

[Dashbo...](#) / [My cour...](#) / [Computer Engineering...](#) / [CEIT-Odd-sem-22...](#) / [Devops-odd-sem-22...](#) / [10 October - 16 Octo...](#) / [Midsem Ex...](#)

Started on	Friday, 14 October 2022, 12:04 PM
State	Finished
Completed on	Friday, 14 October 2022, 1:49 PM
Time taken	1 hour 45 mins
Grade	32.29 out of 40.00 (80.73%)

Question 1

Complete

Mark 1.50 out of 1.50

Which of the following statements are true about Kube API Server?

- ☐ a. kube-apiserver deploys a pod on a node
- ☒ b. Each component of kubernetes interacts with the kube-apiserver
- ☒ c. kube-apiserver updates the data in 'etcd' for all pods
- ☒ d. pod objects are created by kube-apiserver, but without assigning to a node
- ☐ e. kube-apiserver schedules a pod on a node
- ☒ f. Non kubernetes applications can also connect the kube-apiserver using HTTP protocol

The correct answers are: Each component of kubernetes interacts with the kube-apiserver, Non kubernetes applications can also connect the kube-apiserver using HTTP protocol, pod objects are created by kube-apiserver, but without assigning to a node, kube-apiserver updates the data in 'etcd' for all pods

Question 2

Complete

Mark 2.00 out of 2.00

Match the actions with each Kubernetes component, for the purpose of creation of a pod

register a node with cluster	kubelet
application image is deployed	container-runtime
send reports of pods regularly to kube-apiserver	kubelet
node to run the pod is identified	kube-scheduler
request docker to run the instance	kubelet
authentication request is validated	kube-apiserver
pod object is created	kube-apiserver
pod is created on the node	kubelet
detect that new pod object is created	kube-scheduler
authentication request is generated	kubectl
pod info is updated in etcd	kube-apiserver

The correct answer is: register a node with cluster → kubelet, application image is deployed → container-runtime, send reports of pods regularly to kube-apiserver → kubelet, node to run the pod is identified → kube-scheduler, request docker to run the instance → kubelet, authentication request is validated → kube-apiserver, pod object is created → kube-apiserver, pod is created on the node → kubelet, detect that new pod object is created → kube-scheduler, authentication request is generated → kubectl, pod info is updated in etcd → kube-apiserver

Question 3

Complete

Mark 1.00 out of 2.00

Write a Dockerfile that does the following

- (a) uses a Ubuntu base image
- (b) Add the binary "hello" available in current directory to the /usr/local/bin/ path in the image
- (c) Installs the tree command also
- (d) Installs apache in the image
- (e) Runs apache on port 3030

From ubuntu

run cp hello /usr/local/bin/

run apt update

run apt install tree

run aptinstall apache2

Comment:

Question 4

Complete

Mark 2.00 out of 2.00

Suppose it is required to create a custom ubuntu docker image, saved as a tar file, with the base ubuntu and the package "vim" into it. Complete the following description of commands, which aim to achieve the above aim.

On Host

docker run -it
ubuntu

On container prompt

apt
update

apt
install

vim

On Host, while container is running

docker
images

get list of images. Output shown below.

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
<none>	<none>	55fb3b7c6802	5 seconds ago	116MB
my-fedora	latest	1862a1e17ccb	3 days ago	434MB
capitalserver	latest	e4c2e09a1ef7	3 weeks ago	124MB

docker
tag
55fb3b7c6802

ubuntu-vim
docker save
ubuntu-vim

-o /tmp/ubuntu-vim.tar

Question 5

Complete

Mark 1.14 out of 2.00

JSON files were extracted from the tar image of a docker.

