
[node2] (local) root@192.168.0.7 ~
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 11 minutes ago Up 11 minutes 0.0.0.0:3000->3000/tcp boring_agnesi
[node2] (local) root@192.168.0.7 ~
$
```

A screenshot of a web browser displaying a Docker tutorial titled "Persisting our DB". The page shows a sidebar with "Docker 101" navigation and a main content area with three numbered steps. Step 1: "Create a volume by using the `docker volume create` command." Step 2: "Start the todo container, but add the `-v` flag to specify a volume mount. We will use the named volume and mount it to `/etc/todos`, which will capture all files created at the path." Step 3: "Once the container starts up, open the app and add a few items to your todo list." Below the steps is a screenshot of a "New Item" dialog box showing three todo items: "Finish the workshop", "Do amazing things!", and "Be awesome!". The browser's address bar shows a non-secure URL.

A screenshot of a web browser displaying a session monitoring interface. The top bar shows the URL `labs.play-with-docker.com/p/clat16efml8g009gehr0#clat16ef_clauc1ggftqg00bi6sq0`. The main area shows session details for "clat16ef\_clauc1ggftqg00bi6sq0" with IP 192.168.0.7, port 3000, and memory usage of 9.87%. It includes an "OPEN PORT" button and an "SSH" terminal window showing a root shell on the host. On the left, there's a sidebar with "Instances" and two entries: "node1" (IP 192.168.0.8) and "node2" (IP 192.168.0.7). The bottom of the screen shows a taskbar with various icons and system status.

02:07:40

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

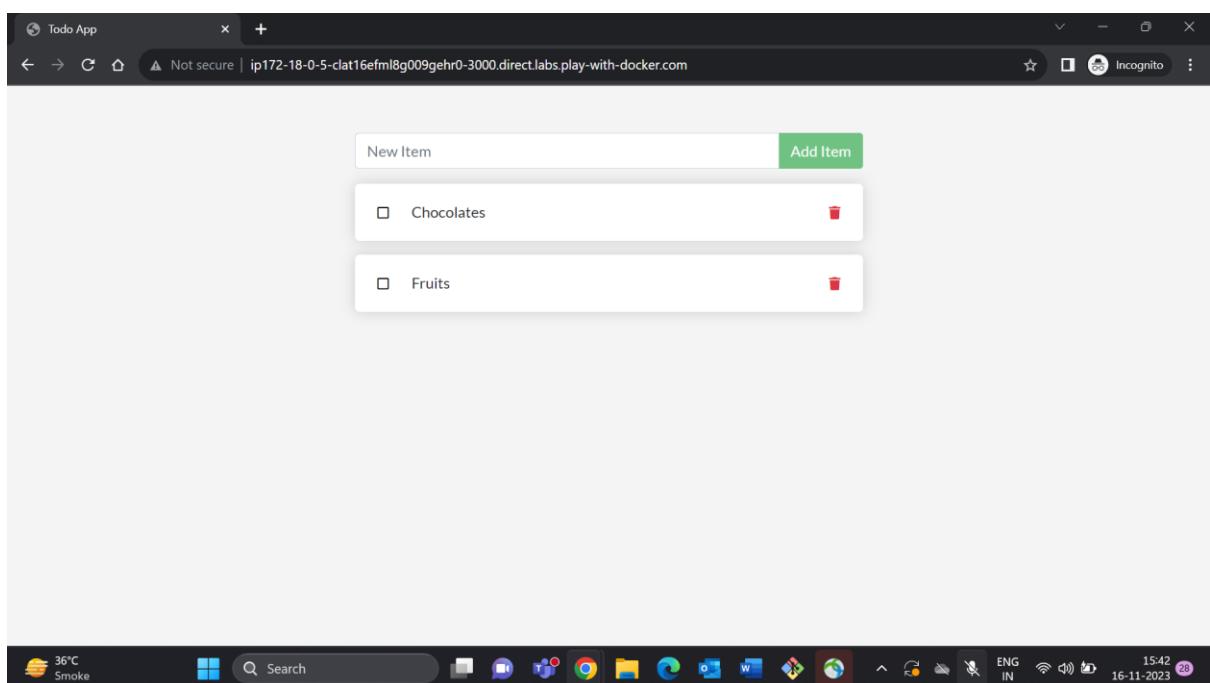
192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT 3000

Memory: 6.51% (260.3MiB / 3.906GiB) CPU: 0.35%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-with-docker.com

**[node2] (local) root@192.168.0.7 ~**  
\$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app  
91ffe3d2df765412d45711e986e10e0b027ee72b845da16f84714af9ace12  
docker: Error response from daemon: driver failed programming external connectivity on endpoint zealous\_kirch (eb81eac22e9fed62939866dd69fde663df32de069a85921d20c8e78ec04clda): Bind for 0.0.0.0:3000 failed: port is already allocated.  
[node2] (local) root@192.168.0.7 ~  
\$ docker container stop 66a  
66a  
[node2] (local) root@192.168.0.7 ~  
\$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app  
960593d8ce47783537f17cd69f22021b69679ebc0de1a5d532ea431677465cd7  
[node2] (local) root@192.168.0.7 ~  
\$



02:06:02

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT

Memory: 5.45% (218MiB / 3.906GiB) CPU: 8.51%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

DELETE EDITOR

