Practical 3: Integrate Jenkins CI/CD for a fullstack project in AWS Linux environment

1. RDS Database

Choose a database creation method : **standard create**

Engine options: **mySQL**

Templates: **free tier**

DB instance identifier: **database-1 (anything)**

Credentials Settings:

Master username: **admin**

Master password: **adminadmin**

Instance configuration: **db.t3.micro**

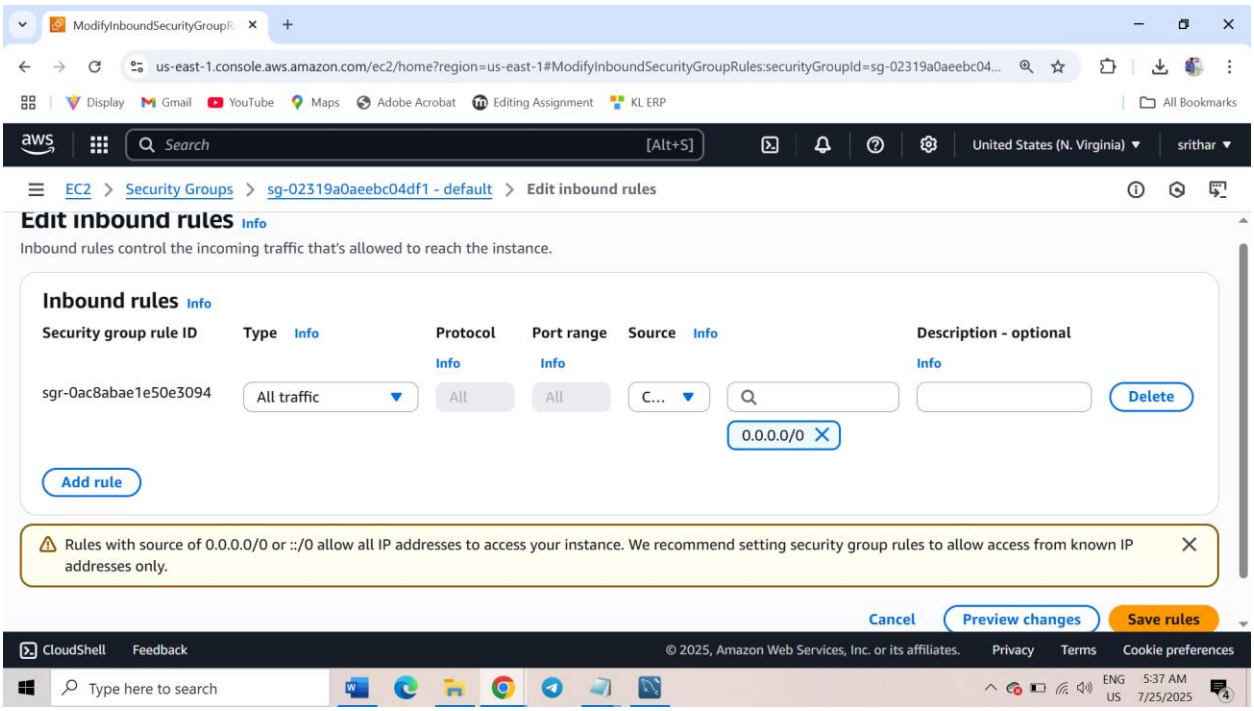
Public access: **yes**

Existing VPC security groups: **default (Enable all traffic)**

Additional configuration:

Initial database name: **cicd**

Wait for 5 mins to complete setup till creates End point & port number.