The files are listed below:

File1

```
{
  "architecture": "amd64",
  "config": {
    "Hostname": "896ed4258c84",
    "Domainname": "",
    "User": "",
    "AttachStdin": true,
    "AttachStdout": true,
    "AttachStderr": true,
    "Tty": true,
    "OpenStdin": true,
    "StdinOnce": true,
    "Env": [
      "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
    ],
    "Cmd": [
      "bash"
    ],
    "Image": "ubuntu",
    "Volumes": null,
    "WorkingDir": "",
    "Entrypoint": null,
    "OnBuild": null,
    "Labels": {}
  },
  "container": "896ed4258c8422489a183fc1a0f7d110e4e5c432fc5d62971b24015d58620701",
  "container_config": {
    "Hostname": "896ed4258c84",
    "Domainname": "",
    "User": "",
    "AttachStdin": true,
    "AttachStdout": true,
    "AttachStderr": true,
    "Tty": true,
    "OpenStdin": true,
    "StdinOnce": true,
    "Env": [
      "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
    ],
    "Cmd": [
      "bash"
    ],
    "Image": "ubuntu",
    "Volumes": null,
    "WorkingDir": "",
    "Entrypoint": null,
    "OnBuild": null,
    "Labels": {}
  },
  "created": "2022-09-21T15:31:32.742756984Z",
  "docker_version": "20.10.18",
  "history": [
    {
      "created": "2022-09-01T23:46:35.026691064Z",
      "created_by": "/bin/sh -c #(nop) ADD file:a7268f82a86219801950401c224cabbdd83ef510a7c71396b25f70c2639ae4fa in /"
```

```

    },
    {
      "created": "2022-09-01T23:46:35.375057619Z",
      "created_by": "/bin/sh -c #(nop) CMD [\"bash\"]",
      "empty_layer": true
    },
    {
      "created": "2022-09-21T15:31:32.742756984Z",
      "created_by": "bash"
    }
  ],
  "os": "linux",
  "rootfs": {
    "type": "layers",
    "diff_ids": [
      "sha256:7f5cbd8cc787c8d628630756bcc7240e6c96b876c2882e6fc980a8b60cdfa274",
      "sha256:0a7af9a8e086cfd665e913c752013f6c918fc132485e3b152b538179f0314467"
    ]
  }
}

```

file2:

```

{
  "id": "03c159c5eead07601110c82ad724cfe7b85eb6c9c9a91e44775338053e8e988a",
  "created": "1970-01-01T05:30:00+05:30",
  "container_config": {
    "Hostname": "",
    "Domainname": "",
    "User": "",
    "AttachStdin": false,
    "AttachStdout": false,
    "AttachStderr": false,
    "Tty": false,
    "OpenStdin": false,
    "StdinOnce": false,
    "Env": null,
    "Cmd": null,
    "Image": "",
    "Volumes": null,
    "WorkingDir": "",
    "Entrypoint": null,
    "OnBuild": null,
    "Labels": null
  },
  "os": "linux"
}

```

File3:

```

{
  "id": "e81f28d0db59ae988d99e9484ef94debd8b84cc4afe9159a4816aeb0febdd1cd",
  "parent": "03c159c5eead07601110c82ad724cfe7b85eb6c9c9a91e44775338053e8e988a",
  "created": "2022-09-21T15:31:32.742756984Z",
  "container": "896ed4258c8422489a183fc1a0f7d110e4e5c432fc5d62971b24015d58620701",
  "container_config": {
    "Hostname": "896ed4258c84",
    "Domainname": "",
    "User": "",
    "AttachStdin": true,
    "AttachStdout": true,
    "AttachStderr": true,
    "Tty": true,
    "OpenStdin": true,
    "StdinOnce": true,
  }
}

```

```

"Env": [
  "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
],
"Cmd": [
  "bash"
],
"Image": "ubuntu",
"Volumes": null,
"WorkingDir": "",
"Entrypoint": null,
"OnBuild": null,
"Labels": {}
},
"docker_version": "20.10.18",
"config": {
  "Hostname": "896ed4258c84",
  "Domainname": "",
  "User": "",
  "AttachStdin": true,
  "AttachStdout": true,
  "AttachStderr": true,
  "Tty": true,
  "OpenStdin": true,
  "StdinOnce": true,
  "Env": [
    "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
  ],
  "Cmd": [
    "bash"
  ],
  "Image": "ubuntu",
  "Volumes": null,
  "WorkingDir": "",
  "Entrypoint": null,
  "OnBuild": null,
  "Labels": {}
},
"architecture": "amd64",
"os": "linux"
}

```

file4:

```

[
  {
    "Config": "aef552620a0f9b00184fd236648d3ec745e50126a6ac82a0117b3ed517296280.json",
    "RepoTags": [
      "my-ubuntu-tar:latest"
    ],
    "Layers": [
      "03c159c5eead07601110c82ad724cfe7b85eb6c9c9a91e44775338053e8e988a/layer.tar",
      "e81f28d0db59ae988d99e9484ef94debd8b84cc4afe9159a4816aeb0febdd1cd/layer.tar"
    ]
  }
]

```

Mark those statements as True which can be definitely deduced from this information. Wrong/incomplete deductions should be marked as False.

