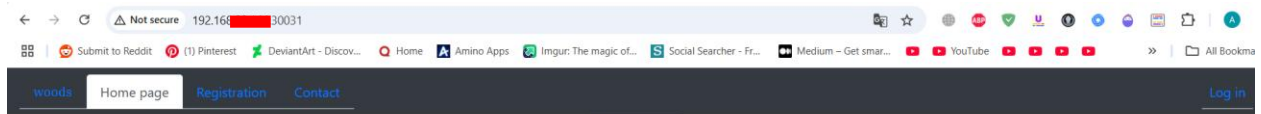
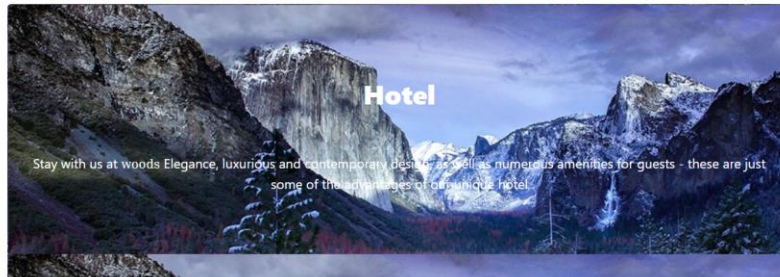


Spring boot Hotel (Docker & k8s)

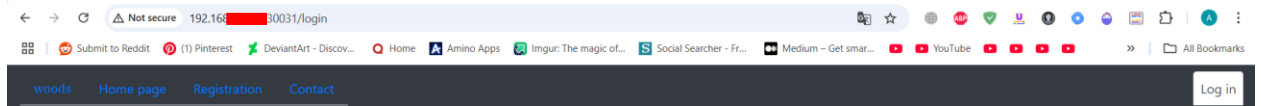


woods

A hotel located among trees, overlooking the mountains, where nature combines with modern style.



Activate Windows



Login

Login:

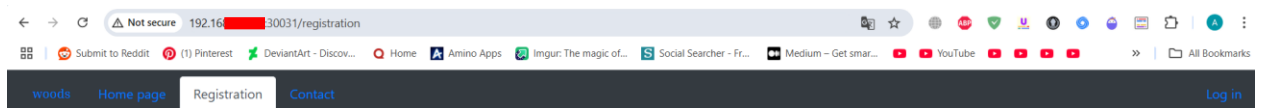
Username

Password:

Password

Log in

© 2019 Hotel woods



Please register

Login:

Name:

Last name:

Email:

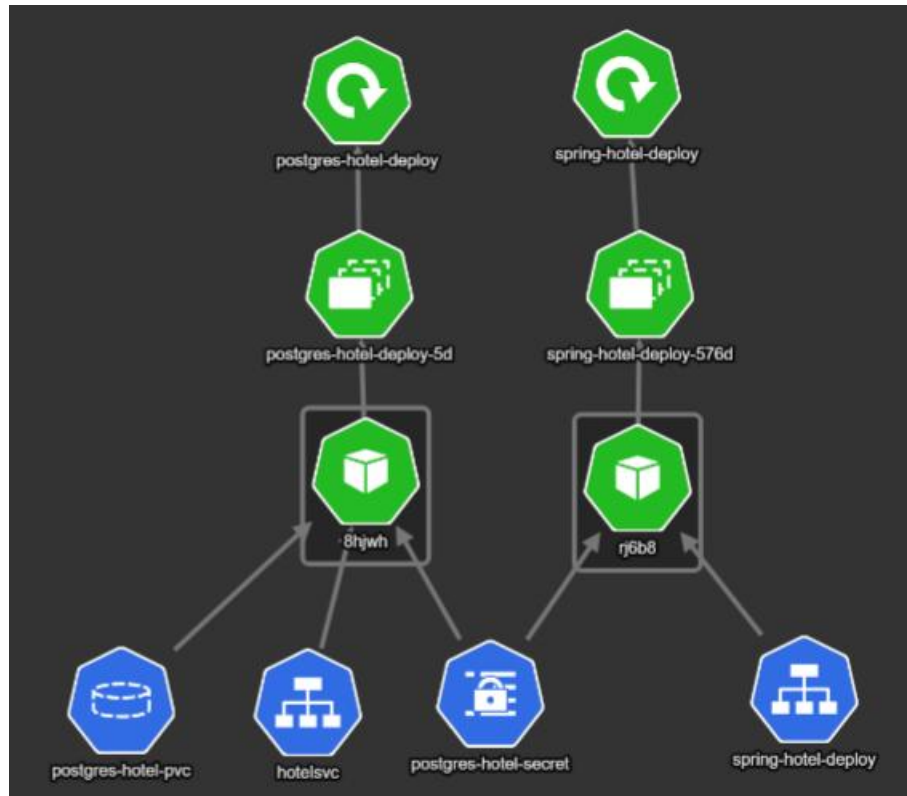
Phone number:

