Co1_Notes

16 July 2025 09:57 AM

Devops=Develop,meynt(plan,code test,build) + Operations(relesase,deploy,monitor ,operate)

Collaboration between development and opearations team
Automation of softwate delivery process
Automation Integration(ci) and continuous delivery(cd)
Immproved quality and speed

Contiunos intergration

Automantically build and test when there is change in code
Ensure bug free code
Ensures higher code quality and reduces integration issues
Packaging the application(war/jar/docker image)

Continuous deployment

-Automationcally deploy for production

Life cycle
Code->build->test->deployment

Tools
Code-Git-,github,gitlab ,bitbucket
Build-war/jar/docker image
Test-Pytest , Junittest,sonarqube(pluggins to be installed in jenkins)
Deploy apache /tomcat,docker ,kubernetes

Git-
Version control source code repository

```
git init                    # Initialize a new Git repository
git add .                   # Stage all files for commit
git commit -m "Initial commit"     # Commit with a message
git branch -M main              # Rename current branch to 'main'
git remote add origin <repo-URL> && git push -u origin main   # Add remote repo and push
```

Cogiure git hub accounft

```
git config --global user.name "Your Name"
git config --global user.email "your.email@example.com"
```

configure GitHub Account:

```
git config --global user.name "srithars"
git config --global user.email "sss.srithar@gmail.com"
```

Git Operations:

```
git init    # git initialization

git add .   # Stage all files

git commit -m "first commit"  # Stage all files

git branch -M main       # create a branch

git remote add origin https://github.com/srithars/cicd_repo.git

git push -u origin main
```

19 July 2025 10:54 AM

La1 Branching

Practical 1:

Create a GitHub repository named webAppProject. In the main branch, add index.html. Create three branches: adminfeature, usefeature, and managerfeature. In each, add the respective HTML file admin.html, user.html, and manager.html. Merge all branches into main and push the final result to GitHub.

Pre requirement
----
1. webAppProject local Directory
2. webAppProject remore repositoty

Steps:
1. create main branch and push index.html
2. create 3 branches named with adminfeature, userfeature and manager feature push respective html files into branches
3. merge adminfeature, userfeature and manager branch with main branch.

Step1:

```
git init
git add index.html
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/url
git push -u origin main
```

Step2

```
git branch adminfeature
git checkout adminfeature
git add admin.html
git commit -m "admin page commited in branch"
git push -u origin adminfeature

git checkout main
git branch userfeature
git checkout userfeature
git add user.html
git commit -m "user page commited in branch"
git push -u origin userfeature

git checkout main
git branch managerfeature
git checkout managerfeature
git add manager.html
git commit -m "manager page commited in branch"
git push -u origin managerfeature
```

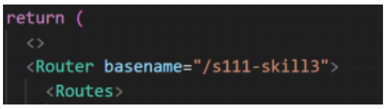
step3:

```
git checkout main
git merge adminfeature -m "adminfeature merged with main"
git merge userfeature -m "userfeature merged with main"
git merge managerfeature -m "managerfeature merged with main"
git push -u origin main
```

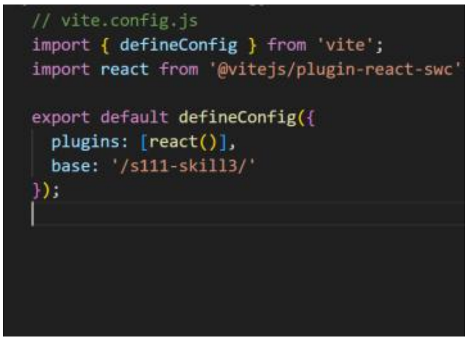
finally clone project into your local directory

```
git clone https://github.com/srithar/webAppProject.git
```

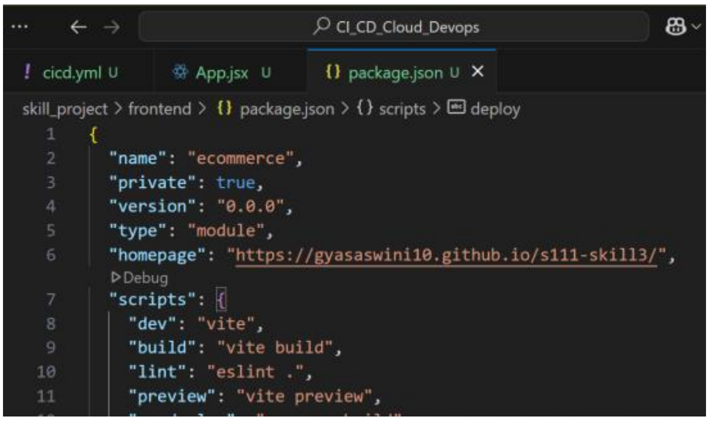
Npm install =>npm i. vite



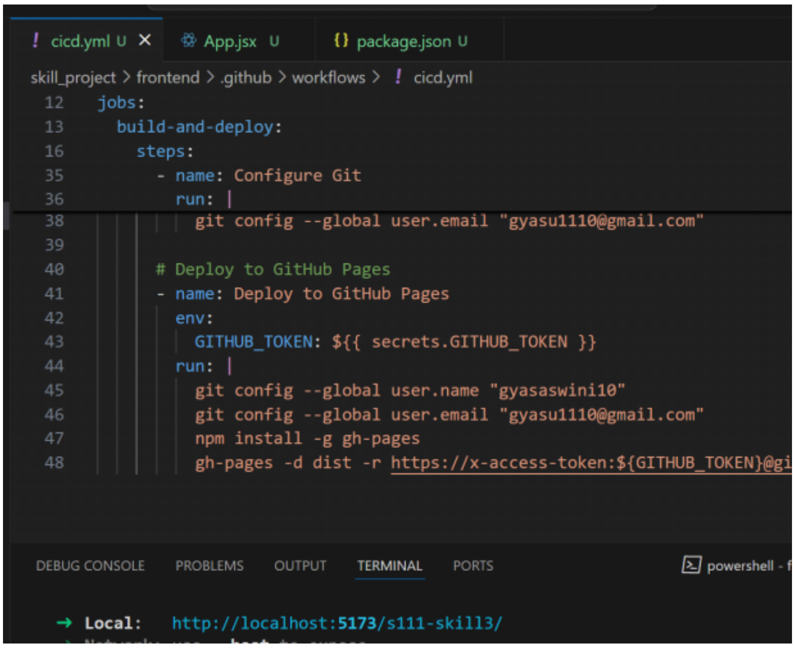
```
return (
  <>
  <Router basename="/s111-skill3">
    <Routes>
```



```
// vite.config.js
import { defineConfig } from 'vite';
import react from '@vitejs/plugin-react-swc'

export default defineConfig({
  plugins: [react()],
  base: '/s111-skill3/'
});
```

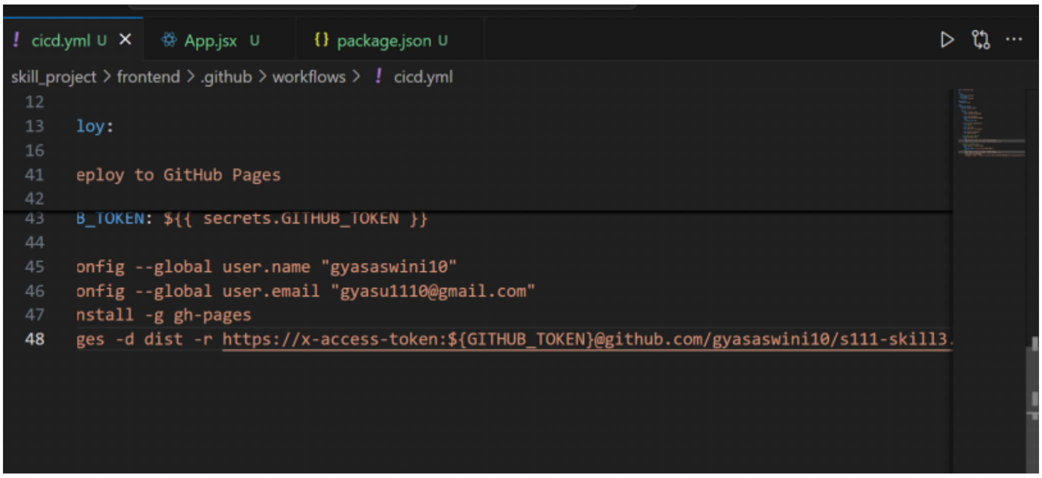


```
{
  "name": "ecommerce",
  "private": true,
  "version": "0.0.0",
  "type": "module",
  "homepage": "https://gyasaswini10.github.io/s111-skill3/",
  "scripts": {
    "dev": "vite",
    "build": "vite build",
    "lint": "eslint .",
    "preview": "vite preview",
```



```yaml
jobs:
  build-and-deploy:
    steps:
      - name: Configure Git
        run: |
          git config --global user.email "gyasu1110@gmail.com"

      # Deploy to GitHub Pages
      - name: Deploy to GitHub Pages
        env:
          GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
        run: |
          git config --global user.name "gyasaswini10"
          git config --global user.email "gyasu1110@gmail.com"
          npm install -g gh-pages
          gh-pages -d dist -r https://x-access-token:${GITHUB_TOKEN}@gi
```

```
➜  Local:   http://localhost:5173/s111-skill3/
```



```
loy:

eploy to GitHub Pages

B_TOKEN: ${{ secrets.GITHUB_TOKEN }}

onfig --global user.name "gyasaswini10"
onfig --global user.email "gyasu1110@gmail.com"
nstall -g gh-pages
ges -d dist -r https://x-access-token:${GITHUB_TOKEN}@github.com/gyasaswini10/s111-skill3.
```