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	There is at least one layer added on top of base image

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	The "id" of base image is 03c159c5eead07601110c82ad724cfe7b85eb6c9c9a91e44775338053e8e988a
<input type="radio"/>	<input checked="" type="radio"/>	The "id" of base image is 896ed4258c8422489a183fc1a0f7d110e4e5c432fc5d62971b24015d58620701
<input checked="" type="radio"/>	<input type="radio"/>	The base image is fedora
<input type="radio"/>	<input checked="" type="radio"/>	The "id" of base image is aef552620a0f9b00184fd236648d3ec745e50126a6ac82a0117b3ed517296280
<input type="radio"/>	<input checked="" type="radio"/>	The base image is ubuntu
<input type="radio"/>	<input type="radio"/>	The image contains the command "tar" in it.

There is at least one layer added on top of base image: True

The "id" of base image is 03c159c5eead07601110c82ad724cfe7b85eb6c9c9a91e44775338053e8e988a: True

The "id" of base image is 896ed4258c8422489a183fc1a0f7d110e4e5c432fc5d62971b24015d58620701: False

The base image is fedora: False

The "id" of base image is aef552620a0f9b00184fd236648d3ec745e50126a6ac82a0117b3ed517296280: False

The base image is ubuntu: True

The image contains the command "tar" in it.: False

Question 6

Complete

Mark 1.00 out of 1.00

Given below are few statements differentiating between and comparing containers and virtual machines.

For each statement, mark True/False.

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	Example of container runtime is "dockerd", and an example of hypervisor is "kvm"
<input checked="" type="radio"/>	<input type="radio"/>	Virtual Machine runs its own kernel, but Container uses the kernel of the Host operating system
<input type="radio"/>	<input checked="" type="radio"/>	Example of container runtime is "docker", and an example of hypervisor is "secureboot"
<input checked="" type="radio"/>	<input type="radio"/>	Virtual Machines need more storage compared to containers doing the same job
<input type="radio"/>	<input checked="" type="radio"/>	Container runtime and hypervisor do the same job

Example of container runtime is "dockerd", and an example of hypervisor is "kvm": True

Virtual Machine runs its own kernel, but Container uses the kernel of the Host operating system: True

Example of container runtime is "docker", and an example of hypervisor is "secureboot": False

Virtual Machines need more storage compared to containers doing the same job: True

Container runtime and hypervisor do the same job: False

Question 7

Complete

Mark 1.00 out of 1.00

Match the pairs

Ubuntu Machine in AWS	PaaS
Google Docs	SaaS
VM in Cloud	IaaS
Load Balancer	IaaS
Web Conferencing	SaaS

Your answer is correct.

The correct answer is: Ubuntu Machine in AWS → PaaS, Google Docs → SaaS, VM in Cloud → IaaS, Load Balancer → IaaS, Web Conferencing → SaaS

Question 8

Complete

Mark 1.00 out of 1.00

Select from RHS what is Unique only to the Cloud Service mentioned on the LHS

SaaS	Application
PaaS	Operating System
IaaS	Networking

Your answer is correct.

The correct answer is: SaaS → Application, PaaS → Operating System, IaaS → Networking

Question 9

Complete

Mark 1.00 out of 1.00

Given below is a list of possible "values" for the "Kind" tag in the kuernetes YAML syntax.

Select "Yes" if the said string can be used, and "No" if it is not a valid string.

