Exercise 1:

Task 1: Using docker commands create a volume named ex1_vol.

docker volume creat ex1_vol.

```
jerryzzr — -zsh — 80×24
Last login: Sun Oct  8 14:21:43 on ttys001
(base) jerryzzr@JerryZzrdeMacBook-Pro ~ % docker volume create ex1_vol.
ex1_vol.
(base) jerryzzr@JerryZzrdeMacBook-Pro ~ %
```

Task2: With docker run start an ubuntu:22.04 container named sender. Additionally, mount to the container the \data to the volume we created and make sure to start your container in interactive mode.

docker run -it –-name sender -v ex1_vol.:/data ubuntu:22.04

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ~ % docker run -it --name sender -v ex1_vo
l.:/data ubuntu:22.04
```

Task3: Create a file with content in sender container. Then run receiver container using the same image to read the file I create before.

```
root@890fdace2ca9:/# cd /data
root@890fdace2ca9:/data# echo "zzr" > myname.txt
root@890fdace2ca9:/data# exit
exit

(base) jerryzzr@dhcp-892bf583 ~ % docker run -it --name receiver -v ex1_vol.:/da
ta:ro ubuntu:22.04
root@639f580f5188:/# cd  /data
root@639f580f5188:/data# cat myname.txt
zzr
```

Exercise2:

Create a docker file which use to run execute the install.sh script file.

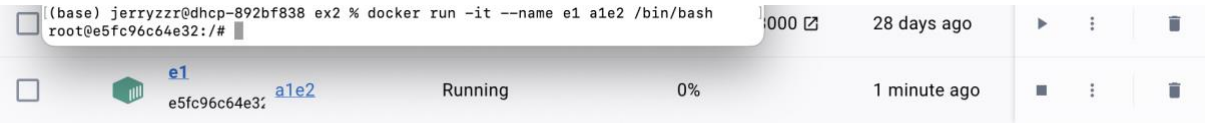
Which use a base image ubuntu and copy the install.sh file in the / directory of the container.

Then we add executable permission in the install.sh file

```
Dockerfile
FROM ubuntu:latest
COPY install.sh /install.sh
RUN chmod +x /install.sh
RUN /install.sh
```

Function 1 with bin bash function to run:

docker run -it --name e1 a1e2 /bin/bash



Function 2 with added CMD in dockerfile :

Dockerfile

```
FROM ubuntu:latest
COPY install.sh /install.sh
RUN chmod +x /install.sh
RUN /install.sh
CMD ["sh"]
```

docker run -it --name e1 a1

Exercise 3:

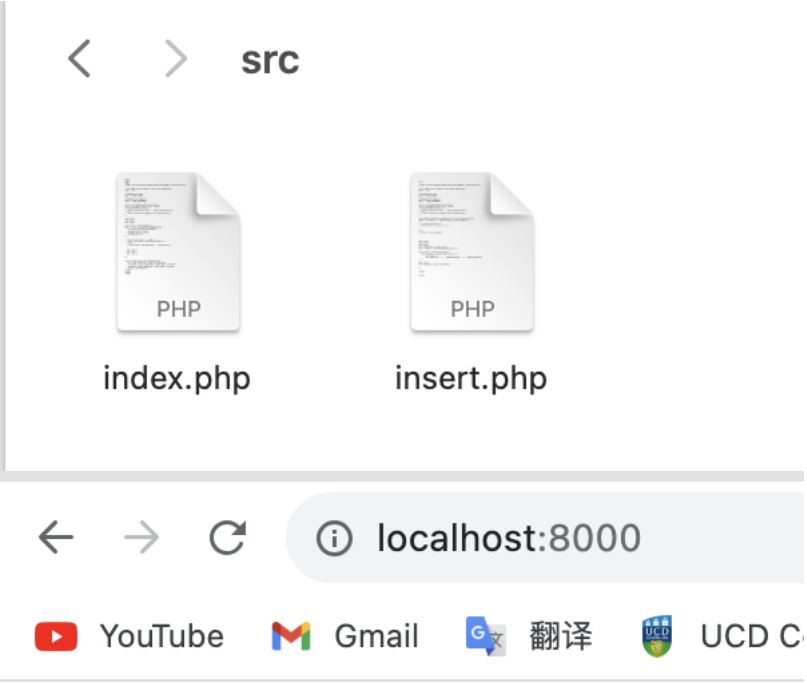
docker-compose 〉 No Selection