To   remove remote origin git hub
Git remote remove origin

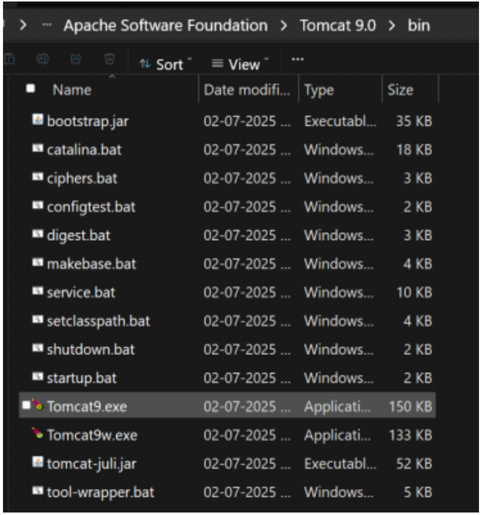
Action genral

SKILL 5- jenkins frontend

```
server.port=8081
```

git push -u origin master
TOMCAT->START

C DRIVE-> PROGRAM FILES->
APPACHE TOMCAT



Name	Date modifi...	Type	Size
bootstrap.jar	02-07-2025 ...	Executabl...	35 KB
catalina.bat	02-07-2025 ...	Windows...	18 KB
ciphers.bat	02-07-2025 ...	Windows...	3 KB
configtest.bat	02-07-2025 ...	Windows...	2 KB
digest.bat	02-07-2025 ...	Windows...	3 KB
makebase.bat	02-07-2025 ...	Windows...	4 KB
service.bat	02-07-2025 ...	Windows...	10 KB
setclasspath.bat	02-07-2025 ...	Windows...	4 KB
shutdown.bat	02-07-2025 ...	Windows...	2 KB
startup.bat	02-07-2025 ...	Windows...	2 KB
Tomcat9.exe	02-07-2025 ...	Applicati...	150 KB
Tomcat9w.exe	02-07-2025 ...	Applicati...	133 KB
tomcat-juli.jar	02-07-2025 ...	Executabl...	52 KB
tool-wrapper.bat	02-07-2025 ...	Windows...	5 KB

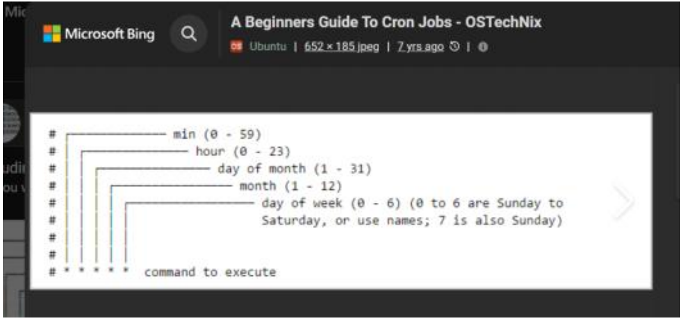
DOUBLE CLICK IT

Juenkins +new intrm

Free style

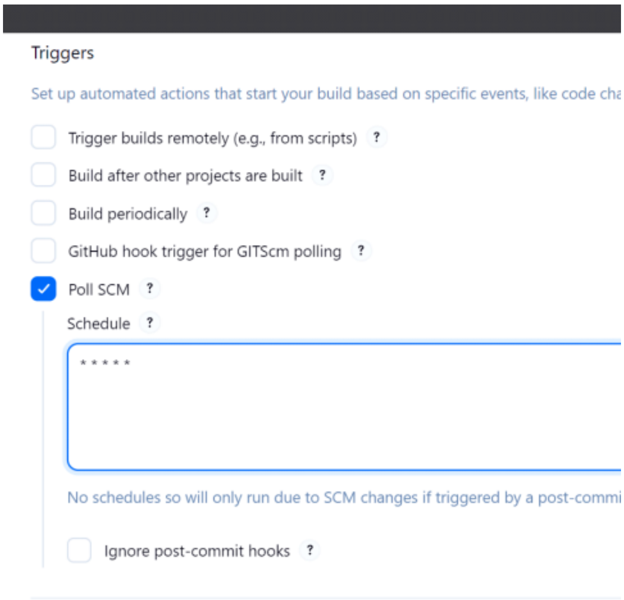
Source code managemnet
Git

Triggers
Poll scm



Microsoft Bing
A Beginners Guide To Cron Jobs - OSTechNix
Ubuntu | 652 × 185 jpeg | 7 yrs ago

```
# ┌───────────── min (0 - 59)
# │ ┌────────────── hour (0 - 23)
# │ │ ┌─────────────── day of month (1 - 31)
# │ │ │ ┌──────────────── month (1 - 12)
# │ │ │ │ ┌───────────────── day of week (0 - 6) (0 to 6 are Sunday to
# │ │ │ │ │                  Saturday, or use names; 7 is also Sunday)
# │ │ │ │ │
# │ │ │ │ │
# * * * * *  command to execute
```



Triggers

Set up automated actions that start your build based on specific events, like code cha

□ Trigger builds remotely (e.g., from scripts) ?

□ Build after other projects are built ?

□ Build periodically ?

□ GitHub hook trigger for GITScm polling ?

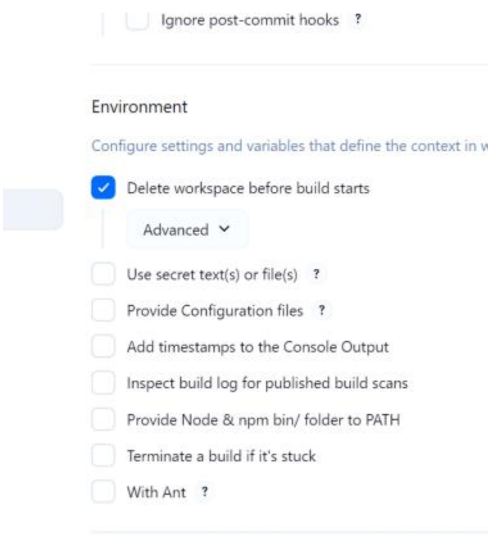
☑ Poll SCM ?

Schedule ?

```
* * * * *
```

No schedules so will only run due to SCM changes if triggered by a post-commi

□ Ignore post-commit hooks ?



□ Ignore post-commit hooks ?

Environment

Configure settings and variables that define the context in w

☑ Delete workspace before build starts

Advanced

□ Use secret text(s) or file(s) ?

□ Provide Configuration files ?

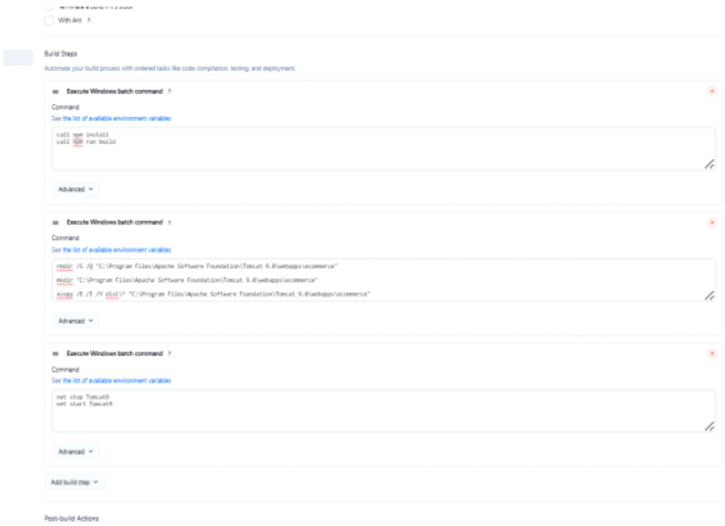
□ Add timestamps to the Console Output

□ Inspect build log for published build scans

□ Provide Node & npm bin/ folder to PATH

□ Terminate a build if it's stuck

□ With Ant ?



□ Terminate a build if it's stuck

□ With Ant ?

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Execute Windows batch command ?

Command

See the list of available environment variables

```
call npm install
call npm run build
```

Advanced

Execute Windows batch command ?

Command

See the list of available environment variables

```
rmdir /S /Q "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"
mkdir "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"
xcopy /E /I /Y dist\* "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"
```

Advanced

Execute Windows batch command ?

Command

See the list of available environment variables

```
net stop Tomcat9
net start Tomcat9
```

Advanced

Add build step

Post-build Actions

5.1 Execute Windows batch command
call npm install
call npm run build

FREE STYLE PROJECT

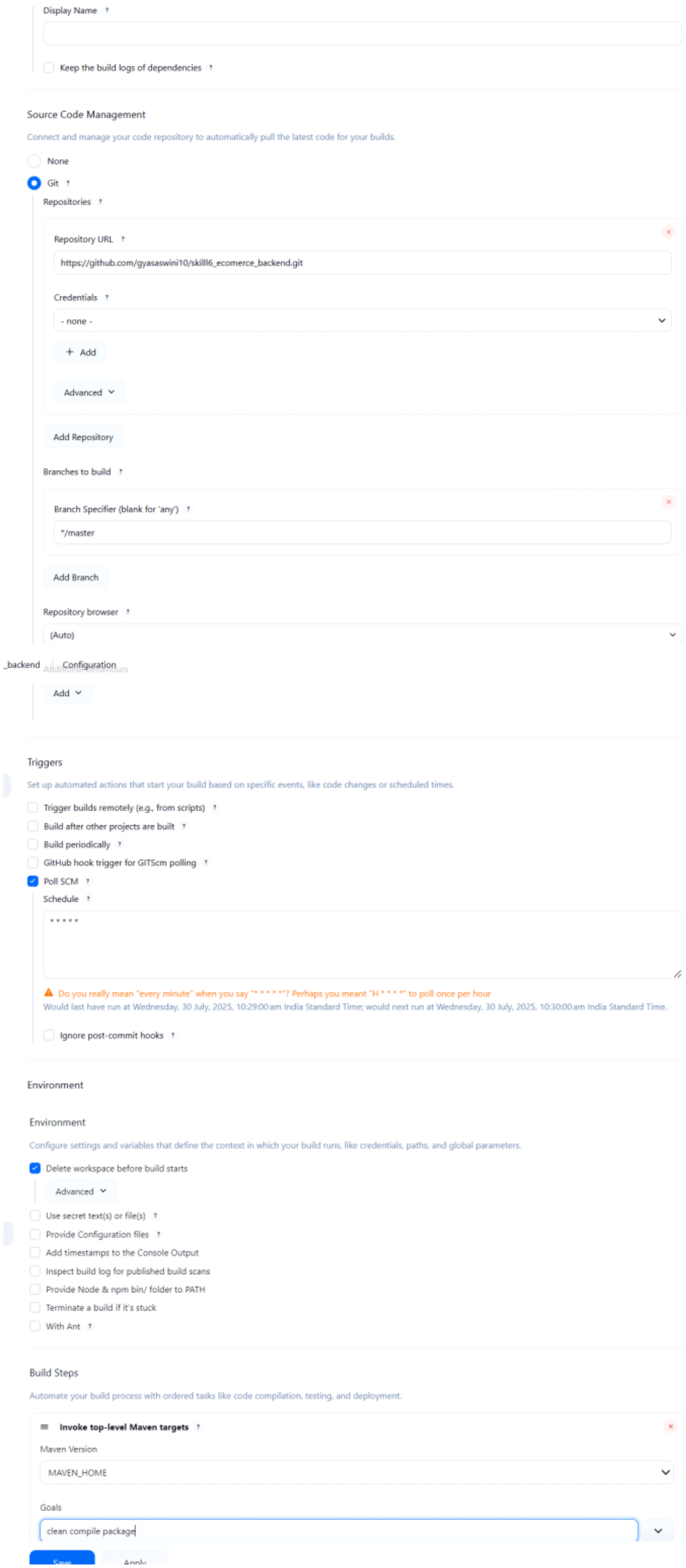
5.2 Execute Windows batch command
rmdir /S /Q "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"
/S = Delete all files and subfolders (recursively).
/Q = Quiet mode; no confirmation prompts.

mkdir "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"

xcopy /E /I /Y dist* "C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ecommerce"
/E = Copy all subdirectories, even empty ones.
/I = Treat destination as a directory (helps when it doesn't exist yet).
/Y = Suppress "overwrite?" confirmation for existing files.