```
docker: Error response from daemon: driver failed programming external connectivity on endpoint zealous_kirch (eb81eac22e9fed62939866dd69fde663df32de069a8592ld20c8e78ec04clda): Bind for 0.0.0.0:3000 failed: port is already allocated.
[node2] (local) root@192.168.0.7 ~
$ docker container stop 66a
66a
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
960593d8ce47783537f17cd69f22021b69679ebc0del1a5d532ea431677465cd7
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 960
960
[node2] (local) root@192.168.0.7 ~
$
```

02:05:32

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT

Memory: 5.45% (218.1MiB / 3.906GiB) CPU: 2.63%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

DELETE EDITOR

```
$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[node2] (local) root@192.168.0.7 ~
$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
91ffffe3d2d0f ruchikamoon15/101-todo-app "docker-entrypoint.s..." 3 minutes ago Created
zealous_kirch
866c3db1684 ubuntu "ls /" 13 minutes ago Exited (0) 13 minutes ago
flamboyant_meitner
66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 21 minutes ago Exited (0) 3 minutes ago
boring_agnesi
[node2] (local) root@192.168.0.7 ~
$
```

02:04:41

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT: 3000

Memory: 6.04% (241.5MiB / 3.906GiB) CPU: 35.21%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-1

**DELETE** **EDITOR**

```
$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
91ffe3d2df ruchikamoon15/101-todo-app "docker-entrypoint.s..." 3 minutes ago Created
      Ports NAMES
      91ffe3d2df zealous_kirch
      866c3dba1684 ubuntu "ls /" 13 minutes ago Exited (0) 13 minutes ago
      ago flamboyant_meitner
      66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 21 minutes ago Exited (0) 3 minutes ago
      go boring_agnesi
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
090cd166297bd0cf1490e7ee0fac4f9699f6300e978d16ff0b879ee62a6fd1
[node2] (local) root@192.168.0.7 ~
$
```

36°C Smoke Search ENG IN 15:44 16-11-2023

02:02:01

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT: 3000

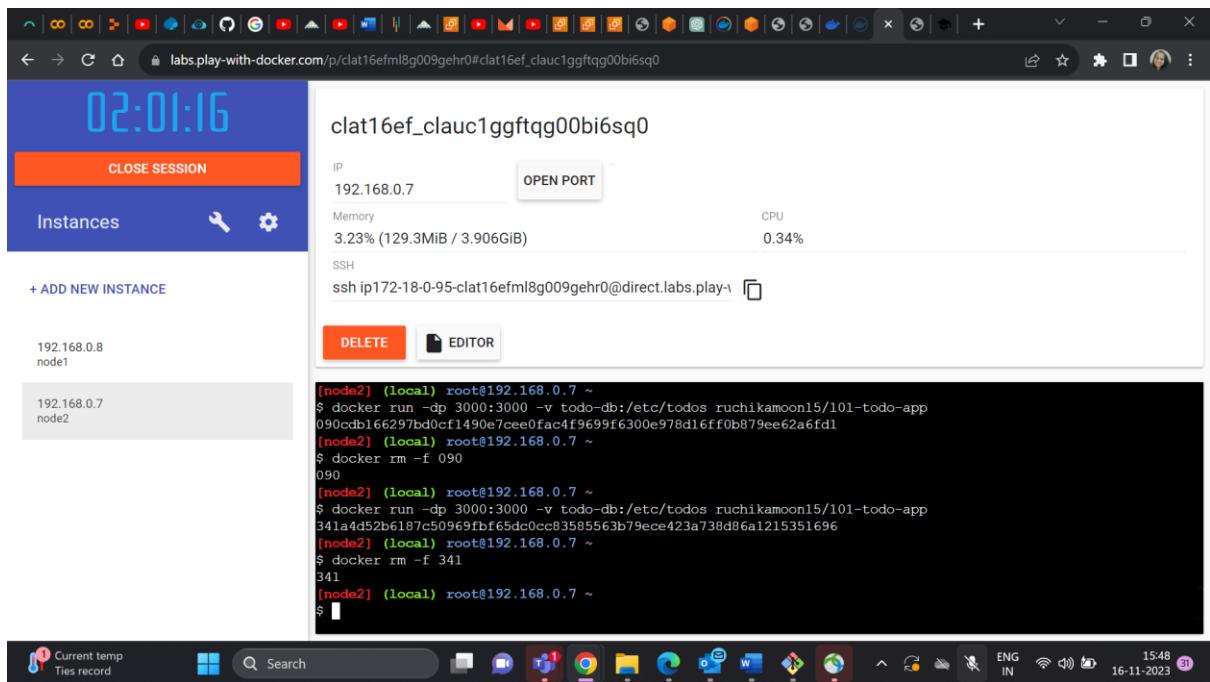
Memory: 3.83% (153.2MiB / 3.906GiB) CPU: 2.27%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-1

**DELETE** **EDITOR**