```yaml
version: '3.9'
services:
    php-apache-environment:
        container_name: php-apache
        build:
            context: .
            dockerfile: Dockerfile
        depends_on:
            - db
        volumes:
            - ./php/src:/var/www/html/
        ports:
            - 8000:80
    db:
        image: "mysql:latest"
        container_name: db
        restart: always
        ports:
            - 9906:3306
        environment:
            MYSQL_ROOT_PASSWORD: MYSQL_ROOT_PASSWORD
            MYSQL_DATABASE: MYSQL_DATABASE
            MYSQL_USER: MYSQL_USER
            MYSQL_PASSWORD: MYSQL_PASSWORD
    phpmyadmin:
        image: phpmyadmin/phpmyadmin
        container_name: phpmyadmin
        restart: always
        ports:
            - 8008:80
        depends_on:
            - db
        environment:
            PMA_HOST: db
```

```
(base) jerryzzr@dhcp-892bf838 source code % cd ex3
(base) jerryzzr@dhcp-892bf838 ex3 % docker compose up
```

php

now we got a folder called php, we should put these two file into php folder source



Connected to MySQL server successfully!

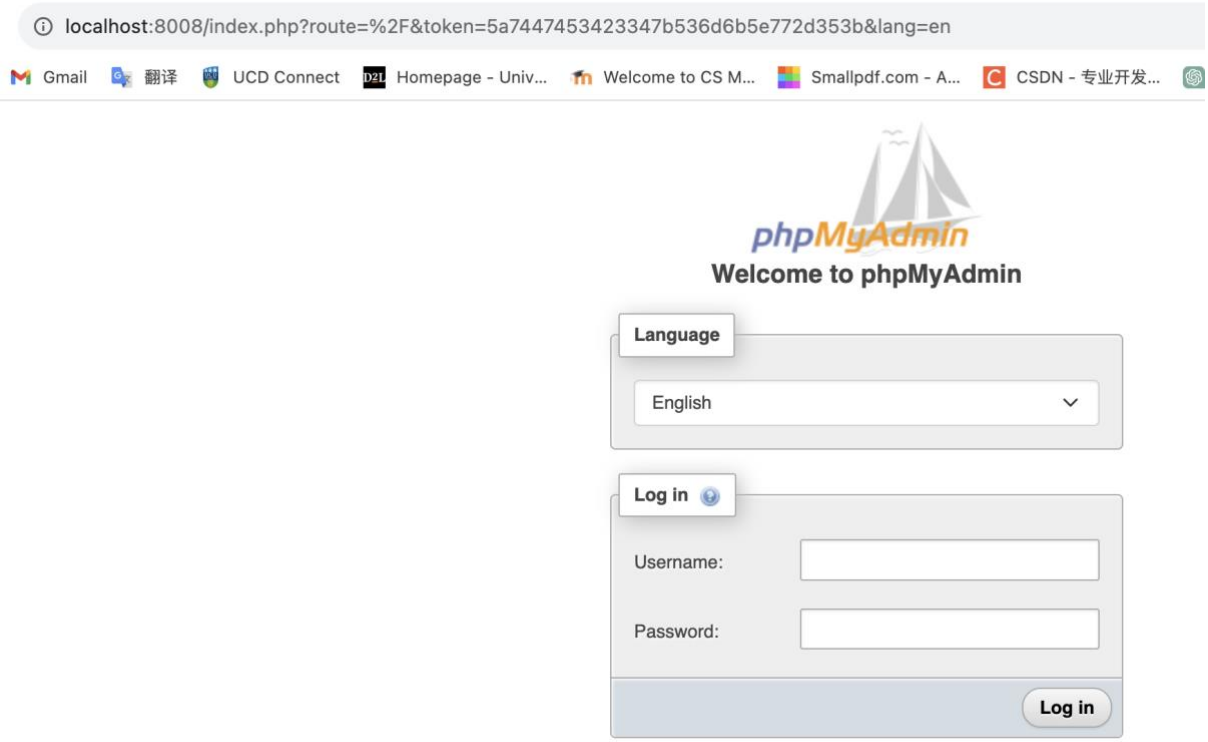
Table users created successfully

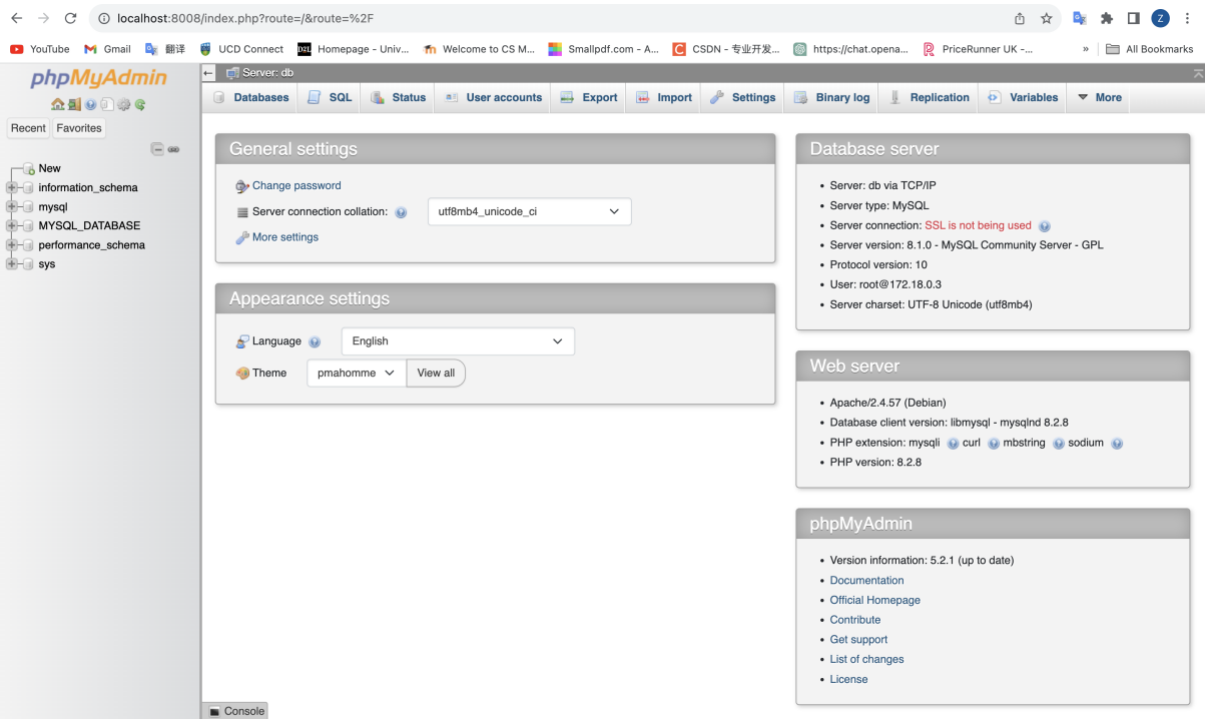
ID:

Firstname:

password:

Submit





Exercise 4:

First step:

Start a new Minikube cluster with two nodes and use cluster name ex4

minikube start –-nodes 2 -p ex4

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % minikube start --nodes 2 -p ex4