(e.g. refer to the files like this

apiVersion: apps/v1

kind: Deployment

)

Yes	No	
<input checked="" type="radio"/>	<input type="radio"/>	ReplicaSet
<input checked="" type="radio"/>	<input type="radio"/>	PersistentVolume
<input checked="" type="radio"/>	<input type="radio"/>	PersistentVolumeClaim
<input checked="" type="radio"/>	<input type="radio"/>	Deployment
<input checked="" type="radio"/>	<input type="radio"/>	Service
<input checked="" type="radio"/>	<input type="radio"/>	Pod

ReplicaSet: Yes

PersistentVolume: Yes

PersistentVolumeClaim: Yes

Deployment: Yes

Service: Yes

Pod: Yes

Question **10**

Complete

Mark 2.00 out of 2.00

Consider the following sequence of commands

```
git init
```

```
vi a.c # file edited /* A */
```

```
git add a.c /* B */
```

```
vi a.c # file edited /* C */
```

```
git add a.c
```

```
git commit /* D */
```

```
vi a.c /* E */
```

```
git add a.c /* F */
```

```
git commit
```

For each of the lines, labled as A to F, select the proper option describing the state of the file a.c

E	<input type="text" value="modified"/>
D	<input type="text" value="unmodified"/>
A	<input type="text" value="untracked"/>
C	<input type="text" value="modified"/>
B	<input type="text" value="staged"/>
F	<input type="text" value="staged"/>

Your answer is correct.

The correct answer is: E → modified, D → unmodified, A → untracked, C → modified, B → staged, F → staged

Question **11**

Complete

Mark 1.00 out of 1.00

Math pairs

node affinity	<input type="text" value="A property of Pods, attracting it to specified nodes"/>
Taint	<input type="text" value="Nodes repelling a set of pods"/>
Toleration	<input type="text" value="Let the scheduler schedule a pod with matching taints"/>

Your answer is correct.

The correct answer is: node affinity → A property of Pods, attracting it to specified nodes, Taint → Nodes repelling a set of pods, Toleration → Let the scheduler schedule a pod with matching taints

Question **12**

Complete

Mark 1.60 out of 2.00

Which of the following images (for x86a and linux) are downloaded less than a billion times from docker-hub?

- ☐ a. golang
- ☒ b. elasticsearch
- ☐ c. nodejs
- ☐ d. redis
- ☐ e. rabbitmq
- ☒ f. haproxy
- ☐ g. busybox
- ☒ h. php
- ☒ i. tomcat
- ☐ j. httpd