5.3 Execute Windows batch command
net stop Tomcat9
net start Tomcat9

Skill6 backend ecomentrcwe



Display Name

□ Keep the build logs of dependencies

Source Code Management

Connect and manage your code repository to automatically pull the latest code for your builds.

○ None

● Git

Repositories

Repository URL

https://github.com/gyasaswini10/skilll6_ecomerce_backend.git

Credentials

- none -

Branches to build

Branch Specifier (blank for 'any')

*/master

Repository browser

(Auto)

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

□ Trigger builds remotely (e.g., from scripts)

□ Build after other projects are built

□ Build periodically

□ GitHub hook trigger for GITScm polling

☑ Poll SCM

Schedule

```
* * * * *
```

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour

Would last have run at Wednesday, 30 July, 2025, 10:29:00 am India Standard Time; would next run at Wednesday, 30 July, 2025, 10:30:00 am India Standard Time.

□ Ignore post-commit hooks

Environment

Environment

Configure settings and variables that define the context in which your build runs, like credentials, paths, and global parameters.

☑ Delete workspace before build starts

□ Use secret text(s) or file(s)

□ Provide Configuration files

□ Add timestamps to the Console Output

□ Inspect build log for published build scans

□ Provide Node & npm bin/ folder to PATH

□ Terminate a build if it's stuck

□ With Ant

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

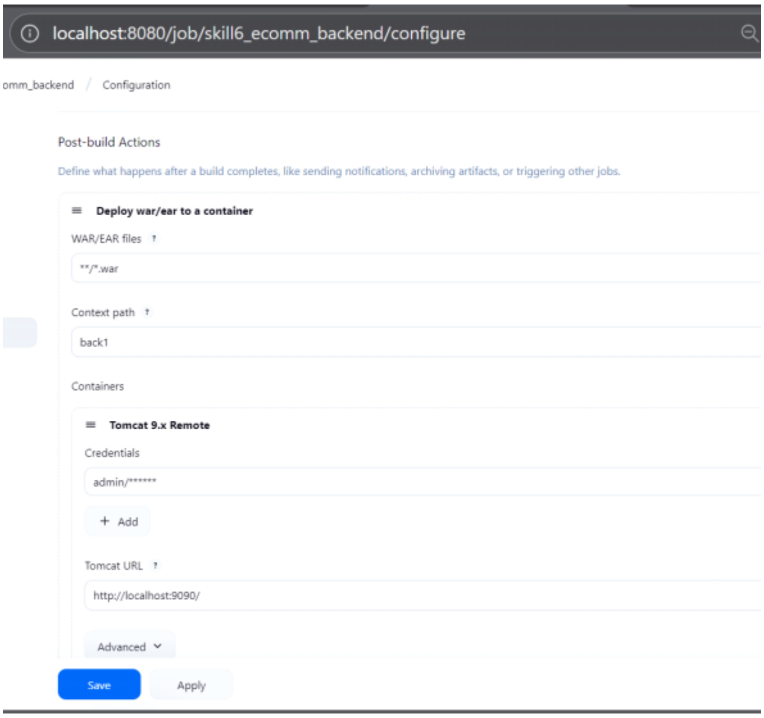
Invoke top-level Maven targets

Maven Version

MAVEN_HOME

Goals

clean compile package



localhost:8080/job/skill6_ecomm_backend/configure

omm_backend / Configuration

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Deploy war/ear to a container

WAR/EAR files: `**/*.war`

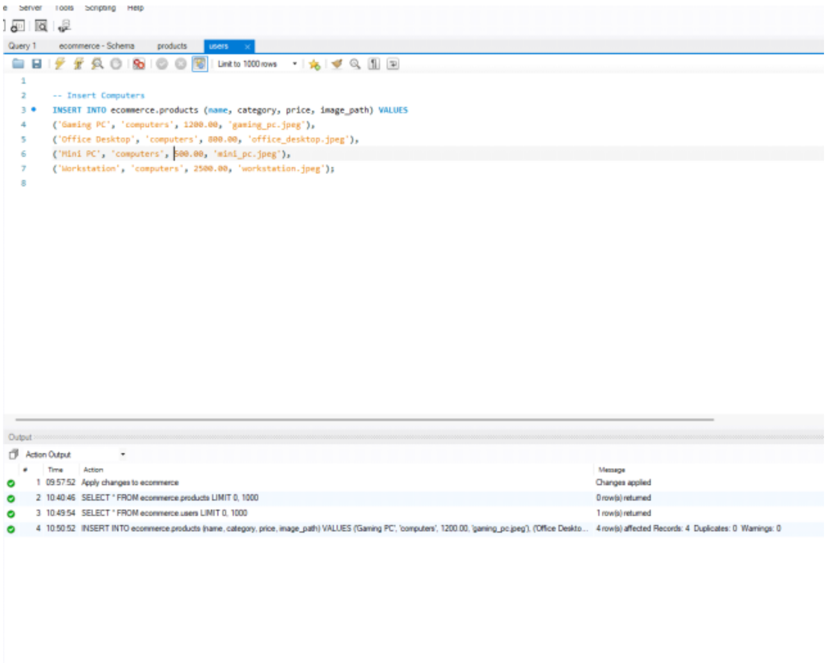
Context path: `back1`

Containers

Tomcat 9.x Remote

Credentials: admin/******

Tomcat URL: http://localhost:9090/



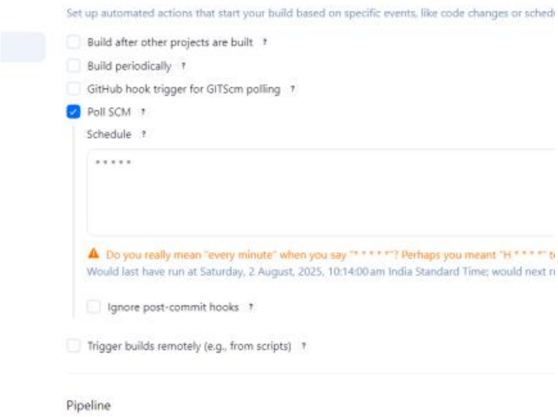
```sql

-- Insert Computers
INSERT INTO ecommerce.products (name, category, price, image_path) VALUES
('Gaming PC', 'computers', 1200.00, 'gaming_pc.jpeg'),
('Office Desktop', 'computers', 800.00, 'office_desktop.jpeg'),
('Mini PC', 'computers', 500.00, 'mini_pc.jpeg'),
('Workstation', 'computers', 2500.00, 'workstation.jpeg');

```

Output

Action Output

```
#  Time      Action                                                                                                                          Message
1  09:57:52  Apply changes to ecommerce                                                                                                      Changes applied
2  10:40:46  SELECT * FROM ecommerce.products LIMIT 0, 1000                                                                                  0 row(s) returned
3  10:49:54  SELECT * FROM ecommerce.users LIMIT 0, 1000                                                                                     1 row(s) returned
4  10:50:52  INSERT INTO ecommerce.products (name, category, price, image_path) VALUES ('Gaming PC', 'computers', 1200.00, 'gaming_pc.jpeg'), ('Office Deskto...   4 row(s) affected Records: 4  Duplicates: 0  Warnings: 0
```

PIPILEINES PRACTICAL 2   NOT FREE STYLE PROJECT



Set up automated actions that start your build based on specific events, like code changes or sched

- □ Build after other projects are built
- □ Build periodically
- □ GitHub hook trigger for GITScm polling
- ☑ Poll SCM

Schedule

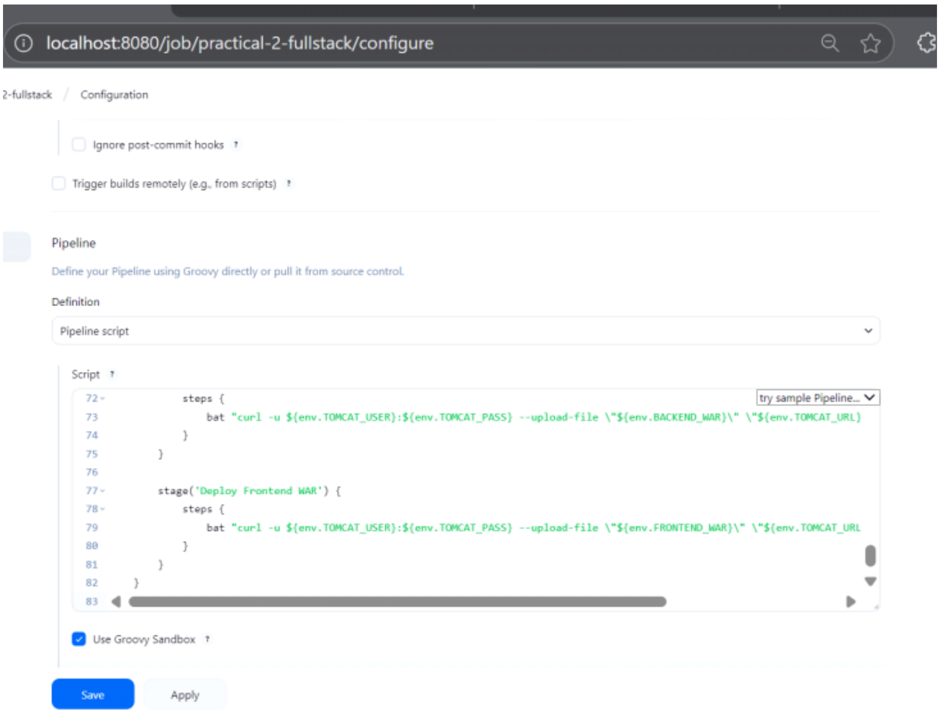
```
* * * * *
```

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" t

Would last have run at Saturday, 2 August, 2025, 10:14:00 am India Standard Time; would next r