😄  [ex4] minikube v1.31.2 on Darwin 13.5.1
✨  Using the docker driver based on existing profile
👍  Starting control plane node ex4 in cluster ex4
🚜  Pulling base image ...
🤦  docker "ex4" container is missing, will recreate.
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.27.4 on Docker 24.0.4 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...-\ ^R
| ^[- ^[- ^R
^[[A^- ^R
^[[A| ^R
^[
    ▪ Configuring RBAC rules ...
🔗  Configuring CNI (Container Networking Interface) ...
🌟  Enabled addons:
🔎  Verifying Kubernetes components...
❗  The cluster ex4 already exists which means the --nodes parameter will be ignored. Use "minikube node add" to ad
d nodes to an existing cluster.
👍  Starting worker node ex4-m02 in cluster ex4
🚜  Pulling base image ...
🤦  docker "ex4-m02" container is missing, will recreate.
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🌐  Found network options:
    ▪ NO_PROXY=192.168.58.2
🐳  Preparing Kubernetes v1.27.4 on Docker 24.0.4 ...
    ▪ env NO_PROXY=192.168.58.2
E1019 20:51:30.679949    5889 node.go:121] unable to delete node "m02": nodes "ex4-m02" not found
E1019 20:51:30.679997    5889 start.go:316] error removing existing worker node before rejoining cluster, will cont
inue anyway: nodes "ex4-m02" not found
🔎  Verifying Kubernetes components...
🏄  Done! kubectl is now configured to use "ex4" cluster and "default" namespace by default
```