The correct answers are: rabbitmq, elasticsearch, haproxy, php, tomcat

Question **13**

Complete

Mark 1.00 out of 1.50

Consider the following declaration in a YML file

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: myapp-ha
  labels:
    app: myapp
    type: front-end
spec:
  template:
    metadata:
      name: myapp
      labels:
        app: myapp
        type: frontend
    spec:
      containers:
        - name: container-1
          image: redis
        - name: container-2
          image: httpd
  replicas: 3
  selector:
    matchLabels:
      type: frontend
```

This configuration will

create

containers

create

pods

create a

Question **14**

Complete

Mark 1.00 out of 1.00

Why is "kubectl edit" not recommended, against "kubectl apply"?

- ☐ a. because it changes the pod when it is created next time
- ☐ b. because it does change a running pod, and it may crash the pod
- ☐ c. because it edits the YML file but does not apply it
- ☒ d. because it does change a running pod, and no record of changes is maintained
- ☐ e. because it changes a running pod

The correct answer is: because it does change a running pod, and no record of changes is maintained

Question **15**

Not answered

Marked out of 2.00

Write a shell program which does the following:

Takes two file-names as arguments on the command line.

Reads from the first file a pair of characters where the first is the "find" character and the second is the "replace" character.

Replaces all "find" characters in the second file with the "replace" character.

For example,

if the first file contains:

a m

[]

j t

Then the program will replace all "a" by "m" in the second file, all "[" by "]" in the second file, etc.

Question **16**

Complete

Mark 1.00 out of 1.00

Select all the statements that correctly identify the need, use, and limitations for containers.

- ☒ a. Containers can not be used where applications run close to the hardware
- ☒ b. Containers have helped in micro-services architecture.
- ☒ c. Containers offer more portability, efficiency
- ☒ d. Isolated sandbox environment helps in more security
- ☒ e. Managing library version dependency is a major concern, and containers ease this problem.
- ☒ f. A containerized application can be launched in minimal time.
- ☒ g. Containers offer more elasticity, reusability
- ☒ h. Changes in a shared library could break applications, but containers simplify upgrade and rollback.
- ☒ i. Isolated sandbox environment helps in more reliability and uptime
- ☒ j. Managing library version dependency is a major concern, and containers ease this problem.

Your answer is correct.

The correct answers are: Managing library version dependency is a major concern, and containers ease this problem., Managing library version dependency is a major concern, and containers ease this problem., Changes in a shared library could break applications, but containers simplify upgrade and rollback., Isolated sandbox environment helps in more security, Isolated sandbox environment helps in more reliability and uptime, Containers offer more portability, efficiency, Containers offer more elasticity, reusability, Containers can not be used where applications run close to the hardware, Containers have helped in micro-services architecture., A containerized application can be launched in minimal time.

Question **17**

Complete

Mark 0.80 out of 1.00

Select the correct statements, which describe why the clusterIP is used, how it is used and what it does.

- ☒ a. Using Pod's IP address leads to non-portability, as when Pod is deleted and respawned, it will have a different IP address
- ☒ b. ClusterIP service is reachable only within the cluster
- ☐ c. If ClusterIP goes down, then it's respawned but with same IP address, so the existing connections are maintained
- ☒ d. The ClusterIP exports the IP address to which the clients of the service attach
- ☒ e. ClusterIP service in Kubernetes is a REST object, similar to a Pod

Your answer is partially correct.

You have correctly selected 4.

The correct answers are: Using Pod's IP address leads to non-portability, as when Pod is deleted and respawned, it will have a different IP address, ClusterIP service in Kubernetes is a REST object, similar to a Pod, The ClusterIP exports the IP address to which the clients of the service attach, If ClusterIP goes down, then it's respawned but with same IP address, so the existing connections are maintained, ClusterIP service is reachable only within the cluster

Question **18**

Complete

Mark 2.00 out of 2.00

Which of the following are NOT features of kubernetes?

- ☒ a. written in C
- ☐ b. is open source
- ☒ c. supports only docker
- ☐ d. supports declarative configuration
- ☐ e. is extensible
- ☐ f. automates software deployment
- ☐ g. written in GO
- ☐ h. is portable

The correct answers are: supports only docker, written in C

Question **19**

Complete

Mark 2.00 out of 2.00

The command

