Node js project (Docker & k8s)

```
.
└── project-code
    ├── app.js
    ├── Dockerfile
    ├── k8-deploy.yml
    ├── k8-pvc.yml
    ├── k8-pv.yml
    ├── k8-svc.yml
    ├── package.json
    └── test
        └── app.test.js
```

```
adham@master:/adhamjenkins$ sudo mkdir nodejs-project
[sudo] password for adham:
adham@master:/adhamjenkins$ cd nodejs-project
adham@master:/adhamjenkins/nodejs-project$ sudo git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /adhamjenkins/nodejs-project/.git/
adham@master:/adhamjenkins/nodejs-project$
```

```javascript
const express = require('express');
const app = express();
const port = 3000;

app.get('/', (req, res) => {
    res.send('Hello I am Adham and this is nodejs website');
});

app.listen(port, () => {
    console.log(`App running on port ${port}`);
});

module.exports = app;
```

```
adham@master:/adhamjenkins/nodejs-project$ sudo mkdir project-code
adham@master:/adhamjenkins/nodejs-project$ cd project-code
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim app.js
```

```json
{
    "name": "my-node-app",
    "version": "1.0.0",
    "scripts": {
        "start": "node app.js",
        "test": "mocha"
    },
    "dependencies": {
        "express": "^4.17.1"
    },
    "devDependencies": {
        "chai": "^4.3.4",
        "chai-http": "^4.3.0",
        "mocha": "^10.2.0",
        "supertest": "^6.3.1"
    },
    "type": "module"
}
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim package.json
```

```javascript
import chai from 'chai';
import chaiHttp from 'chai-http';
import app from '/app/app.js'; // Import the Express app

chai.use(chaiHttp);
const { expect } = chai;

describe('Express App Tests', () => {
    it('should return "Hello I am Adham and this is nodejs website" on GET /', (done) => {
        chai.request(app)
            .get('/')
            .end((err, res) => {
                expect(res).to.have.status(200);
                expect(res.text).to.equal('Hello I am Adham and this is nodejs website');
                done();
            });
    });
});
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo mkdir test
adham@master:/adhamjenkins/nodejs-project/project-code$ cd test
adham@master:/adhamjenkins/nodejs-project/project-code/test$
adham@master:/adhamjenkins/nodejs-project/project-code/test$ sudo vim app.test.js
```

```dockerfile
FROM node:16-alpine
WORKDIR /app


COPY ./project-code  /app


RUN npm install
RUN npm install mocha supertest chai --save-dev


CMD [ "npm", "start"]


EXPOSE 3000
#same port inside app.js
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim Dockerfile
```

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: node-pv
spec:
  capacity:
    storage: 1G
  accessModes:
    - ReadWriteMany
  hostPath:
    path: "/adhamjenkins"
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim k8-pv.yml
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: node-pvc
spec:
  volumeName: node-pv
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 1G
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim k8-pvc.yml
```

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: node-deploy
  labels:
    node: js
spec:
  replicas: 3
  selector:
    matchLabels:
      node: js
  template:
    metadata:
      name: pod-node
      labels:
        node: js
    spec:
      containers:
        - name: cont-node
          image: adhmabdein/mynodeimg
          ports:
            - containerPort: 3000 # same port inside app.js
          volumeMounts:
            - name: pod-volume
              mountPath: /app/data
      volumes:
        - name: pod-volume
          persistentVolumeClaim:
            claimName: node-pvc
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim k8-deploy.yml
```

```yaml
apiVersion: v1
kind: Service
metadata:
  name: node-svc
spec:
  type: LoadBalancer
  selector:
    node: js
  ports:
    - protocol: TCP
      port: 3000
      nodePort: 30007
      targetPort: 3000
```

```
adham@master:/adhamjenkins/nodejs-project/project-code$ sudo vim k8-svc.yml
```

```
adham@master:/adhamjenkins/nodejs-project$ sudo git add .
adham@master:/adhamjenkins/nodejs-project$
adham@master:/adhamjenkins/nodejs-project$ sudo git commit -m "initial commit"
[master (root-commit) 8ab345b] initial commit
 Committer: root <root@master.localdomain>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 8 files changed, 116 insertions(+)
 create mode 100644 project-code/Dockerfile
 create mode 100644 project-code/app.js
 create mode 100644 project-code/k8-deploy.yml
 create mode 100644 project-code/k8-pv.yml
 create mode 100644 project-code/k8-pvc.yml
 create mode 100644 project-code/k8-svc.yml
 create mode 100644 project-code/package.json
 create mode 100644 project-code/test/app.test.js
adham@master:/adhamjenkins/nodejs-project$
```

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
Import a repository.