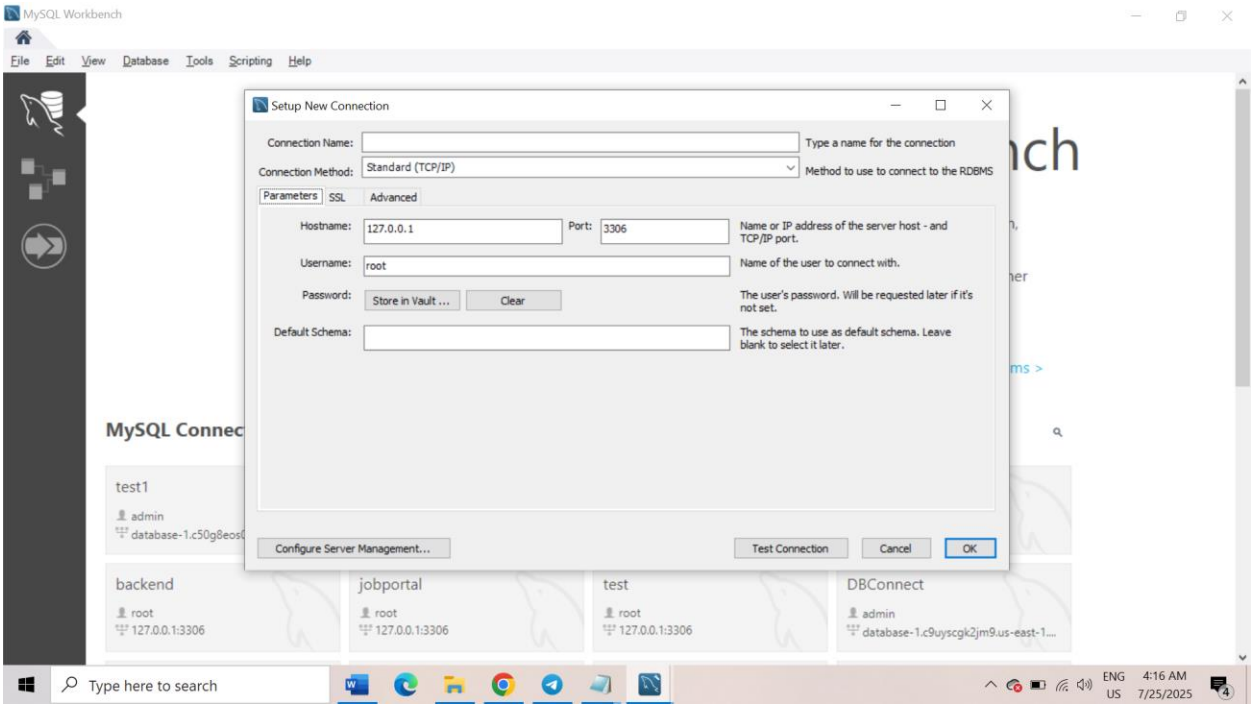
Ensure the security group having all traffic.

Endpoint: database-2.ckhoqoim0209.us-east-1.rds.amazonaws.com

Port: 3306

Connecting with mysql workbench

Open mySQL wokbench



Connection name: **cicd**

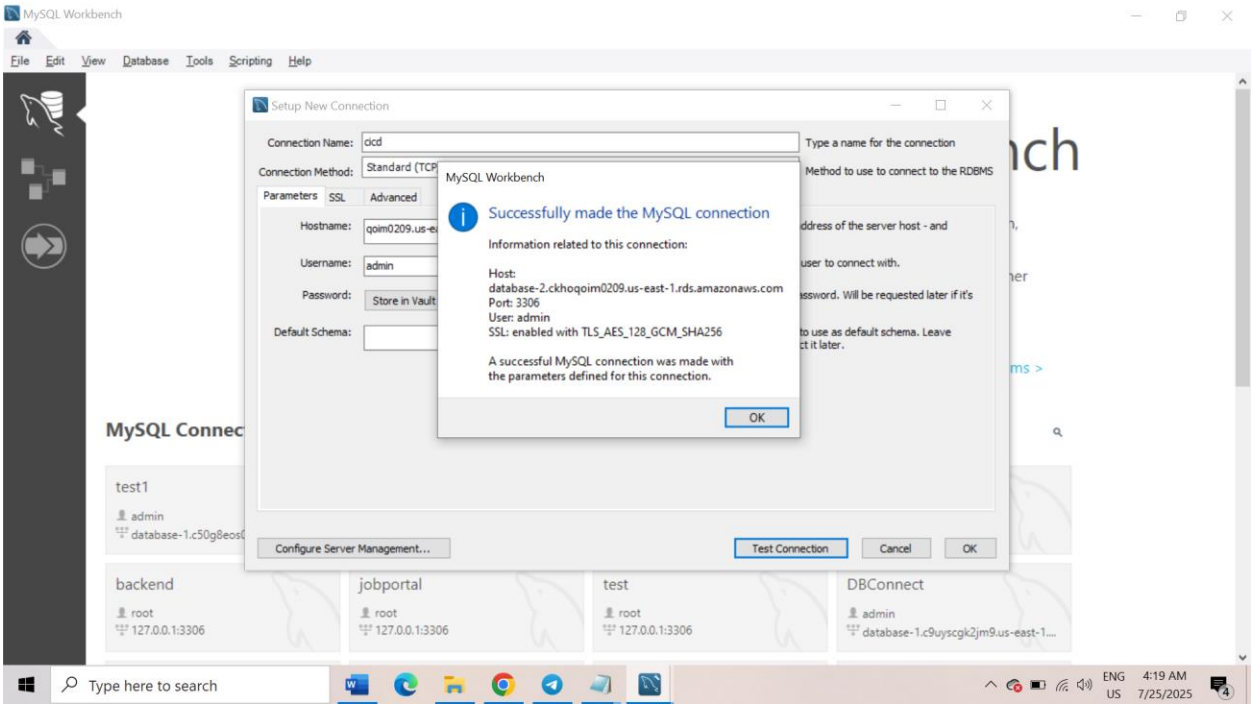
Host name: **database-2.ckhoqoim0209.us-east-1.rds.amazonaws.com**