```
ago flamboyant_meitner
66a86c055c2d ruchikamoon15/101-todo-app "docker-entrypoint.s..." 21 minutes ago Exited (0) 3 minutes ago
      go boring_agnesi
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
090cd166297bd0cf1490e7ee0fac4f9699f6300e978d16ff0b879ee62a6fd1
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 090
090
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
341a4d52b6187c50969fb65dc0cc83585563b79ece423a738d06a1215351696
[node2] (local) root@192.168.0.7 ~
$
```

Top events Event brief Search ENG IN 15:47 16-11-2023



02:01:16

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

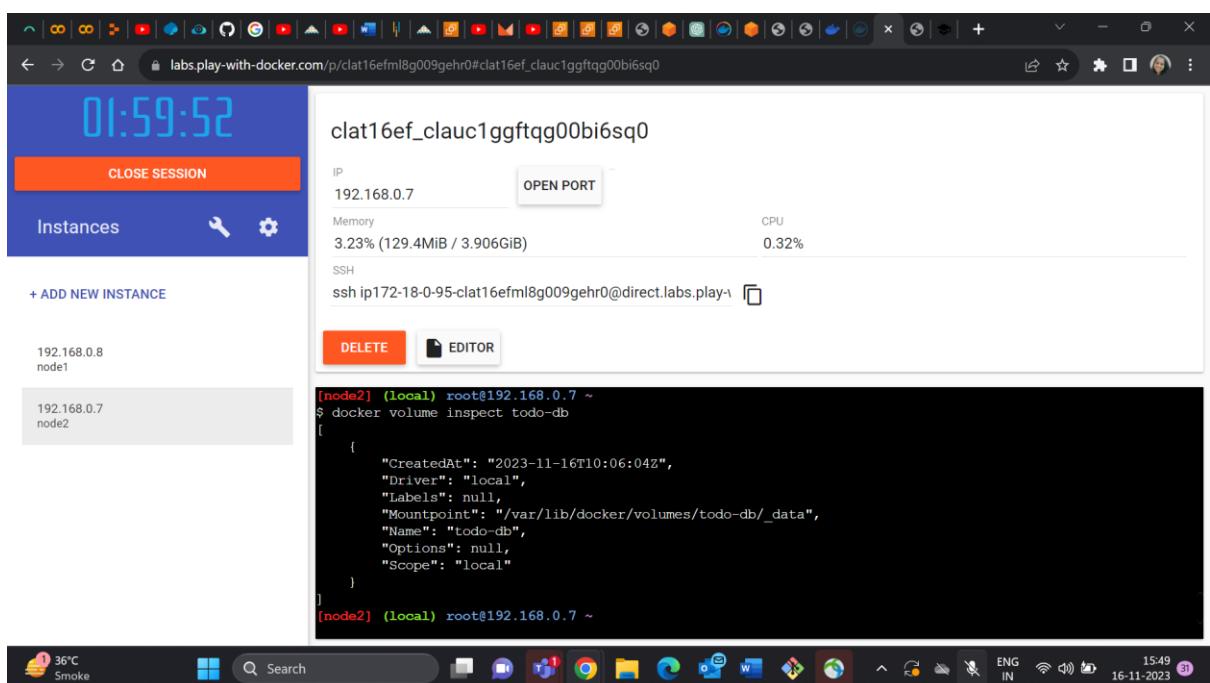
192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT

Memory: 3.23% (129.3MiB / 3.906GiB) CPU: 0.34%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

```
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
090cd166297bd0cf1490e7ce0fac4f9699f6300e978d16ff0b879ee62a6fd1
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 090
090
[node2] (local) root@192.168.0.7 ~
$ docker run -dp 3000:3000 -v todo-db:/etc/todos ruchikamoon15/101-todo-app
341a4d52b6187c50969fb65dc0cc83585563b79ece423a738d86a1215351696
[node2] (local) root@192.168.0.7 ~
$ docker rm -f 341
341
[node2] (local) root@192.168.0.7 ~
$
```



01:59:52

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8  
node1

192.168.0.7  
node2

IP: 192.168.0.7 OPEN PORT

Memory: 3.23% (129.4MiB / 3.906GiB) CPU: 0.32%

SSH: ssh ip172-18-0-95-clat16efml8g009gehr0@direct.labs.play-l

```
[node2] (local) root@192.168.0.7 ~
$ docker volume inspect todo-db
[
    {
        "CreatedAt": "2023-11-16T10:06:04Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/todo-db/_data",
        "Name": "todo-db",
        "Options": null,
        "Scope": "local"
    }
]
[node2] (local) root@192.168.0.7 ~
```