Required fields are marked with an asterisk (*).

Owner *   **Repository name ***

🌐 AdhmAbdein ▾ / nodejs-project

✅ **nodejs-project is available.**

Great repository names are short and memorable. Need inspiration? How about **super-duper-rotary-phone** ?

Description (optional)

⦿ 📖 **Public**
Anyone on the internet can see this repository. You choose who can commit.

◯ 🔒 **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: **None** ▾

Choose which files not to track from a list of templates. Learn more about ignoring files.

```
adham@master:/adhamjenkins/nodejs-project$ sudo git remote add origin https://github.com/AdhmAbdein/nodejs-project.git
```

```
adham@master:/adhamjenkins/nodejs-project$ sudo git push -u origin master
Username for 'https://github.com': AdhmAbdein
Password for 'https://AdhmAbdein@github.com':
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 2 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (12/12), 1.86 KiB | 951.00 KiB/s, done.
Total 12 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AdhmAbdein/nodejs-project.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
adham@master:/adhamjenkins/nodejs-project$
adham@master:/adhamjenkins/nodejs-project$
```
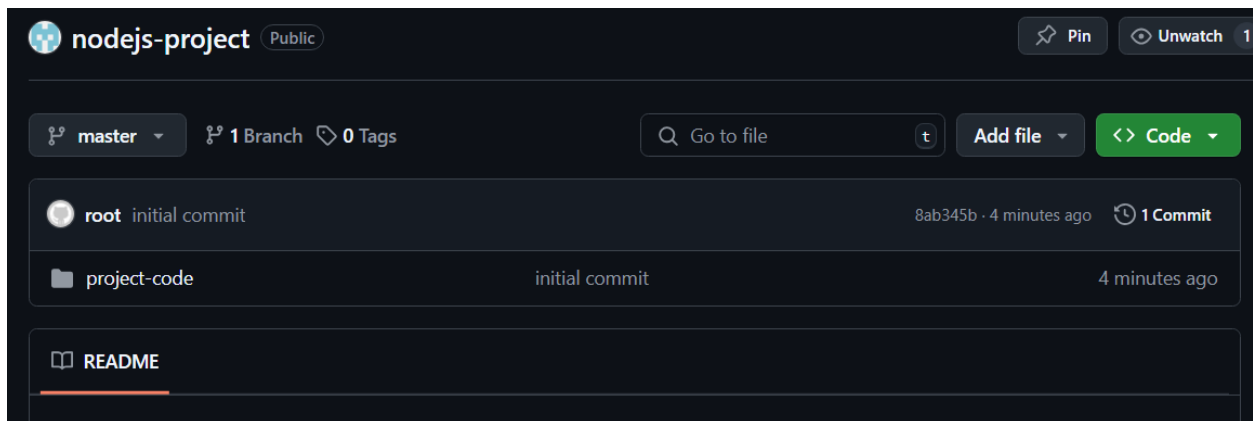
```
pipeline{
    agent {label 'node1'}

    environment {
        docker_img_name = 'adhmabdein/mynodeimg'
        //test_port = '3001' // Use a different port for testing

    }
    stages{
        stage ('pull code from github'){
            steps{
                script{
                    git "https://github.com/AdhmAbdein/nodejs-project.git"
                }
            }
        }
        stage('Login to docker hub'){
            steps{
                script{
                    withCredentials([usernamePassword(credentialsId:'docker_hub' , usernameVariable:'d_hub_usr' , passwordVariable:'d_hub_pass')]){
                        sh 'docker login -u ${d_hub_usr} -p ${d_hub_pass}'
                    }
                }
            }

        }
        stage('CI - Build docker image'){
            steps{
                script{
                    sh 'docker build -t ${docker_img_name} -f project-code/Dockerfile . '
                }
            }
        }
        stage('CI - Test inside container(temperory) (npm test)'){
            steps{
                script{
                    sh 'docker run --rm -d ${docker_img_name} /bin/sh -c " ls -R /app && npm test"'
                    //docker run --rm -it adhmabdein/mynodeimg /bin/sh -c "ls -R /app && npm test"
                    //sh "docker run --rm -e PORT=${test_port} -p ${test_port}:${test_port} ${docker_img_name} npm test"

                }
            }
        }
        stage('push image to docker hub'){
            steps{
                script{
                    sh 'docker push ${docker_img_name}'
                }
            }
        }
        stage('CD - Deploy in k8s'){
            steps{
                script{
                    dir('project-code'){
                        sh 'kubectl apply -f k8-pv.yml'
                        sh 'kubectl apply -f k8-pvc.yml'
                        sh 'kubectl apply -f k8-deploy.yml'
                        sh 'kubectl apply -f k8-svc.yml'


                    }
                }
            }
        }
    }
    post {
        always{
            archiveArtifacts artifacts : '**/test-result.xml' , allowEmptyArchive: true
            //**/: This means "any directory" (recursively)
        }
    }
}
```