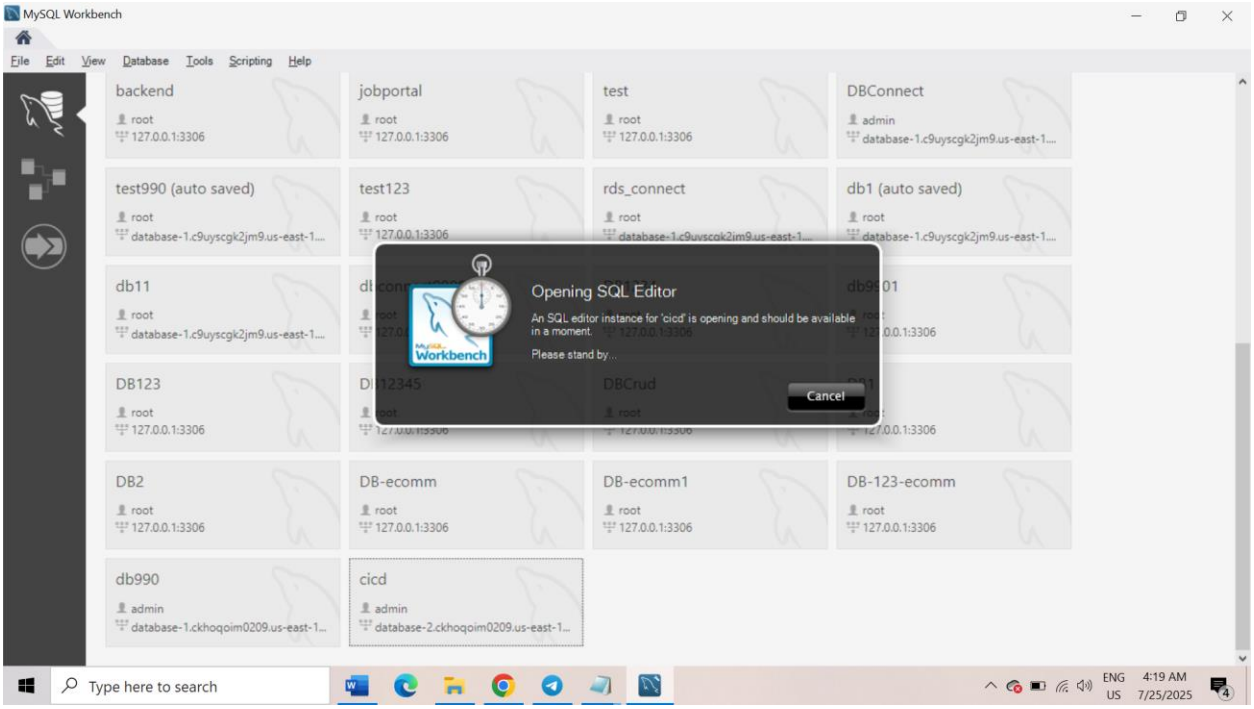
Port: **3306**

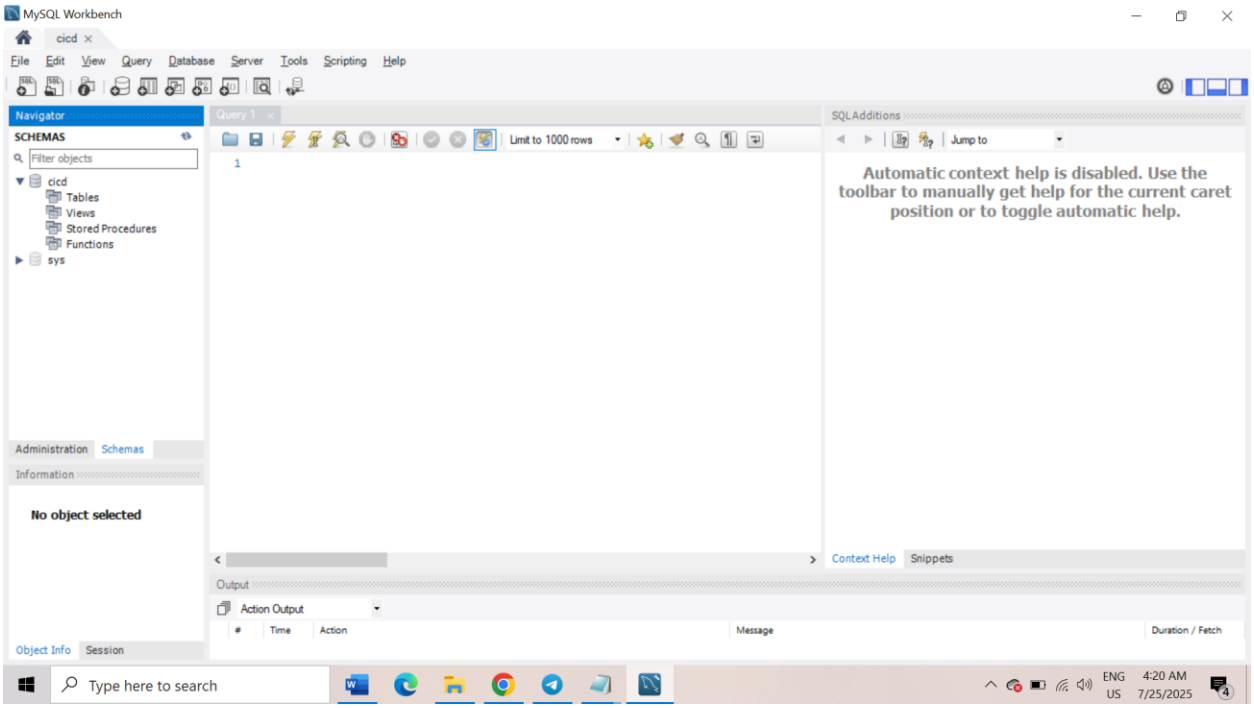
User name: **admin**

Password: **adminadmin**

Test connection→Ok→ok


MySQL Workbench
File Edit View Database Tools Scripting Help
Setup New Connection
Connection Name: cicd
Type a name for the connection
Connection Method: Standard (TCP
Method to use to connect to the RDBMS
Parameters SSL Advanced
Hostname:
Username: admin
Password: Store in Vault
Default Schema:
MySQL Workbench
Successfully made the MySQL connection
Information related to this connection:
Host:
database-2.ckhoqoim0209.us-east-1.rds.amazonaws.com
Port: 3306
User: admin
SSL: enabled with TLS_AES_128_GCM_SHA256
A successful MySQL connection was made with
the parameters defined for this connection.
OK
Configure Server Management...
Test Connection
Cancel
OK
MySQL Connec
test1
admin
backend
root
127.0.0.1:3306
jobportal
test
DBConnect
admin
Type here to search
ENG
US
4:19 AM
7/25/2025