Password:

Register

Cancel

Activate Windows
Go to Settings to activate Windows.



```
FROM maven:3.8.4-openjdk-17-slim
```

```
WORKDIR /app
```

```
COPY mvnw .
```

```
#COPY .mvn .mvn
```

```
COPY pom.xml .
```

```
COPY src ./src
```

```
RUN mvn clean package -DskipTests
```

```
RUN ls -l target/
```

```
EXPOSE 8080
```

```
ENTRYPOINT ["java", "-jar", "target/demo-0.0.1-SNAPSHOT.jar"]
```

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: postgres-hotel-pv
spec:
  capacity:
    storage: 2Gi
  accessModes:
    - ReadWriteMany
  hostPath:
    path: "/adhamjenkins"
```

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: postgres-hotel-pvc
spec:
  volumeName: postgres-hotel-pv
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 2Gi
```

```
apiVersion: v1
kind: Secret
metadata:
  name: postgres-hotel-secret
type: Opaque
data:
  POSTGRES_PASSWORD: cGxhdG9u
```

```
apiVersion: v1
kind: Service
metadata:
  name: hotelsvc
spec:
  selector:
    hotel: postgres
  type: ClusterIP
  ports:
    - protocol: TCP
      port: 5432
      targetPort: 5432
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: spring-hotel-deploy
  labels:
    hotel: spring
spec:
  replicas: 1
  selector:
    matchLabels:
      hotel: spring
  template:
    metadata:
      name: pod-spring-hotel
      labels:
        hotel: spring
    spec:
      containers:
        - name: cont-spring-hotel
          image: adhmabdein/springhotel
          ports:
            - containerPort: 8080
          env:
            - name: SPRING_DATASOURCE_URL
              value: "jdbc:postgresql://hotelsvc:5432/HotelDB"
            - name: SPRING_DATASOURCE_USERNAME
              value: "postgres"
            - name: SPRING_DATASOURCE_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: postgres-hotel-secret
                  key: POSTGRES_PASSWORD
```


```
apiVersion: v1
kind: Service
metadata:
  name: spring-hotel-deploy
spec:
  selector:
    hotel: spring
  type: NodePort
  ports:
    - protocol: TCP
      port: 8080
      targetPort: 8080
      nodePort: 30031
```

```
spring.datasource.driverClassName=org.postgresql.Driver
spring.datasource.url= jdbc:postgresql://hotelsvc:5432/HotelDB
spring.datasource.username=postgres
spring.datasource.password=platon
spring.jpa.hibernate.ddl-auto=update

spring.jpa.database-platform=org.hibernate.dialect.PostgreSQLDialect
spring.jpa.generate-ddl=true
spring.jpa.show-sql=true


spring.mvc.view.prefix: /
spring.mvc.view.suffix: .jsp
spring.messages.basename=validation

spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=slacktest185@gmail.com
spring.mail.password=Hotel123
spring.mail.properties.mail.smtp.starttls.enable=true
spring.mail.properties.mail.smtp.starttls.required=true
spring.mail.properties.mail.smtp.auth=true;
```


spring-hotel
Public
Pin
Unwatch
1

master
1 Branch
0 Tags

Add file
Code

 root	initial commit	7d9efab · 29 minutes ago	🕒 94 Commits
src	initial commit	29 minutes ago	
.gitignore	[init]	6 years ago	
Dockerfile	initial commit	1 hour ago	
k8s-postgres-deploy.yml	initial commit	2 hours ago	
k8s-postgres-pv.yml	initial commit	3 hours ago	
k8s-postgres-pvc.yml	initial commit	3 hours ago	
k8s-postgres-secret.yml	initial commit	3 hours ago	
k8s-postgres-svc.yml	initial commit	2 hours ago	
k8s-spring-hotel-deploy.yml	initial commit	3 hours ago	
k8s-spring-hotel-svc.yml	initial commit	3 hours ago	
mvnw	[init]	6 years ago	
mvnw.cmd	[init]	6 years ago	
pom.xml	[add] mail dependency	6 years ago	

```
pipeline{
  agent {label 'node2'}

  environment{
    docker_img_name = 'adhmabdein/springhotel'
  }

  stages{
    stage('checkout'){
      steps{
        script{
          git "https://github.com/AdhmAbdein/spring-hotel.git"
        }
      }
    }
    stage('log in 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 Dockerfile .'
        }
      }
    }
  }
}
```

```

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

      sh 'kubectl apply -f k8s-spring-hotel-deploy.yml'
      sh 'kubectl apply -f k8s-spring-hotel-svc.yml'
    }
  }
}
}

