Create 3-tier microservice (frontend/backend/database)

STEP 1: Create 3-tier microservice (frontend/backend/database) by oc new-app .

Command: oc new-app –name=_____ --image =_____ -n _____

```
[student@workstation ~]$ oc get pods -n net-sec-lab
NAME                        READY   STATUS    RESTARTS   AGE
backend-8549f8c9d7-882wb    1/1     Running   0          22m
database-6798974d77-mhs7c   1/1     Running   0          22m
frontend-7594b89b45-6mlz8   1/1     Running   0          49m
```

STEP 2: Expose frontend via Route. oc expose service _____

```
[student@workstation ~]$ oc apply -f frontend-route.yaml
route.route.openshift.io/frontend created
[student@workstation ~]$ oc get route
NAME       HOST/PORT                                PATH   SERVICES   PORT
 TERMINATION   WILDCARD
frontend   frontend-net-sec-lab.apps.ocp4.example.com          frontend   8080
             None
[student@workstation ~]$ vim frontend-edge.yaml
```

STEP 3: Secure traffic using NetworkPolicies Control traffic at pod level

```
[student@workstation ~]$ vim scc.yaml
[student@workstation ~]$ oc apply -f scc.yaml
securitycontextconstraints.security.openshift.io/nonroot-custom created
[student@workstation ~]$ oc create sa custom-sa -n net-sec-lab
serviceaccount/custom-sa created
[student@workstation ~]$ oc adm policy add-scc-to-user nonroot-custom -z custom
```

STEP 4: Enforce non-root execution via SCC . What actions pod can perform and under what conditions they are allowed to run.

```
student@workstation ~]$ oc adm policy add-scc-to-user nonroot-custom -z custom-
a -n net-sec-lab
lusterrole.rbac.authorization.k8s.io/system:openshift:scc:nonroot-custom added:
"custom-sa"
student@workstation ~]$ vim secure-pod.yaml
```

STEP 5: Create RBAC to restrict developer permissions. (who can do what)

```
[student@workstation ~]$ oc create rolebinding pod-reader-binding   --role=pod-r
eader   --user=testuser  --password=testuser  -n net-sec-lab
olebinding.rbac.authorization.k8s.io/pod-reader-binding created
[student@workstation ~]$ oc login -u testuser
```

STEP 6: Add Ingress for Kubernetes compatibility (for traffic Control)

```
[student@workstation ~]$ vim ingress.yaml
[student@workstation ~]$ oc apply -f ingress.yaml
ngress.networking.k8s.io/frontend-ingress created
[student@workstation ~]$ vim role.yaml
[student@workstation ~]$ oc apply -f role.yaml
```