adhmabdein/mynodeimg

By [adhmabdein](#) · Updated 5 minutes ago
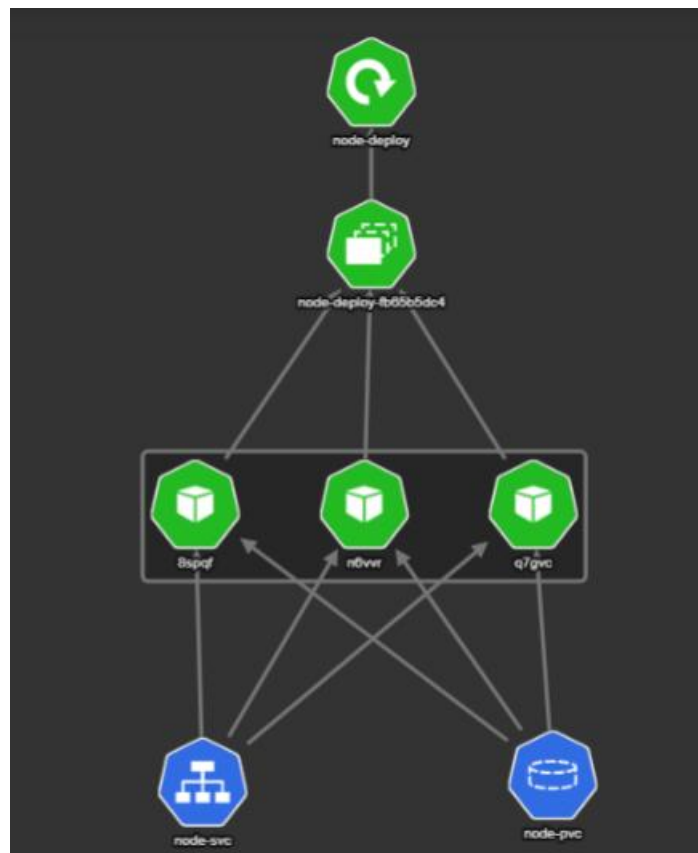
---

nodejs-pipeline   55 sec   #26   3 min 30 sec #25   45 sec

← → C   ⚠ Not secure   192.168.▮▮▮:30007

⊞ | 🔴 Submit to Reddit   📌 (1) Pinterest   🎨 DeviantArt - Discov...   Q Home   A Amino Apps