```
docker run --rm -ti -v /root/data:/data:z --name fedora-1 fedora bash
```

```
docker run --rm -ti --volumes-from fedora-1 --name fedora-2 fedora
```

means

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	Two docker containers of the fedora image are created, namely fedora-1 and fedora-2
<input checked="" type="radio"/>	<input type="radio"/>	The /root/data folder on the Host is shared with the container(s)
<input checked="" type="radio"/>	<input type="radio"/>	The commands basically say that whatever is the volume for fedora1, should be shared with fedora2
<input type="radio"/>	<input checked="" type="radio"/>	Running <code>ls /var/lib/docker/volumes/root/data</code> on host, <code>ls /data</code> on fedora1 and <code>ls /data</code> on fedora2 will show same result
<input type="radio"/>	<input checked="" type="radio"/>	One docker container of the fedora image is created, namely fedora-1 , and fedora-2 is an alias for that instace
<input checked="" type="radio"/>	<input type="radio"/>	Running <code>ls /root/data</code> on host, <code>ls /data</code> on fedora1 and <code>ls /data</code> on fedora2 will show same result
<input type="radio"/>	<input checked="" type="radio"/>	The commands basically say that whatever is the volume for fedora1, should be copied ditto with fedora2

Two docker containers of the fedora image are created, namely fedora-1 and fedora-2: True

The /root/data folder on the Host is shared with the container(s): True

The commands basically say that whatever is the volume for fedora1, should be shared with fedora2: True

Running `ls /var/lib/docker/volumes/root/data` on host, `ls /data` on fedora1 and `ls /data` on fedora2 will show same result: False

One docker container of the fedora image is created, namely fedora-1 , and fedora-2 is an alias for that instace: False

Running `ls /root/data` on host, `ls /data` on fedora1 and `ls /data` on fedora2 will show same result: True

The commands basically say that whatever is the volume for fedora1, should be copied ditto with fedora2: False

Question **20**

Complete

Mark 0.50 out of 1.00

Select correct statements about the command kubectl

- ☐ a. runs on the kubernetes user's laptop
- ☒ b. talks to kube-apiserver
- ☒ c. in minikube installation the kubectl is already pre-configured to talk to kube-apiserver
- ☐ d. is used to manage pods on a single node cluster on your laptop

The correct answers are: runs on the kubernetes user's laptop, talks to kube-apiserver, in minikube installation the kubectl is already pre-configured to talk to kube-apiserver, is used to manage pods on a single node cluster on your laptop

Question **21**

Complete

Mark 1.50 out of 2.00

Select all the correct statements about branches in git

Select one or more:

- ☒ a. 'master' is just another branch
- ☐ b. creating a branch involves copy of the commits
- ☒ c. a branch is just a pointer in git data-structure
- ☒ d. one can work in parallel on different branches
- ☐ e. master branch is treated specially by git
- ☒ f. switching branches is a costly operation
- ☐ g. creation of a branch results in a commit
- ☒ h. creating a branch is cheap with git

Your answer is partially correct.

You have selected too many options.

The correct answers are: a branch is just a pointer in git data-structure, creating a branch is cheap with git, 'master' is just another branch, one can work in parallel on different branches

Question **22**

Complete

Mark 2.00 out of 2.00

Given below is the output of one command:

```
$ kubectl describe replicaset.apps nginx-deployment
```

```
Name:          nginx-deployment-6768c68f7b
Namespace:     default
Selector:      app=nginx,pod-template-hash=6768c68f7b
Labels:        app=nginx
                pod-template-hash=6768c68f7b
Annotations:   deployment.kubernetes.io/desired-replicas: 3
                deployment.kubernetes.io/max-replicas: 4
                deployment.kubernetes.io/revision: 2
Controlled By: Deployment/nginx-deployment
Replicas:      3 current / 3 desired
Pods Status:   3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
```

```
  Labels:  app=nginx
            pod-template-hash=6768c68f7b
```

```
Containers:
```

```
  nginx:
    Image:      nginx:1.20
    Port:       <none>
    Host Port:  <none>
    Environment: <none>
    Mounts:     <none>
  Volumes:     <none>
```

```
Events:
```

Type	Reason	Age	From	Message
Normal	SuccessfulCreate	3m7s	replicaset-controller	Created pod: nginx-deployment-6768c68f7b-sqqbf
Normal	SuccessfulCreate	2m43s	replicaset-controller	Created pod: nginx-deployment-6768c68f7b-xbtr8
Normal	SuccessfulCreate	2m41s	replicaset-controller	Created pod: nginx-deployment-6768c68f7b-pq2nt

```
Name:          nginx-deployment-77979d4865
Namespace:     default
Selector:      app=nginx,pod-template-hash=77979d4865
Labels:        app=nginx
                pod-template-hash=77979d4865
Annotations:   deployment.kubernetes.io/desired-replicas: 3
                deployment.kubernetes.io/max-replicas: 4
                deployment.kubernetes.io/revision: 1
Controlled By: Deployment/nginx-deployment
Replicas:      0 current / 0 desired
Pods Status:   0 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
```