- □ Ignore post-commit hooks
- □ Trigger builds remotely (e.g., from scripts)

Pipeline



localhost:8080/job/practical-2-fullstack/configure

2-fullstack / Configuration

□ Ignore post-commit hooks

□ Trigger builds remotely (e.g., from scripts)

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition

Pipeline script

Script

```
            steps {
                bat "curl -u ${env.TOMCAT_USER}:${env.TOMCAT_PASS} --upload-file \"${env.BACKEND_WAR}\" \"${env.TOMCAT_URL}
            }
        }

        stage('Deploy Frontend WAR') {
            steps {
                bat "curl -u ${env.TOMCAT_USER}:${env.TOMCAT_PASS} --upload-file \"${env.FRONTEND_WAR}\" \"${env.TOMCAT_URL
            }
        }
    }
```

☑ Use Groovy Sandbox

```
pipeline {
  agent any //wbere I want to deploy in jenkins from multiple nodes

  tools {
    jdk 'JDK_HOME'
    maven 'MAVEN_HOME'
  }

  environment {
    TOMCAT_URL = 'http://localhost:9090/manager/text' // eherer I want to deploy
    TOMCAT_USER = 'admin'
    TOMCAT_PASS = 'admin'

    BACKEND_REPO = 'https://github.com/gyasaswini10/CICD_Practicle_2_Backend.git
    FRONTEND_REPO = 'https://github.com/gyasaswini10/CI_CD_Practicle_2_frontend.git'

    BACKEND_DIR = 'backend'
    FRONTEND_DIR = 'frontend'

    BACKEND_WAR = 'backend/target/springapp1.war'
    FRONTEND_WAR = 'frontend/frontapp1.war'
  }

  stages {
    stage('Clone Repositories') {
      steps {
        dir("${env.BACKEND_DIR}") {
          git branch: 'main', url: "${env.BACKEND_REPO}"
        }
        dir("${env.FRONTEND_DIR}") {
          git branch: 'main', url: "${env.FRONTEND_REPO}"
        }
      }
    }

    stage('Build React Frontend') {
      steps {
        script {
          def nodeHome = tool name: 'NODE_HOME', type: 'jenkins.plugins.nodejs.tools.NodeJSInstallation'
          env.PATH = "${nodeHome}\\bin;${env.PATH}"
        }
        dir("${env.FRONTEND_DIR}") {
          bat 'npm install'
          bat 'npm run build'
        }
      }
    }

    stage('Package React as WAR') {
      steps {
        script {
          def warDir = "${env.FRONTEND_DIR}\\war_content"
          bat "if exist ${warDir} rmdir /S /Q ${warDir}"
          bat "mkdir ${warDir}\\META-INF"
          bat "mkdir ${warDir}\\WEB-INF"
          bat "xcopy /E /Y /I \"${env.FRONTEND_DIR}\\dist\\*\" \"${warDir}\\\""
          bat "jar -cvf ${env.FRONTEND_WAR} -C ${warDir} ."
        }
      }
    }

    stage('Build Spring Boot App') {
      steps {
        dir("${env.BACKEND_DIR}") {
          bat 'mvn clean package'
          bat 'rename target\\*.war springapp1.war'
        }
      }
    }

    stage('Deploy Spring Boot WAR') {
      steps {
        bat "curl -u ${env.TOMCAT_USER}:${env.TOMCAT_PASS}--upload-file \"${env.BACKEND_WAR}\" \"${env.TOMCAT_URL}/deploy?path=/springapp1&update=true\""
      }
    }

    stage('Deploy Frontend WAR') {
      steps {
        bat "curl -u ${env.TOMCAT_USER}:${env.TOMCAT_PASS}--upload-file \"${env.FRONTEND_WAR}\" \"${env.TOMCAT_URL}/deploy?path=/frontapp1&update=true\""
      }
    }
  }
}
```

Git hub

25 July 2025 03:30 PM

Install

GitHub Desktop

desktop.github.com

Git for All Platforms

git-scm.com

Configure tooling

Configure user information for all local repositories

```
$ git config --global user.name "[name]"
```

Sets the name you want attached to your commit transactions

```
$ git config --global user.email "[email address]"
```

Sets the email you want attached to your commit transactions

```
$ git config --global color.ui auto
```

Enables helpful colorization of command line output

Branches

Branches are an important part of working with Git. Any commits you make will be made on the branch you're currently "checked out" to. Use `git status` to see which branch that is.

```
$ git branch [branch-name]
```

Creates a new branch

```
$ git switch -c [branch-name]
```

Switches to the specified branch and updates the working directory

```
$ git merge [branch]
```

Combines the specified branch's history into the current branch. This is usually done in pull requests, but is an important Git operation.

```
$ git branch -d [branch-name]
```

Deletes the specified branch

Create repositories

A new repository can either be created locally, or an existing repository can be cloned. When a repository was initialized locally, you have to push it to GitHub afterwards.

```
$ git init
```

The git init command turns an existing directory into a new Git repository inside the folder you are running this command. After using the git init command, link the local repository to an empty GitHub repository using the following command:

```
$ git remote add origin [url]
```

Specifies the remote repository for your local repository. The url points to a repository on GitHub.

```
$ git clone [url]
```

Clone (download) a repository that already exists on GitHub, including all of the files, branches, and commits

The .gitignore file

Sometimes it may be a good idea to exclude files from being tracked with Git. This is typically done in a special file named .gitignore. You can find helpful templates for .gitignore files at github.com/github/gitignore.

Synchronize changes

Synchronize your local repository with the remote repository on GitHub.com

```
$ git fetch
```

Downloads all history from the remote tracking branches

```
$ git merge
```

Combines remote tracking branches into current local branch

```
$ git push
```

Uploads all local branch commits to GitHub

```
$ git pull
```

Updates your current local working branch with all new commits from the corresponding remote branch on GitHub. git pull is a combination of git fetch and git merge

Make changes

Browse and inspect the evolution of project files

```
$ git log
```

Lists version history for the current branch

```
$ git log --follow [file]
```

Lists version history for a file, beyond renames (works only for a single file)

```
$ git diff [first-branch]...[second-branch]
```

Shows content differences between two branches

```
$ git show [commit]
```

Outputs metadata and content changes of the specified commit

```
$ git add [file]
```

Snapshots the file in preparation for versioning

```
$ git commit -m "[descriptive message]"
```

Records file snapshots permanently in version history

Redo commits

Erase mistakes and craft replacement history

```
$ git reset [commit]
```

Undoes all commits after `[commit]`, preserving changes locally

```
$ git reset --hard [commit]
```

Discards all history and changes back to the specified commit

CAUTION! Changing history can have nasty side effects. If you need to change commits that exist on GitHub (the remote), proceed with caution. If you need help, reach out at github.community or contact support.

Glossary

- git: an open source, distributed version-control system
- GitHub: a platform for hosting and collaborating on Git repositories
- commit: a Git object, a snapshot of your entire repository compressed into a SHA
- branch: a lightweight movable pointer to a commit
- clone: a local version of a repository, including all commits and branches
- remote: a common repository on GitHub that all team members use to exchange their changes
- fork: a copy of a repository on GitHub owned by a different user
- pull request: a place to compare and discuss the differences introduced on a branch with reviews, comments, integrated tests, and more
- HEAD: representing your current working directory, the HEAD pointer can be moved to different branches, tags, or commits when using `git switch`

Co2_Notes

13 August 2025 09:44 AM

DOCKERS ABD CONTAINARIZATION Gmail1125@#
Virtualization it is used to run multiple number of same or different oss,
which are completely isolated from each other

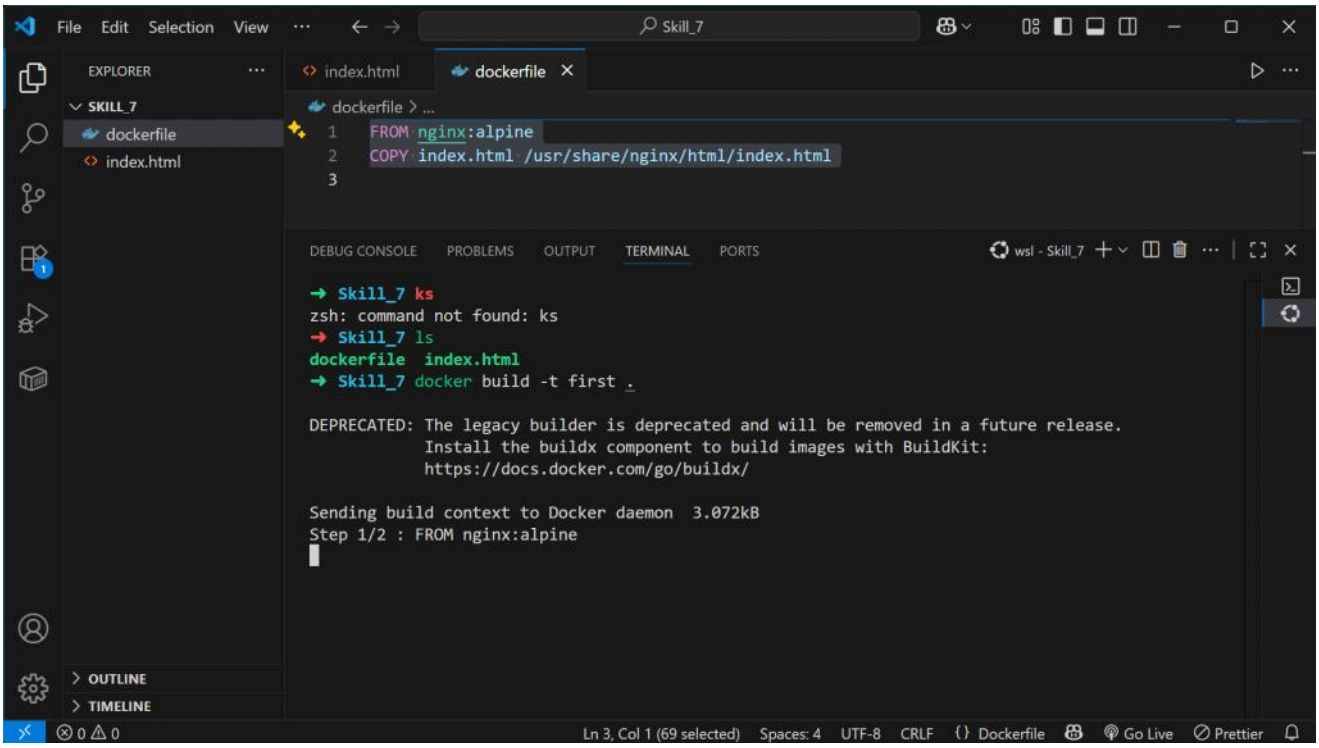
Hypervisor is a software that allows multiple os to run simulataneoulsy
on single physical machine by virtualizing the hard ware

Containarization is a software deployment process that bundles an
application code with a ll the files and libraries it needs to run on any infrastrunctre

Docker is an containerization platform that provides easy way to containerize your applications ,
which means using docker you can build container images ,run the image to create containers
and also push these containers to container registeries such as docker hub and quay.io,...etc.

