Node.js

In this lesson, you will be provided with the basic application image for Node.js.

WE'LL COVER THE FOLLOWING

Result

The resulting image is published as *learnbook/node-server*. You can run a container from it with the following command:

```
docker run --rm -it -p 8087:80 learnbook/node-server
```

Then point your browser to `http://localhost:8087`

Files

> You can find these files in the *code/common-development-profiles/demos/nodejs* folder.

A static HTML file to be served:

www/index.htm

```
<h1>Hello !</h1>
<p>This page is served by Node.JS</p>
<p>Try our <a href="/v1/square/4">multiply API</a>.</p>
```

Define and run an HTTP server, a static files server and a REST API:

index.js

```
const HTTP_PORT = 80;

const cors = require('cors');
const express = require('express');
const path = require('path');

var sourcesDirectory = path.resolve(__dirname, 'www');
var app = express();
app.use(cors());

// API
app.get('/v1/square/:value', function (req, res) {
    const value = req.params.value;
    const square = Math.pow(value, 2);
    res.send({
        value,
        square
    });
});

// Static files
app.use(express.static(sourcesDirectory, {
    index: 'index.htm',
    extensions: ['htm']
}));

var server = require('http').createServer(app);
server.listen(HTTP_PORT);

console.log(`Listening on http://localhost:${HTTP_PORT}`);
```

List dependencies that will need to be restored during the build:

package.json

```
{
  "name": "node-server",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "dependencies": {
    "async": "^2.4.1",
    "cors": "^2.8.3",
    "express": "^4.15.2"
  }
}
```

Docker image definition:

Dockerfile

```
FROM node:10-alpine

# Create app directory
RUN mkdir -p /usr/src/app
WORKDIR /usr/src/app

# Install app dependencies
COPY package.json /usr/src/app/
RUN npm install

WORKDIR /usr/src/app

# Bundle app source
COPY . /usr/src/app/

EXPOSE 80

CMD ["npm", "start"]
```

Let’s look at another image in the next lesson.