```



spring-hotel

28 min

#17

2 hr 8 min

#11

11 min



adham@node2:~\$ kubectl get po

NAME	READY	STATUS	RESTARTS	AGE
mysql-crud-deploy-68558dbc74-h75b8	1/1	Running	2 (9m57s ago)	119m
mysql-dream-deploy-5c8c79bc88-69587	1/1	Running	2 (9m57s ago)	119m
mysql-spring-deploy-7f4886486c-f25fh	1/1	Running	2 (9m57s ago)	119m
mysql-spring-deploy-7f4886486c-z2tvq	1/1	Running	2 (9m57s ago)	119m
postgres-hotel-deploy-5d96574987-8hjwh	1/1	Running	1 (9m57s ago)	20m
spring-app-56b4845f99-2kbnz	1/1	Running	2 (9m58s ago)	119m
spring-crud-deploy-5fb7bdd99-8nchh	1/1	Running	19 (9m57s ago)	119m
spring-dream-deploy-65676b8946-lnvm2	1/1	Running	2 (9m57s ago)	119m
spring-hotel-deploy-576d997f45-rj6b8	1/1	Running	1 (9m58s ago)	20m

adham@node2:~\$ kubectl get deploy

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
mysql-crud-deploy	1/1	1	1	2d1h
mysql-dream-deploy	1/1	1	1	3d2h
mysql-spring-deploy	2/2	2	2	5d1h
postgres-hotel-deploy	1/1	1	1	20m
spring-app	1/1	1	1	5d1h
spring-crud-deploy	1/1	1	1	2d1h
spring-dream-deploy	1/1	1	1	3d2h
spring-hotel-deploy	1/1	1	1	20m


```

adham@node2:~$ kubectl get svc
NAME                TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
crudsvc             ClusterIP   10.96.1.1        <none>           3306/TCP         2d2h
hotelsvc            ClusterIP   10.99.1.1        <none>           5432/TCP         128m
k8s-dream-shop-svc ClusterIP   10.110.1.1       <none>           3306/TCP         3d4h
kubernetes          ClusterIP   10.96.1.1        <none>           443/TCP          5d23h
movie-svc           NodePort    10.100.1.1       <none>           8080:30013/TCP   3d9h
mysql-service-movie ClusterIP   10.108.1.1       <none>           3306/TCP         3d9h
mysqlservice        ClusterIP   10.110.1.1       <none>           3306/TCP         5d2h
spring-crud-svc     NodePort    10.106.1.1       <none>           8080:30030/TCP   2d2h
spring-dream-svc    NodePort    10.100.1.1       <none>           9191:30015/TCP   3d4h
spring-hotel-deploy NodePort    10.107.1.1       <none>           8080:30031/TCP   128m
adham@node2:~$ kubectl get pv
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                                STORAGECLASS
mysql-crud-pv       2Gi        RWX            Retain            Bound    default/mysql-crud-pvc              <unset>
mysql-dream-pv      2Gi        RWX            Retain            Bound    default/mysql-dream-pvc             <unset>
mysql-movie-pv      2Gi        RWX            Retain            Bound    default/mysql-movie-pvc             <unset>
mysql-spring-pv     2Gi        RWX            Retain            Bound    default/mysql-spring-pvc            <unset>
postgres-hotel-pv   2Gi        RWX            Retain            Bound    default/postgres-hotel-pvc          <unset>
adham@node2:~$ kubectl get pvc
NAME                STATUS      VOLUME          CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTES
mysql-crud-pvc      Bound      mysql-crud-pv    2Gi        RWX            <unset>
mysql-dream-pvc     Bound      mysql-dream-pv   2Gi        RWX            <unset>
mysql-movie-pvc     Bound      mysql-movie-pv   2Gi        RWX            <unset>
mysql-spring-pvc    Bound      mysql-spring-pv  2Gi        RWX            <unset>
postgres-hotel-pvc  Bound      postgres-hotel-pv 2Gi        RWX            <unset>
adham@node2:~$ kubectl get secret
NAME                TYPE      DATA   AGE
mysql-crud-secret   Opaque    1       2d2h
mysql-dream-secret  Opaque    1       3d4h
mysql-movie-secret  Opaque    1       3d23h
postgres-hotel-secret Opaque    1       138m

```

```
adham@node2:~$ kubectl exec -it postgres-hotel-deploy-5d96574987-8hjwh -- /bin/sh
# psql -U postgres -d HotelDB
psql (17.2 (Debian 17.2-1.pgdg120+1))
Type "help" for help.
```

```
HotelDB=# \dt
```

List of relations			
Schema	Name	Type	Owner
public	authorities	table	postgres
public	room_reservations	table	postgres
public	room_types	table	postgres
public	rooms	table	postgres
public	users	table	postgres
public	users_authorities	table	postgres

(6 rows)