Dcoker life cycle
3 imp things
Docker build :build docker images from docker file
Docker run: runs contaier from docker images
Docker push : push the container image to public or private regestries to share docker images

DEPLOYING INDEX.HTML IN DOCEKR

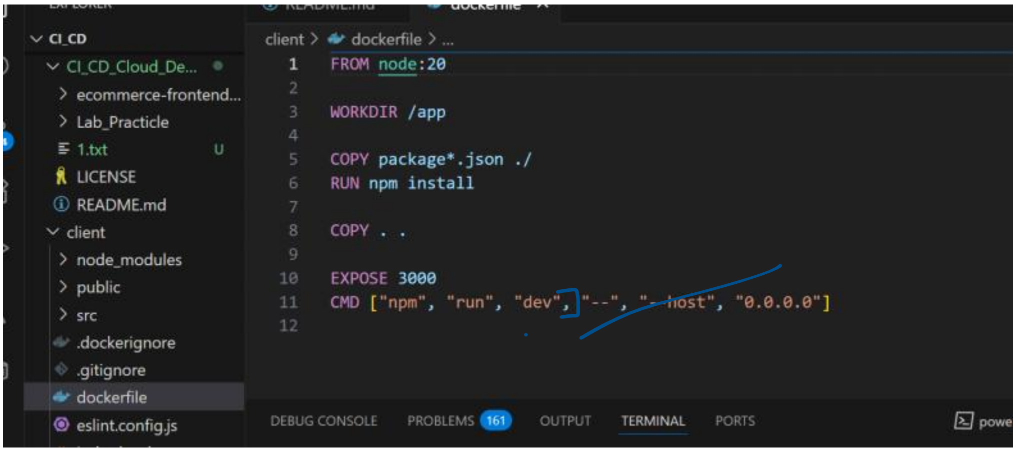

File Edit Selection View ...
Skill_7
EXPLORER
SKILL_7
dockerfile
index.html
index.html
dockerfile
dockerfile > ...
DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL PORTS
wsl - Skill_7
OUTLINE
TIMELINE
⊗ 0 ⚠ 0
Ln 3, Col 1 (69 selected) Spaces: 4 UTF-8 CRLF {} Dockerfile Go Live ⊘ Prettier

```
FROM nginx:alpine
COPY index.html /usr/share/nginx/html/index.html
```

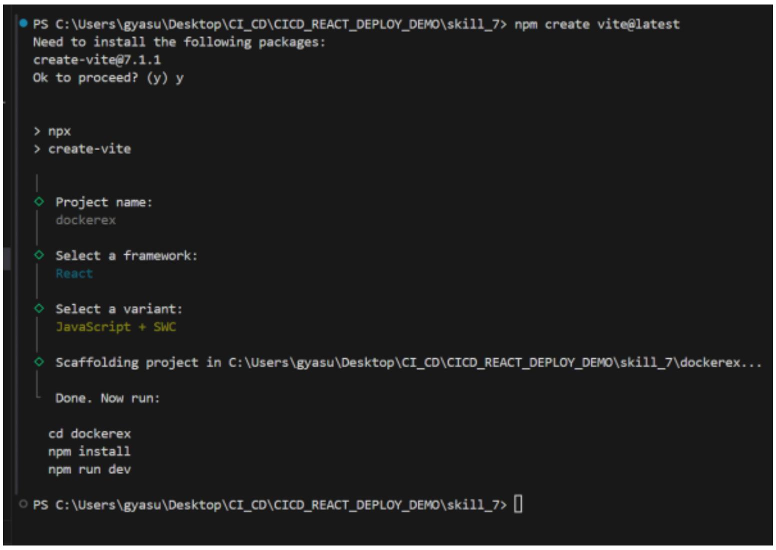
```
➜ Skill_7 ks
zsh: command not found: ks
➜ Skill_7 ls
dockerfile  index.html
➜ Skill_7 docker build -t first .

DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  3.072kB
Step 1/2 : FROM nginx:alpine
█
```

Docker run -d -p 8081:80 first

DEPLOYING REACT APP IN DOCKER



```
FROM node:20

WORKDIR /app

COPY package*.json ./
RUN npm install

COPY . .

EXPOSE 3000
CMD ["npm", "run", "dev", "--", "--host", "0.0.0.0"]
```

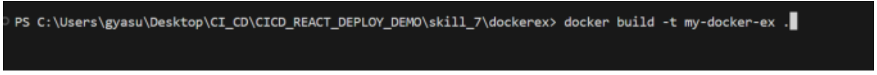


```
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7> npm create vite@latest
Need to install the following packages:
create-vite@7.1.1
Ok to proceed? (y) y


> npx
> create-vite

│
◇  Project name:
│  dockerex
│
◇  Select a framework:
│  React
│
◇  Select a variant:
│  JavaScript + SWC
│
◇  Scaffolding project in C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex...
│
└  Done. Now run:

  cd dockerex
  npm install
  npm run dev

PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7> 
```

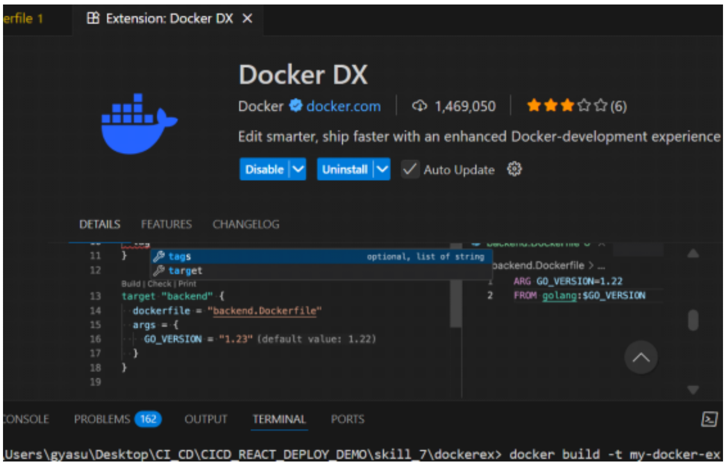
```
FROM node:20-alpine
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 3000
CMD ["npm", "run", "dev", "--", "--host", "0.0.0.0"]
```

```
If alpine or os not specify choose linux default
```

-- and --host will deploy in linux but we cant seee result in windows browser



```
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker build -t my-docker-ex .
```

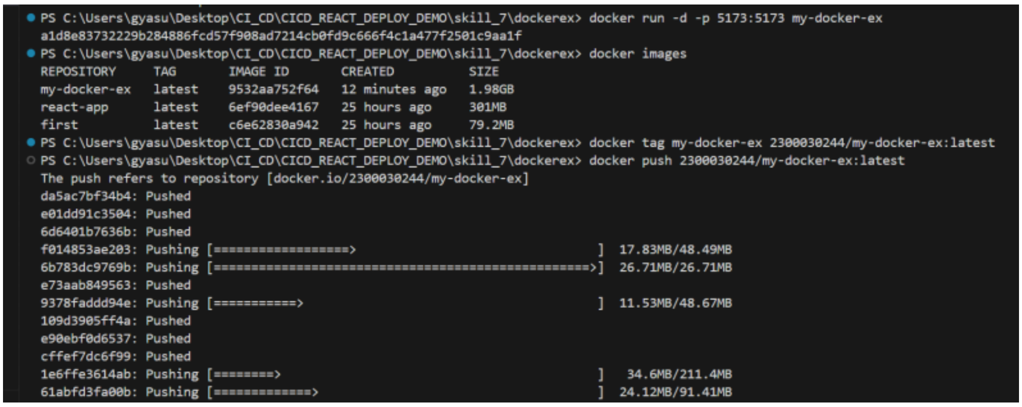


```
}
target "backend" {
  dockerfile = "backend.Dockerfile"
  args = {
    GO_VERSION = "1.23" (default value: 1.22)
  }
}
```

```
ARG GO_VERSION=1.22
FROM golang:$GO_VERSION
```

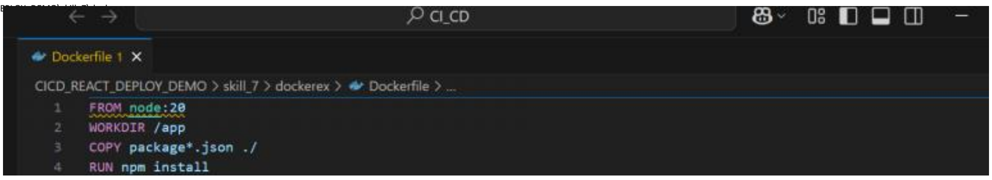
```
Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker build -t my-docker-ex
```