Hello I am Adham and this is nodejs website

```
adham@node1:~$ kubectl get pod
NAME                             READY   STATUS    RESTARTS        AGE
anti-54c6bff586-76wnf            1/1     Running   5 (5h1m ago)    4d3h
anti-54c6bff586-dlzht            1/1     Running   5 (5h1m ago)    4d3h
anti-54c6bff586-jqt5b            1/1     Running   5 (5h1m ago)    4d3h
anti-54c6bff586-xbz9z            1/1     Running   5 (5h1m ago)    4d3h
appv1-5c8d84f987-bdfkb           1/1     Running   5 (5h1m ago)    4d2h
appv1-5c8d84f987-glnc4           1/1     Running   5 (5h1m ago)    4d2h
appv1-5c8d84f987-hw98j           1/1     Running   5 (5h1m ago)    4d2h
appv2-598b497c44-mp4rd           1/1     Running   5 (5h1m ago)    4d1h
auto-deploy-6bcdb5d77b-264k7     1/1     Running   7 (5h1m ago)    5d2h
auto-deploy-6bcdb5d77b-2wkj5     1/1     Running   7 (5h1m ago)    5d2h
auto-deploy-6bcdb5d77b-cx6zh     1/1     Running   7 (5h1m ago)    5d2h
auto-deploy-6bcdb5d77b-w6bhs     1/1     Running   7 (5h1m ago)    5d2h
dep-ing-79bf94fd84-7cpqv         1/1     Running   7 (5h1m ago)    5d2h
dep-ing-79bf94fd84-kllfc         1/1     Running   7 (5h1m ago)    5d2h
dep-ing-79bf94fd84-m8ngw         1/1     Running   7 (5h1m ago)    5d2h
deploy-web-6498fd9b8c-k952b      1/1     Running   7 (5h1m ago)    5d2h
deploy-web-6498fd9b8c-qpk22      1/1     Running   7 (5h1m ago)    5d2h
deploy-web-6498fd9b8c-rd7lp      1/1     Running   7 (5h1m ago)    5d2h
kubeview-78dd597bd8-gswkz        1/1     Running   7 (5h1m ago)    5d2h
node-deploy-fb65b5dc4-8spqf      1/1     Running   0               2m1s
node-deploy-fb65b5dc4-n6vvr      1/1     Running   0               2m1s
node-deploy-fb65b5dc4-q7gvc      1/1     Running   0               2m1s
podaff                           1/1     Running   7 (5h1m ago)    5d1h
podselector                      1/1     Running   5 (5h1m ago)    4d
rollout-deploy-7647484798-9stjn  1/1     Running   7 (5h1m ago)    5d2h
rollout-deploy-7647484798-fqtkd  1/1     Running   7 (5h1m ago)    5d2h
rollout-deploy-7647484798-tp6vl  1/1     Running   7 (5h1m ago)    5d2h
adham@node1:~$
```

```
adham@node1:~$ kubectl get svc
NAME          TYPE           CLUSTER-IP    EXTERNAL-IP    PORT(S)           AGE
kubernetes    ClusterIP      10.96         <none>         443/TCP           10d
kubeview      LoadBalancer   10.96         <pending>      80:31262/TCP      10d
node-svc      LoadBalancer   10.100        <pending>      3000:30007/TCP    6m34s
service-ing   NodePort       10.99         <none>         8889:31000/TCP    5d3h
service2      NodePort       10.107        <none>         8088:30001/TCP    9d
svc           NodePort       10.98         <none>         6666:30003/TCP    4d1h
web-service   NodePort       10.108        <none>         4444:30000/TCP    10d
adham@node1:~$ kubectl get pvc
NAME       STATUS   VOLUME    CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
node-pvc   Bound    node-pv   1G         RWX                           <unset>                 12m
pvc        Bound    pv        1G         RWO                           <unset>                 6d5h
adham@node1:~$ kubectl get pv
NAME      CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM              STORAGECLASS   VOLUMEATTRIB
UTESCLASS   REASON   AGE
node-pv   1G         RWX            Retain           Bound    default/node-pvc                  <unset>
                     12m
pv        1G         RWO            Retain           Bound    default/pvc                       <unset>
                     6d5h
adham@node1:~$ kubectl get deploy
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
anti           4/4     4            4           4d3h
appv1          3/3     3            3           4d2h
appv2          1/1     1            1           4d1h
auto-deploy    4/4     4            4           9d
dep-ing        3/3     3            3           5d3h
deploy-web     3/3     3            3           9d
kubeview       1/1     1            1           10d
node-deploy    3/3     3            3           6m50s
rollout-deploy 3/3     3            3           8d
```

Activate Windows