

OPTIMISING DOCKERFILE WITH ONBUILD

1. pada contoh ini kita optimasi dockerfile dengan opsi onbuild, membangun image node.js onbuild

The screenshot shows a web browser window displaying a Katacoda course page. The page title is "Optimising Dockerfile with OnBuild". The content is divided into two main sections: a text area on the left and a terminal window on the right.

Text Area (Left):

- It starts with "Step 1 of 3".
- Text: "dependent on the application which you're building, for example the application's `package.json` file."
- Text: "Below is the [Node.js OnBuild Dockerfile](#). Unlike in our previous scenario the application specific commands have been prefixed with ONBUILD."
- A code block contains the following Dockerfile content:

```
FROM node:7
RUN mkdir -p /usr/src/app
WORKDIR /usr/src/app
ONBUILD COPY package.json /usr/src/app/
ONBUILD RUN npm install
ONBUILD COPY . /usr/src/app
CMD [ "npm", "start" ]
```
- Text: "The result is that we can build this image but the application specific commands won't be executed until the built image is used as a base image. They'll then be executed as part of the base image's build."
- Text: "We'll see how this base image looks in the next step."
- A "CONTINUE" button is at the bottom.

Terminal Window (Right):

- The terminal shows the command `cat Dockerfile` being executed.
- The output of the command is:

```
FROM node:7-onbuild
EXPOSE 3000
```

2. kita bisa cek repo onbuild di github

The screenshot shows a web browser window displaying the GitHub repository page for `docker-library/node`. The page is titled "node/Dockerfile at 70741d88bf688389bfac7b147573f3b761f9ede9 · docker-library/node · GitHub".

Repository Information:

- Repository name: `docker-library / node`
- Forked from: `nodejs/node-v0.x-archive`
- Watch: 11, Star: 66, Fork: 7,943

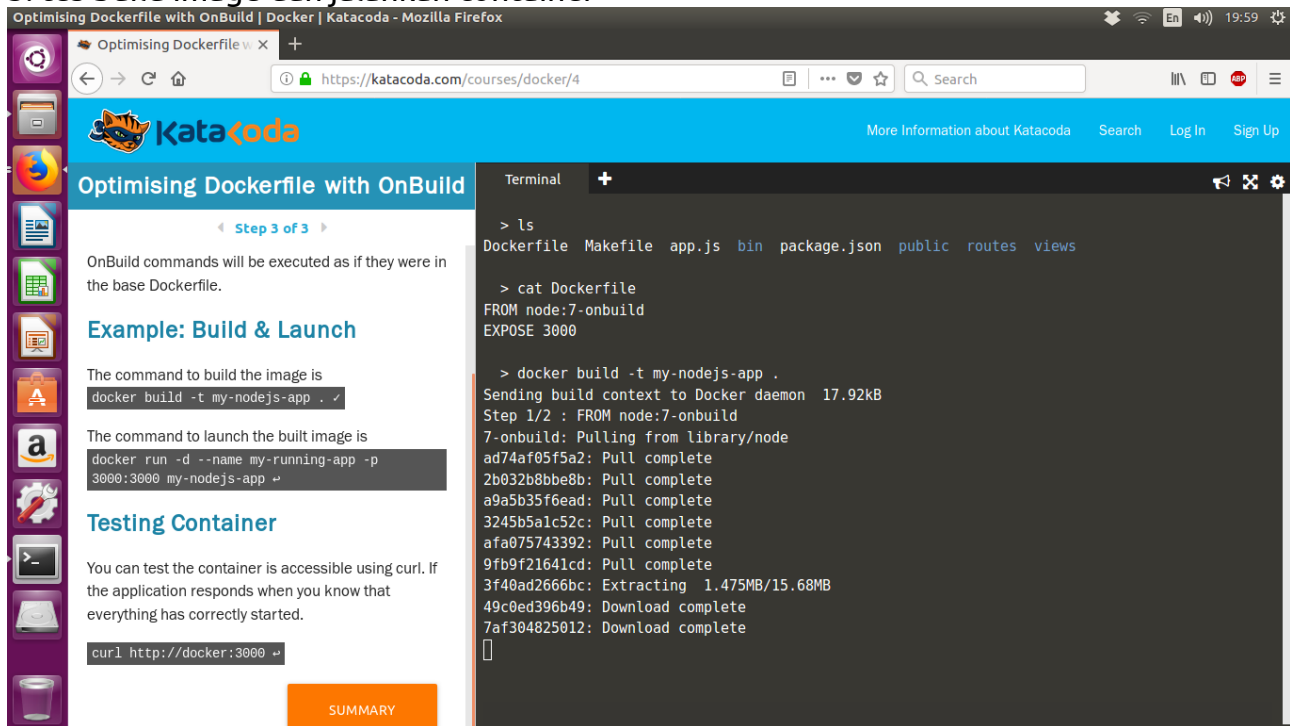
Code View:

- The "Code" tab is selected.
- The file path is `node / 0.10 / onbuild / Dockerfile`.
- The commit hash is `70741d88bf`.
- The commit message is "Run update.sh, bump 0.10 and npm" by `yosifkit`.
- The commit date is "70741d8 on Dec 19, 2014".
- There are 3 contributors.

Dockerfile Content:

```
1 FROM node:0.10.34
2
3 RUN mkdir -p /usr/src/app
4 WORKDIR /usr/src/app
5
6 ONBUILD COPY package.json /usr/src/app/
7 ONBUILD RUN npm install
8 ONBUILD COPY . /usr/src/app
9
10 CMD [ "npm", "start" ]
```

3. tes build image dan jalankan container



The screenshot shows a web browser window with the URL <https://katacoda.com/courses/docker/4>. The page title is "Optimising Dockerfile with OnBuild | Docker | Katacoda - Mozilla Firefox". The page content is titled "Optimising Dockerfile with OnBuild" and is part of a "Step 3 of 3" tutorial. The tutorial explains that OnBuild commands will be executed as if they were in the base Dockerfile. It provides an example of building and launching a container, showing the commands and their output in a terminal window.

Example: Build & Launch

The command to build the image is

```
docker build -t my-nodejs-app .
```

The command to launch the built image is

```
docker run -d --name my-running-app -p 3000:3000 my-nodejs-app
```

Testing Container

You can test the container is accessible using curl. If the application responds when you know that everything has correctly started.

```
curl http://docker:3000
```

Terminal Output:

```
> ls
Dockerfile  Makefile  app.js  bin  package.json  public  routes  views

> cat Dockerfile
FROM node:7-onbuild
EXPOSE 3000

> docker build -t my-nodejs-app .
Sending build context to Docker daemon 17.92kB
Step 1/2 : FROM node:7-onbuild
7-onbuild: Pulling from library/node
ad74af05f5a2: Pull complete
2b032b8bbe8b: Pull complete
a9a5b35f6ead: Pull complete
3245b5a1c52c: Pull complete
afa075743392: Pull complete
9fb9f21641cd: Pull complete
3f40ad2666bc: Extracting 1.475MB/15.68MB
49c0ed396b49: Download complete
7af304825012: Download complete
[]
```

dok 03/05/2018 padliyulian@gmail.com