-t is tag
docker build -t my-docker-ex .
docker run -d -p 5173:5173 dockerex



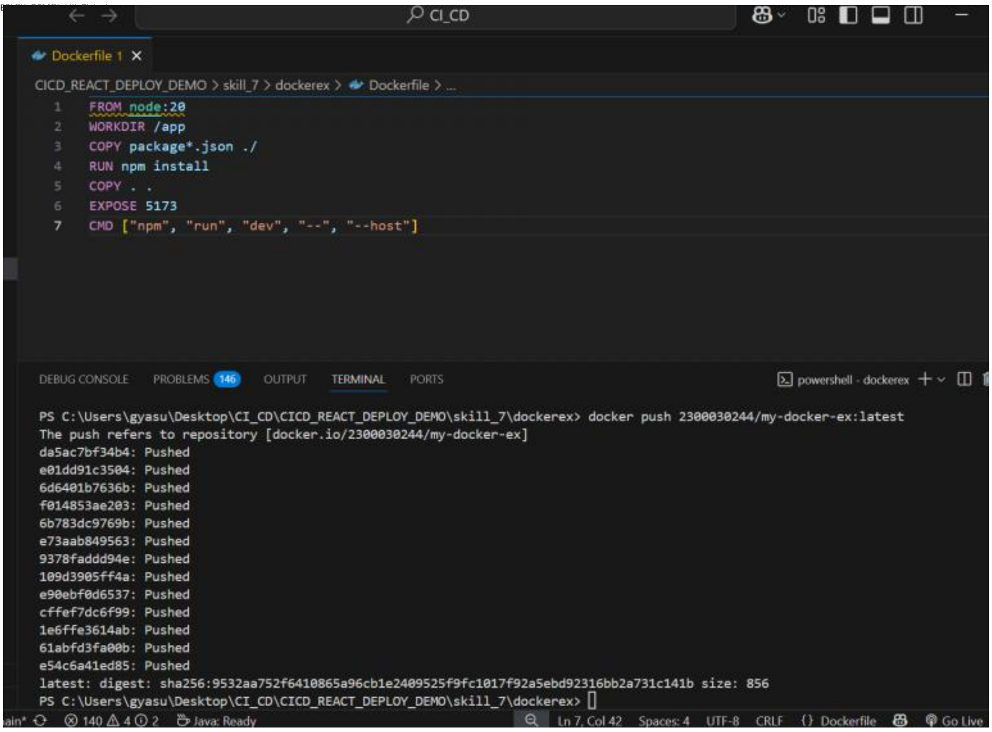
```
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker run -d -p 5173:5173 my-docker-ex
a1d8e83732229b284886fcd57f908ad7214cb0fd9c666f4c1a477f2501c9aa1f
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker images
REPOSITORY     TAG       IMAGE ID       CREATED          SIZE
my-docker-ex   latest    9532aa752f64   12 minutes ago   1.98GB
react-app      latest    6ef90dee4167   25 hours ago     301MB
first          latest    c6e62830a942   25 hours ago     79.2MB
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker tag my-docker-ex 2300030244/my-docker-ex:latest
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker push 2300030244/my-docker-ex:latest
The push refers to repository [docker.io/2300030244/my-docker-ex]
da5ac7bf34b4: Pushed
e01dd91c3504: Pushed
6d6401b7636b: Pushed
f014853ae203: Pushing [=================>                                 ]  17.83MB/48.49MB
6b783dc9769b: Pushing [==================================================>]  26.71MB/26.71MB
e73aab849563: Pushed
9378faddd94e: Pushing [===========>                                       ]  11.53MB/48.67MB
109d3905ff4a: Pushed
e90ebf0d6537: Pushed
cffef7dc6f99: Pushed
1e6ffe3614ab: Pushing [========>                                          ]   34.6MB/211.4MB
61abfd3fa00b: Pushing [=============>                                     ]  24.12MB/91.41MB
```

```
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker tag my-docker-ex 2300030244/my-docker-ex:latest
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker push 2300030244/my-docker-ex:latest
The push refers to repository [docker.io/2300030244/my-docker-ex]
da5ac7bf34b4: Pushed
latest: digest: sha256:9532aa752f6410865a96cb1e2409525f9fc1017f92a5ebd92316bb2a731c141b size: 856
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DE
```


CI_CD
Dockerfile 1
CICD_REACT_DEPLOY_DEMO > skill_7 > dockerex > Dockerfile > ...

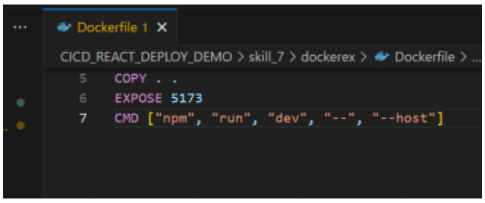
```
FROM node:20
WORKDIR /app
COPY package*.json ./
RUN npm install
```

PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DE

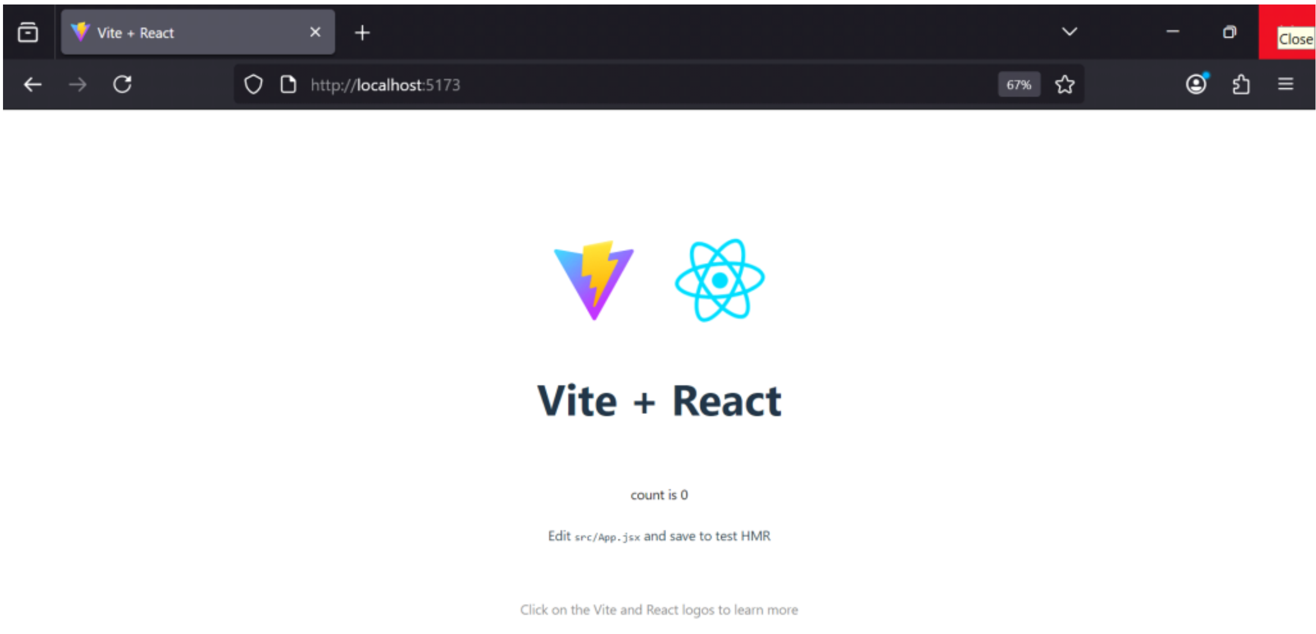

CI_CD
Dockerfile 1
CICD_REACT_DEPLOY_DEMO > skill_7 > dockerex > Dockerfile > ...
DEBUG CONSOLE
PROBLEMS 146
OUTPUT
TERMINAL
PORTS
powershell - dockerex
ain* ⊗ 140 ⚠ 4 ⓘ 2 Java: Ready
Ln 7, Col 42 Spaces: 4 UTF-8 CRLF {} Dockerfile Go Live

```
FROM node:20
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 5173
CMD ["npm", "run", "dev", "--", "--host"]
```

```
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> docker push 2300030244/my-docker-ex:latest
The push refers to repository [docker.io/2300030244/my-docker-ex]
da5ac7bf34b4: Pushed
e01dd91c3504: Pushed
6d6401b7636b: Pushed
f014853ae203: Pushed
6b783dc9769b: Pushed
e73aab849563: Pushed
9378faddd94e: Pushed
109d3905ff4a: Pushed
e90ebf0d6537: Pushed
cffef7dc6f99: Pushed
1e6ffe3614ab: Pushed
61abfd3fa00b: Pushed
e54c6a41ed85: Pushed
latest: digest: sha256:9532aa752f6410865a96cb1e2409525f9fc1017f92a5ebd92316bb2a731c141b size: 856
PS C:\Users\gyasu\Desktop\CI_CD\CICD_REACT_DEPLOY_DEMO\skill_7\dockerex> 
```


Dockerfile 1
CICD_REACT_DEPLOY_DEMO > skill_7 > dockerex > Dockerfile > ...

```
COPY . .
EXPOSE 5173
CMD ["npm", "run", "dev", "--", "--host"]
```



Vite + React

count is 0

Edit src/App.jsx and save to test HMR

Click on the Vite and React logos to learn more

Dockerfile

Base Image---> OS/node/java/python
Application Code--> Host os directory to Base image os directory
Commands--> buliding/deployment

Types of Dockerfiles
----
1. Single stage --->building of project --> static deployment
2. Multi stage --> building and production --> dynamic deployment

Exp: containeraize react application using Single stage

Step1: create react application

```
npm create vite@latest dockerex
 cd dockerex
  npm install
  npm run dev
```

Step2: Configure docker in VSCode (terminal/browser)

-add docker extension in the vscode
-complete docker authentication
docker login docker.io

Step3: create Dockerfile in the projetc root directory

```
FROM node:20
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 5173
CMD ["npm", "run", "dev" "--" "--host"]
```

step4: conver project into docker image

```
docker build -t my-docker-ex .
```

step4: run docker image

```
docker run -p 5173:5173 my-docker-ex
```

step5: push docker image into dockerhub

```
docker images
docker tag my-docker-ex srithar123/my-docker-ex:latest
docker push srithar123/my-docker-ex:latest
```

Exp: containeraize react application using Multi stage

Step3: create Dockerfile in the projetc root directory
MULYI BUILD

```
# -------- Stage 1: Build with Vite --------
FROM node:20 AS builder

WORKDIR /app

COPY package*.json ./
RUN npm install

COPY . .

RUN npm run build

# -------- Stage 2: Serve with Nginx --------
FROM nginx:alpine

# Remove default Nginx content
RUN rm -rf /usr/share/nginx/html/*

# Copy build output to Nginx public directory
COPY --from=builder /app/dist /usr/share/nginx/html

# Optional: Custom Nginx config (uncomment if needed)
# COPY nginx.conf /etc/nginx/nginx.conf

EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
```

step4: conver project into docker image

```
docker build -t my-docker-ex-multi .
```

step5: run docker image

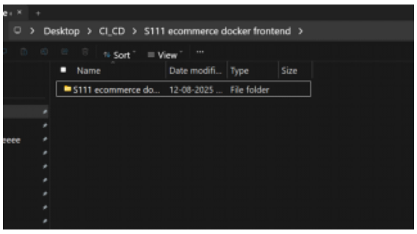
```
docker run -d -p 3000:80 my-docker-ex-multi
```