Then we check the nodes already created or not.

Kubectl get nodes

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl get nodes
NAME      STATUS   ROLES           AGE     VERSION
ex4       Ready    control-plane   6m23s   v1.27.4
ex4-m02   Ready    <none>          119s    v1.27.4
```

Task1:

Inside folder ex4 there is a yaml file called hello-department,now we should deploy it.

Create deployment:

Kubectl create -f hello-deployment.yaml

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl create -f hello-deployment.yaml
deployment.apps/hello created
```

Task 3:

(a)

Show all running pods firstly:

Kubectl get pods -o wide

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl get pods -o wide
NAME                     READY   STATUS    RESTARTS   AGE   IP           NODE      NOMINATED NODE   READINESS GATES
hello-77c947d946-5v4hm   1/1     Running   0          17s   10.244.1.2   ex4-m02   <none>           <none>
hello-77c947d946-sj7ds   1/1     Running   0          17s   10.244.0.3   ex4       <none>           <none>
```

Delete the deployment:

Kubectl delete deployment hello

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl delete deployment hello
deployment.apps "hello" deleted
```

Make sure no pods running now:

Kubectl get pods

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl get pods -o wide
No resources found in default namespace.
```

(b)

Now we set the label to exer4 :

Kubectl label nodes ex4 disktype=exer4

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl label nodes ex4 disktype=exer4
node/ex4 labeled
```

(c)

Kubectl get nodes –show-labels

```
(base) jerryzzr@dhcp-892bf838 ex4 % kubectl get nodes --show-labels
NAME       STATUS   ROLES    AGE   VERSION   LABELS
minikube   Ready    <none>   22m   v1.27.4   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,disktype=exer4,kubernetes.io/arch=amd64,kubernetes.io/hostname=minikube,kubernetes.io/os=linux
```

(d)

Now we create a new yaml file called hello-deployment_updated.yaml, and we set the label to this file. I add the nodeaffinity here and with nodeselectorterms to match the key **disktype** and the label **exer4** which I create in last task. Then I deploy it again and repeat the follows steps.

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: hello-updated
spec:
  replicas: 2
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxUnavailable: 100%
  selector:
    matchLabels:
      app: hello
  template:
    metadata:
      labels:
        app: hello
    spec:
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
              - matchExpressions:
                - key: disktype
                  operator: In
                  values:
                    - exer4
      containers:
        - name: hello-from
          image: pbitty/hello-from:latest
          ports:
            - name: http
              containerPort: 80
      terminationGracePeriodSeconds: 1
```

```
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl create -f hello-deployment_updated.yaml
deployment.apps/hello-updated created
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl get nodes
NAME      STATUS   ROLES           AGE     VERSION
ex4       Ready    control-plane   10m     v1.27.4
ex4-m02   Ready    <none>          5m55s   v1.27.4
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl get pods -o wide
NAME                              READY   STATUS    RESTARTS   AGE   IP           NODE   NOMINATED NODE   READINESS
GATES
hello-updated-7b9b44d7c6-mfswl    1/1     Running   0          14s   10.244.0.4   ex4    <none>           <none>
hello-updated-7b9b44d7c6-vqs9d    1/1     Running   0          14s   10.244.0.5   ex4    <none>           <none>
(base) jerryzzr@JerryZzrdeMacBook-Pro ex4 % kubectl delete deployment hello-updated
deployment.apps "hello-updated" deleted
```