MySQL Workbench
File Edit View Database Tools Scripting Help
backend
jobportal
test
DBConnect
admin
test990 (auto saved)
test123
rds_connect
db1 (auto saved)
db11
DB123
DB2
DB-ecomm
DB-ecomm1
DB-123-ecomm
db990
cicd
Opening SQL Editor
An SQL editor instance for 'cicd' is opening and should be available in a moment.
Please stand by...
Cancel
Type here to search
ENG
US
4:19 AM
7/25/2025



Modify the host,username,password in backend application.properties also.

database-2.ckhoqoim0209.us-east-1.rds.amazonaws.com

Port: **3306**

User name: **admin**

Password: **adminadmin**

2. EC2 Setup:

Goto AWS management console→Ec2→Instance→Launch Instance

Name: **machine1**

Os: **linux (free tier)**

Instance type: **t3.medium**

Create key pair:

Key pair type: **rsa**

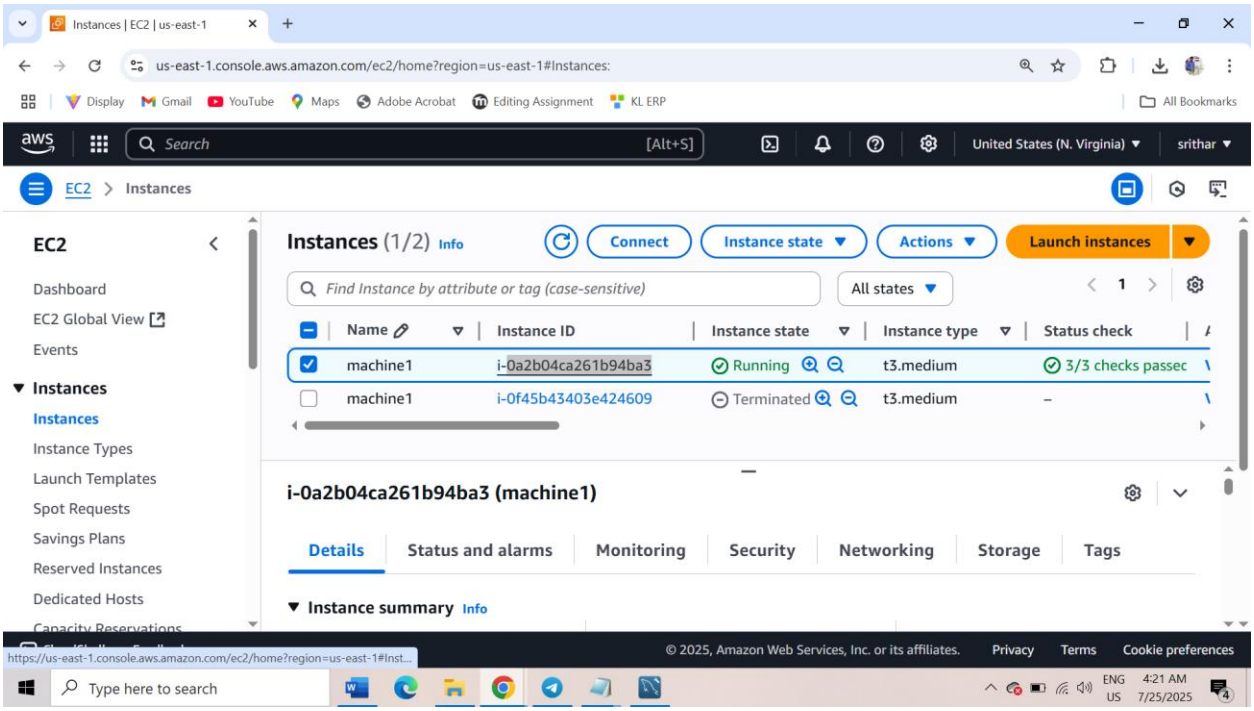
Private key file format: **.pem**

Network settings

Select existing security group: **Default (Enable all traffic)**

Configure storage: **20GB**

Launch.