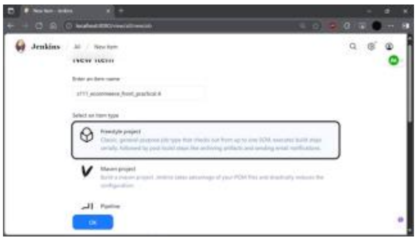
step6: push docker image into dockerhub

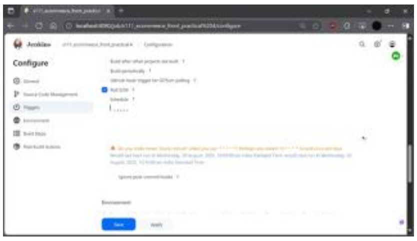
```
docker images
docker tag my-docker-ex-multi srithar123/my-docker-ex-multi:latest
docker push srithar123/my-docker-ex-multi:latest
```

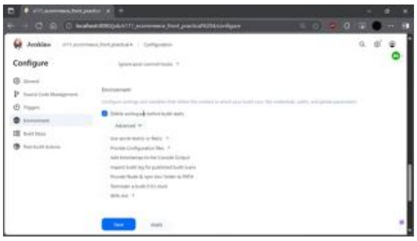
```
# Step 1: Build stage (Node.js for React/Vite)
FROM node:18 AS build-stage
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
RUN npm run build
# Step 2: Runtime stage (Tomcat for serving)
FROM tomcat:9-jdk17
RUN rm -rf /usr/local/tomcat/webapps/*
# Copy React build output into Tomcat webapps
COPY --from=build-stage /app/dist /usr/local/tomcat/webapps/ecommerce
EXPOSE 8082
CMD ["catalina.sh", "run"]   // in tomacat conf we can open through tomcat
```

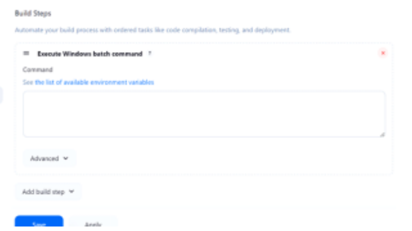
PRACTICAL 4








Jenkins
Configure
General
Source Code Management
Triggers
Environment
Build Steps
Post-build Actions
Environment
Delete workspace before build starts
Advanced
Provide Configuration files
Add timestamps to the Console Output
Terminate a build if it's stuck
With Ant
Save
Apply


Build Steps

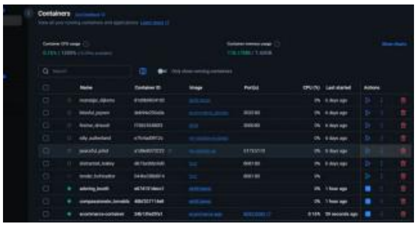
Automate your build process with ordered tasks like code compilation, testing, and deployment.

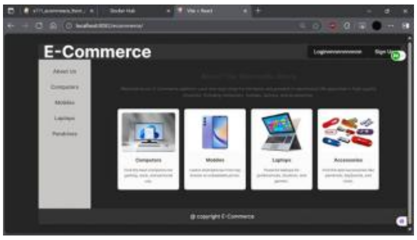
Execute Windows batch command

Command

See the list of available environment variables

```
docker build -t ecommerce-app .


docker stop ecommerce-container || echo "No container to stop"
docker rm ecommerce-container  || echo "No container to remove"


docker run -d -p 8082:8080 --name ecommerce-container ecommerce-app
```





ECOMMMERCE Bbackend docker

ecomm backend d...

23000302 44_skill9 s...

docker

ecomm backend d...

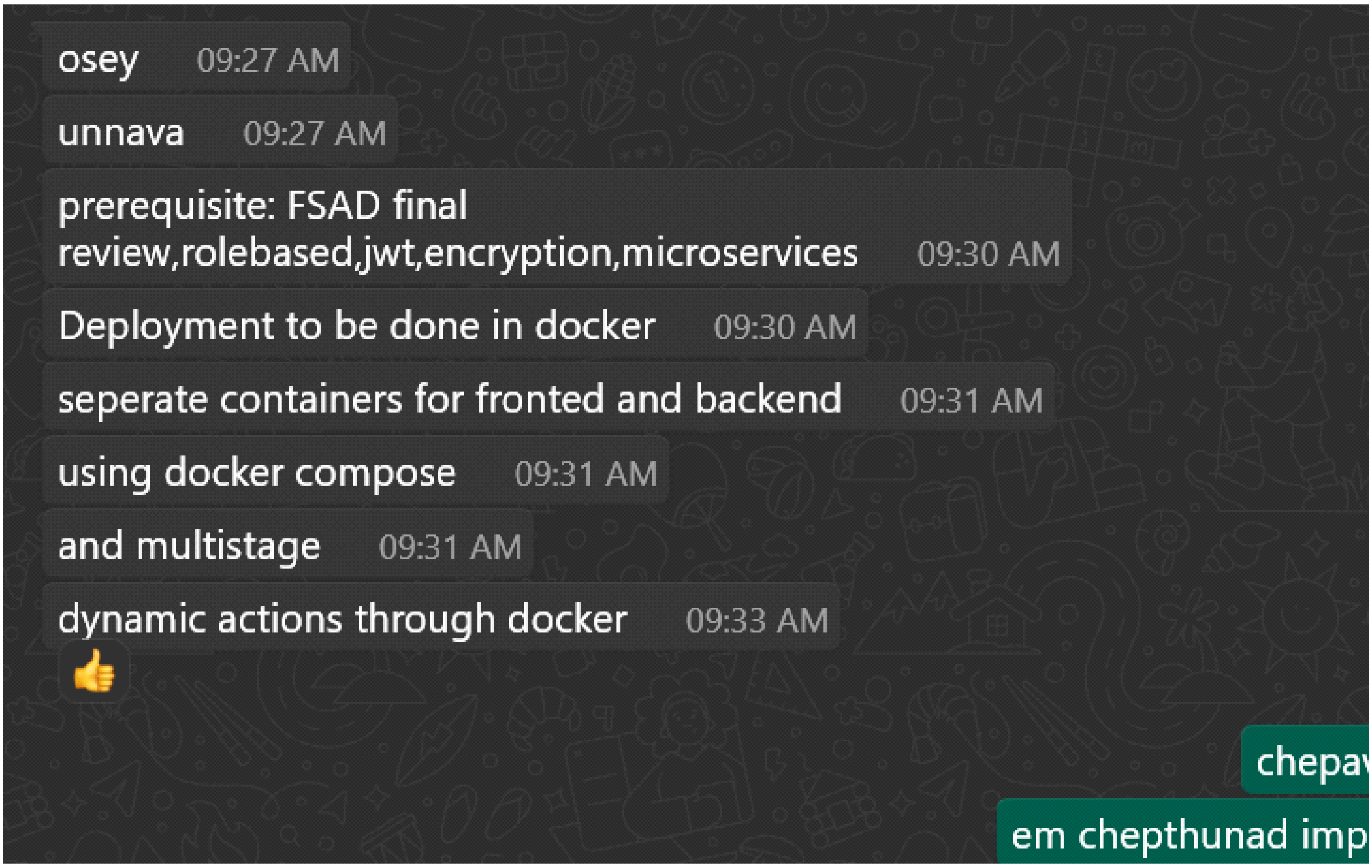
2300030244_skill9 s...