```
  Labels:  app=nginx
            pod-template-hash=77979d4865
```

```
Containers:
```

```
  nginx:
    Image:      nginx:1.18
    Port:       <none>
    Host Port:  <none>
    Environment: <none>
    Mounts:     <none>
  Volumes:     <none>
```

Events:

Type	Reason	Age	From	Message
-----	-----	----	----	-----
Normal	SuccessfulCreate	3m51s	replicaset-controller	Created pod: nginx-deployment-77979d4865-jskrv
Normal	SuccessfulCreate	3m51s	replicaset-controller	Created pod: nginx-deployment-77979d4865-bmxj7
Normal	SuccessfulCreate	3m51s	replicaset-controller	Created pod: nginx-deployment-77979d4865-tfjrs
Normal	SuccessfulDelete	2m43s	replicaset-controller	Deleted pod: nginx-deployment-77979d4865-jskrv
Normal	SuccessfulDelete	2m41s	replicaset-controller	Deleted pod: nginx-deployment-77979d4865-bmxj7
Normal	SuccessfulDelete	2m39s	replicaset-controller	Deleted pod: nginx-deployment-77979d4865-tfjrs

Mark those statements as True, which can be deduced from the information given in the above output.

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	Pod has only one container, that is nginx in it
<input checked="" type="radio"/>	<input type="radio"/>	The latest version of nginx is running right now
<input type="radio"/>	<input checked="" type="radio"/>	The upgrade/downgrade process is in operation and not complete yet
<input type="radio"/>	<input checked="" type="radio"/>	There are 3 replica-sets
<input checked="" type="radio"/>	<input type="radio"/>	nginx was upgraded from 1.18 to 1.20
<input checked="" type="radio"/>	<input type="radio"/>	Number of replicas is 3
<input type="radio"/>	<input checked="" type="radio"/>	nginx was downgraded from 1.20 to 1.18

Pod has only one container, that is nginx in it: True

The latest version of nginx is running right now: True

The upgrade/downgrade process is in operation and not complete yet: False

There are 3 replica-sets: False

nginx was upgraded from 1.18 to 1.20: True

Number of replicas is 3: True

nginx was downgraded from 1.20 to 1.18: False

Question **23**

Complete

Mark 1.00 out of 2.00

Select all the incorrect statements about git merge and git **rebase**.

Select one or more:

- ☒ a. merge and **rebase** are just two names for the same thing
- ☐ b. git **rebase** may not work automatically and fail
- ☒ c. **rebase** is for newbees, and merge for experienced programmers.
- ☐ d. merge is costlier than a **rebase**
- ☐ e. **rebase** is costlier than a merge
- ☐ f. **rebase** can lead to very complicated scenarios in distributed development.
- ☐ g. git **rebase** creates a more linear history
- ☐ h. we can't use git unless we resolve a failed merge
- ☐ i. git merge may not work automatically and fail
- ☐ j. git merge necessarily creates a commit
- ☐ k. git merge creates a more non-linear history
- ☐ l. we can't use git unless we resolve a failed **rebase**

Your answer is partially correct.

You have correctly selected 2.

The correct answers are: merge and **rebase** are just two names for the same thing, merge is costlier than a **rebase**, **rebase** is costlier than a merge, **rebase** is for newbees, and merge for experienced programmers.

Question **24**

Complete

Mark 2.00 out of 2.00

Consider the following command and its output:

```
$ kubectl describe replicaset myapp-ha
Name:          myapp-ha
Namespace:     default
Selector:      type=frontend
Labels:        app=myapp
               type=front-end
Annotations:   <none>
Replicas:      3 current / 3 desired
Pods Status:   2 Running / 1 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=myapp
          type=frontend
Containers:
  container-1:
    Image:          redis
    Port:           <none>
    Host Port:      <none>
    Environment:    <none>
    Mounts:         <none>
  container-2:
    Image:          httpd
    Port:           <none>
    Host Port:      <none>
    Environment:    <none>
    Mounts:         <none>
Volumes:          <none>
Events:
  Type      Reason              Age   From                      Message
  ----      -
  Normal    SuccessfulCreate    24s   replicaset-controller     Created pod: myapp-ha-7kkzl
  Normal    SuccessfulCreate    24s   replicaset-controller     Created pod: myapp-ha-w7pnp
  Normal    SuccessfulCreate    24s   replicaset-controller     Created pod: myapp-ha-rsj4n
```

Followed by few commands, and in the end this output:

```
$ kubectl describe replicaset myapp-ha
Name:          myapp-ha
Namespace:     default
Selector:      type=frontend
Labels:        app=myapp
               type=front-end
Annotations:   <none>
Replicas:      3 current / 3 desired
Pods Status:   0 Running / 3 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=myapp
          type=frontend
Containers:
  container-1:
    Image:          redis
    Port:           <none>
    Host Port:      <none>
    Environment:    <none>
    Mounts:         <none>
  container-2:
```

Image: httpd
Port: <none>
Host Port: <none>
Environment: <none>
Mounts: <none>
Volumes: <none>

Events:

Type	Reason	Age	From	Message
----	-----	----	----	-----
Normal	SuccessfulCreate	93s	replicaset-controller	Created pod: myapp-ha-7kkz1
Normal	SuccessfulCreate	93s	replicaset-controller	Created pod: myapp-ha-w7pnp
Normal	SuccessfulCreate	93s	replicaset-controller	Created pod: myapp-ha-rsj4n
Normal	SuccessfulCreate	25s	replicaset-controller	Created pod: myapp-ha-dwg7r
Normal	SuccessfulCreate	5s	replicaset-controller	Created pod: myapp-ha-dt7nd
Normal	SuccessfulCreate	5s	replicaset-controller	Created pod: myapp-ha-plw49
Normal	SuccessfulCreate	5s	replicaset-controller	Created pod: myapp-ha-tzb9p

Here,

the number of pods defined in the replicaset are:

3

the number of pods deleted so far are :

4

the first re-created pod had the ID:

myapp-ha-dwg7r

At the time of the last command and its output, the number of pods waiting is 3. That means:

3 deleted pods are being re-created

Question **25**

Complete

Mark 0.50 out of 1.00

Consider following the sequence of commands executed either on the host or container or somewhere so that all of them together make some sense.

```
docker run -it ubuntu
apt update; apt install net-tools iputils-ping openssh-server openssh-client
docker commit <container-id> myubuntu
exit

docker network create network1
docker run -it -v /tmp/folder:/folder --network network1 --hostname u1 myubuntu
echo 1 >> /folder/1
docker run -it -v /tmp/folder:/folder --network network1 --hostname u2 myubuntu
echo 2 >> /folder/1
docker run -it -v /tmp/folder:/folder --network network1 --hostname u3 myubuntu
echo 3 >> /folder/1
```

Mark statements as True/False w.r.t. above commands.

True	False	
<input checked="" type="radio"/>	<input type="radio"/>	ssh from u1 to u2 will work now.
<input checked="" type="radio"/>	<input type="radio"/>	The openssh-server installed in the image has helped the containers communicate with each other
<input type="radio"/>	<input checked="" type="radio"/>	The creation of a network does not serve any purpose in the echo commands
<input checked="" type="radio"/>	<input type="radio"/>	The file /folder/1 on container u1 finally contains the data "1\n2\n3\n"
<input checked="" type="radio"/>	<input type="radio"/>	u1 u2 u3 will be able to ping each other
<input checked="" type="radio"/>	<input type="radio"/>	The file /tmp/folder/1 on host finally contains the data "1\n2\n3\n"

ssh from u1 to u2 will work now.: False

The openssh-server installed in the image has helped the containers communicate with each other: False

The creation of a network does not serve any purpose in the echo commands: True

The file /folder/1 on container u1 finally contains the data "1\n2\n3\n": True

u1 u2 u3 will be able to ping each other: True

The file /tmp/folder/1 on host finally contains the data "1\n2\n3\n": True

Question **26**

Complete

Mark 0.75 out of 1.00

Match each docker command with its meaning.

docker build -t new .

Build a new docker image using Dockerfile in current directory, and tag it as "new"

docker port
517065f6ab04

show all port mappings for the container with id 517065f6ab04

docker images

show list of docker images available on the local machine

docker run --rm ubuntu

run the ubuntu image, and remove it when the user runs "exit" inside the docker

Your answer is partially correct.

You have correctly selected 3.

The correct answer is: docker build -t new . → Build a new docker image using Dockerfile in current directory, and tag it as "new", docker port 517065f6ab04 → show all port mappings for the container with id 517065f6ab04, docker images → show list of docker images available on the local machine, docker run --rm ubuntu → run the ubuntu image, and remove it when its done, as a result "docker ps -a" will show nothing here

[◀ \(Assignment\) Kubernetes - 1](#)

Jump to...

[\(Assignment\) Kubernetes - 2 ▶](#)