

Node.Js Deployment Setup:

Node.js is a JavaScript runtime for server-side programming. It allows developers to create scalable backend functionality using JavaScript, a language many are already familiar with from browser-based web development

Installing Node. js

```
# sudo apt update
```

```
# sudo apt install nodejs
```

```
# sudo apt install npm
```

checking node version

```
# node -v
```

Installing Node Version Manager NVM

```
# curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh
```

```
# curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
```

```
# source ~/.bashrc
```

list all node version

```
# nvm list-remote
```

Now you can install any of this node version using this command

```
# nvm install v16.13.1
```

You can see all installed node version using

```
# nvm list
```

Now you installed so many version of node then you have to switch to different version then

```
# nvm use v14.10.0
```

Configure MongoDB for database connection

Create Admin User

```
# mongo
```

```
# use DBname
```

```
# db.createUser({user:"username",pwd:"password",roles:["readWrite","dbAdmin","dbOwner"]})
```

```
# db.auth('username','passwd')
```

Deploying Node Project

Clone The Project To Your Working Path

```
# git clone
```

```
# cd /home/node
```

Node Project Need .env file or need config.json file to add credential

Create All Credential And Add In config.json file or add .env file Depends On Project

```
# 1. mongodb : DataBase, User, Password,
```

```
# 2. redis : DB=NO.
```

```
# 3. rebbitMQ : User, Password, VirtualHost
```

Install Node_modules

```
# npm install
```

```
# node start app.js/index.js
```

Using Pm2 :-

If you have to use different node version then use this command,

```
# npm install pm2@latest -g
```

```
# pm2 install pm2-logrotate
```

Start Pm2 for project

```
# pm2 start index.js
```

List all pm2 :-

```
# pm2 list/ls
```

Managing Pm2 :-

```
# pm2 restart app_name/id
```

```
# pm2 reload app_name/id
```

```
# pm2 stop app_name/id
```

```
# pm2 delete app_name/id
```

Display pm2 Logs :-

```
# pm2 logs
```

CI/CD For Node Deployment

deploy_dev:

stage: deploy

tags:

- dev - gitlab-runner tag

environment:

name: development

url: https://dev.artoon.com:3000/ = Node-port

before_script:

- eval export PATH=/root/.nvm/versions/node/v15.10.0/bin:\${PATH}

script:

- cd /home/node/project - project path
- git pull http://\$usr:\$pwd@gitlab.artoon.in/project/project-demo.git -f development
- npm install --cache
- pm2 restart index = pm2 app name

when: manual

only:

- development