Update the redirect url in frontend.

fullstackapp/crud_frontend/crud_frontend-main/src/App.jsx

const BASE_URL = 'http://**3.82.247.137**:9090/springapp1';

Click over the instance ID→connect→Ec2 instance connect→connect

Tomcat 9 Installation & Configuration on EC2 (Linux)

1. Install Prerequisites

sudo -i

yum update -y

yum install -y java-17-amazon-corretto-devel maven git nginx unzip curl --allowerasing

✅ Check Java installation:

java -version

If java: command not found, run:

export JAVA_HOME=/usr/lib/jvm/java-17-amazon-corretto.x86_64

export PATH=$JAVA_HOME/bin:$PATH

ln -sf $JAVA_HOME/bin/java /usr/bin/java

ln -sf $JAVA_HOME/bin/javac /usr/bin/javac

🔷 **2. Download & Set Up Tomcat**

wget https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.89/bin/apache-tomcat-9.0.89.tar.gz

tar -xvzf apache-tomcat-9.0.89.tar.gz

mv apache-tomcat-9.0.89 tomcat9

chmod +x tomcat9/bin/*.sh

🔷 **3. Change Tomcat Port to 9090**

nano tomcat9/conf/server.xml

Find and replace:

<Connector port="8080" protocol="HTTP/1.1"

with:

<Connector port="9090" protocol="HTTP/1.1"

📌 Save with Ctrl+O, Enter, then Ctrl+X

🔷 4. Add Tomcat Deployment User

nano tomcat9/conf/tomcat-users.xml

Add inside <tomcat-users>:

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<user username="admin" password="admin" roles="manager-gui,manager-script"/>

🔷 5. Allow Remote Access (Manager & Host Manager)

Edit both files:

nano tomcat9/webapps/manager/META-INF/context.xml

nano tomcat9/webapps/host-manager/META-INF/context.xml

Comment this block in **both files**:

xml

CopyEdit

<!--

<Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1" />

-->

🔷 6. Start Tomcat

cd tomcat9/bin

export JAVA_HOME=/usr/lib/jvm/java-17-amazon-corretto.x86_64

export PATH=$JAVA_HOME/bin:$PATH

./startup.sh

🔷 7. Access Tomcat in Browser

Open:

http://<your-ec2-public-ip>:9090

✅ Make sure **port 9090** is open in your EC2 instance's **Security Group**:

- **Type**: Custom TCP
- **Port**: 9090
- **Source**: 0.0.0.0/0 (for testing)

✅ Jenkins Installation & Setup on EC2 (Linux)

Jenkins Setup for Amazon Linux 2 (RHEL-style):

Switch to root

sudo -i

Add Jenkins repo

wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

Install Jenkins

yum install -y jenkins

Start and enable Jenkins service

systemctl start jenkins

systemctl enable jenkins

✅ **Allow Jenkins Port (8080) in Security Group:**

- Go to AWS EC2 → Security Groups → Inbound Rules → Add Rule:
  - Type: **Custom TCP**
  - Port: **8080**
  - Source: **0.0.0.0/0** *(or restrict to your IP)*

✅ **Access Jenkins:**

http://<your-ec2-public-ip>:8080

Get the initial password:

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

✅ **Jenkins Configuration**

1. Install Plugins

From Jenkins Dashboard:

- Manage Jenkins → Plugin Manager → Available
- Install:
  - ✅ Git push Plugin
  - ✅ Pipeline plugin
  - ✅ NodeJS

- ✅ Maven Integration

2. Configure Global Tools

From: Manage Jenkins → Global Tool Configuration

JDK

Check the jdk path

readlink -f $(which java) | sed 's:/bin/java::'

- Name: JDK_HOME
- Path: /usr/lib/jvm/java-17-amazon-corretto.x86_64

Maven

Check the maven path

readlink -f $(which mvn)

- Name: MAVEN_HOME
- Path: /usr/share/maven/bin/mvn

NodeJS

- Name: NODE_HOME
- Check: ✅ Install automatically
- Version: Latest LTS

✅ **Create Jenkins Pipeline Job**

1. Create Job

- Dashboard → New Item → Name: fullstack-deploy
- Select: **Pipeline**
- Click: **OK**

2. Configure Pipeline

- Scroll to **Pipeline** section:
    - Definition: Pipeline script from SCM
    - SCM: Git
    - Repo URL: https://github.com/srithars/fullstackapp.git
    - Branch: */master
    - Script Path: Jenkinsfile
- Click: **Save**

Build now.

Backend deployed: http://54.172.97.72:9090/springapp1

Frontend deployed: http://54.172.97.72:9090/frontapp1