spring framfe work
version 2 other not
wokring
javax for entity

PRACTICLE 5
SKILL 10
DOCKER
Multistage
deploy

Project

30 August 2025 09:39 AM



osey 09:27 AM

unnava 09:27 AM

prerequisite: FSAD final review,rolebased,jwt,encryption,microservices 09:30 AM

Deployment to be done in docker 09:30 AM

seperate containers for fronted and backend 09:31 AM

using docker compose 09:31 AM

and multistage 09:31 AM

dynamic actions through docker 09:33 AM

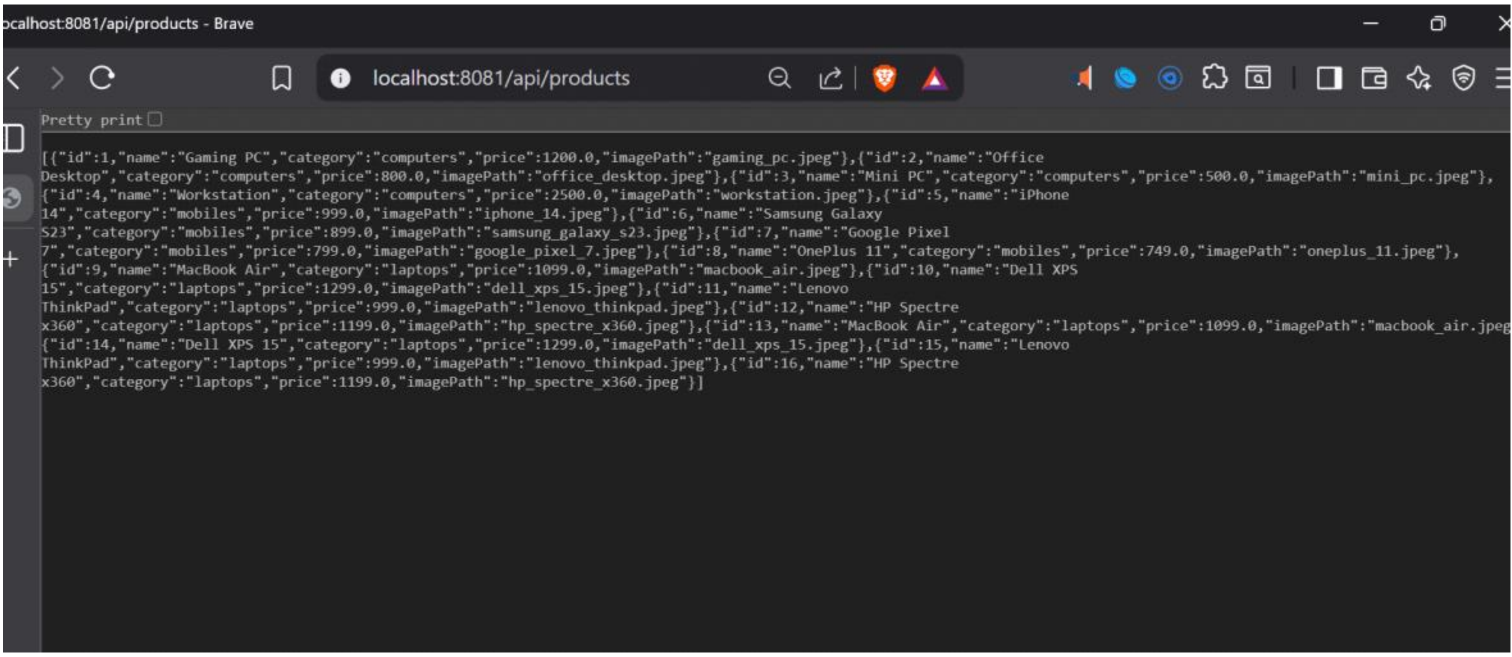
👍

chepa

em chepthunad imp

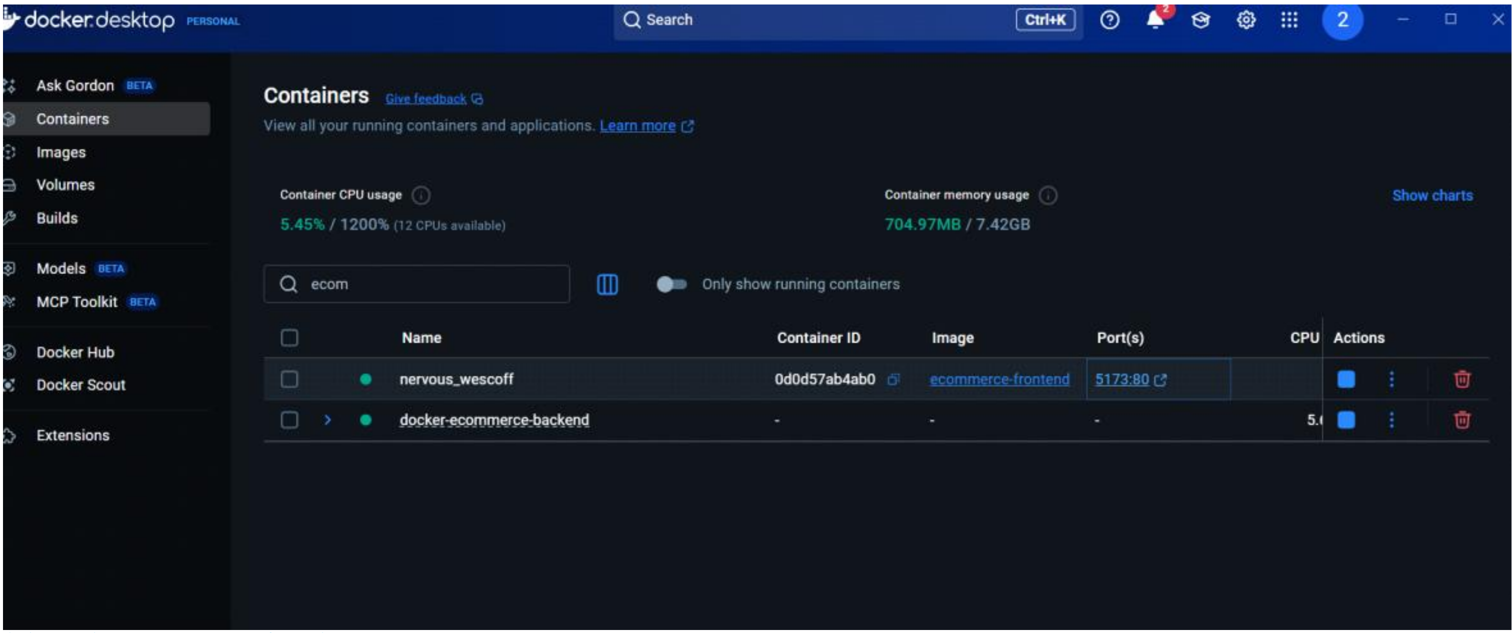
Cicd skill 10 and practical 4

06 September 2025 10:08 AM



ocalhost:8081/api/products - Brave
localhost:8081/api/products
Pretty print □

```
[{"id":1,"name":"Gaming PC","category":"computers","price":1200.0,"imagePath":"gaming_pc.jpeg"},{"id":2,"name":"Office
Desktop","category":"computers","price":800.0,"imagePath":"office_desktop.jpeg"},{"id":3,"name":"Mini PC","category":"computers","price":500.0,"imagePath":"mini_pc.jpeg"},
{"id":4,"name":"Workstation","category":"computers","price":2500.0,"imagePath":"workstation.jpeg"},{"id":5,"name":"iPhone
14","category":"mobiles","price":999.0,"imagePath":"iphone_14.jpeg"},{"id":6,"name":"Samsung Galaxy
S23","category":"mobiles","price":899.0,"imagePath":"samsung_galaxy_s23.jpeg"},{"id":7,"name":"Google Pixel
7","category":"mobiles","price":799.0,"imagePath":"google_pixel_7.jpeg"},{"id":8,"name":"OnePlus 11","category":"mobiles","price":749.0,"imagePath":"oneplus_11.jpeg"},
{"id":9,"name":"MacBook Air","category":"laptops","price":1099.0,"imagePath":"macbook_air.jpeg"},{"id":10,"name":"Dell XPS
15","category":"laptops","price":1299.0,"imagePath":"dell_xps_15.jpeg"},{"id":11,"name":"Lenovo
ThinkPad","category":"laptops","price":999.0,"imagePath":"lenovo_thinkpad.jpeg"},{"id":12,"name":"HP Spectre
x360","category":"laptops","price":1199.0,"imagePath":"hp_spectre_x360.jpeg"},{"id":13,"name":"MacBook Air","category":"laptops","price":1099.0,"imagePath":"macbook_air.jpeg
{"id":14,"name":"Dell XPS 15","category":"laptops","price":1299.0,"imagePath":"dell_xps_15.jpeg"},{"id":15,"name":"Lenovo
ThinkPad","category":"laptops","price":999.0,"imagePath":"lenovo_thinkpad.jpeg"},{"id":16,"name":"HP Spectre
x360","category":"laptops","price":1199.0,"imagePath":"hp_spectre_x360.jpeg"}]
```



Containers

View all your running containers and applications. Learn more

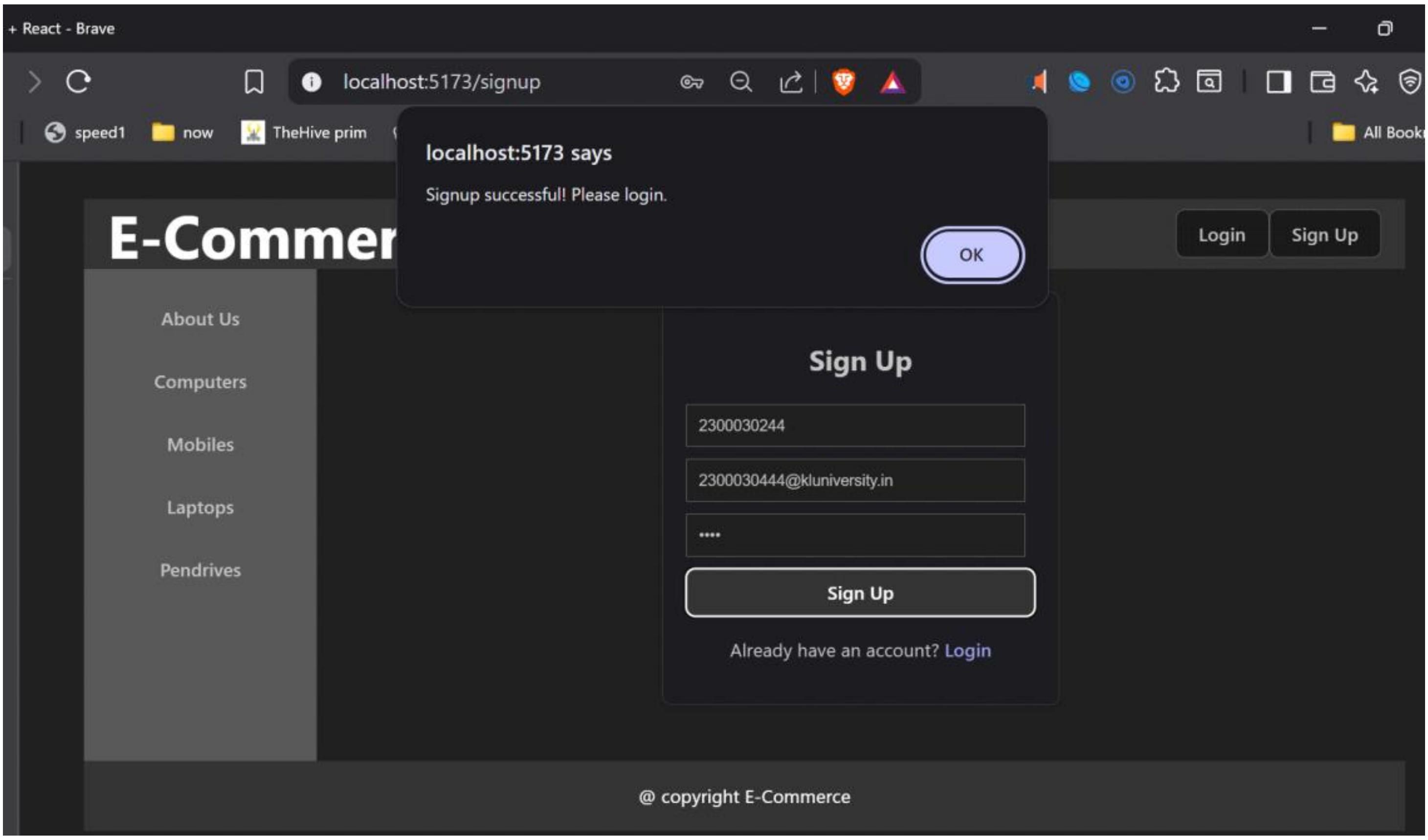
Container CPU usage
5.45% / 1200% (12 CPUs available)

Container memory usage
704.97MB / 7.42GB

ecom

□	Name	Container ID	Image	Port(s)	CPU	Actions
□	● nervous_wescoff	0d0d57ab4ab0	ecommerce-frontend	5173:80		
□	› ● docker-ecommerce-backend	-	-	-	5.(	

docker run -d -p 5173:80 ecommerce-frontend



localhost:5173 says

Signup successful! Please login.

2300030244

2300030444@kluniversity.in

••••

DOCKER

MYSQL