**Basics of Container and Docker**

**Quiz - Basics of Container and Docker – Containers**

Q1 of 2

What is an executable package of all the files and dependencies required to run a container?

* *Image*
* Namespaces
* Cgroup
* Chroot

**Option Image is correct**

**Explanation :**

**An image is an executable package of all the files and dependencies required to run a container**

Q2 of 2

An organisation with limited OS resource availability should opt for

* Virtualisation
* *Containerisation*

**Option Containerisation is correct**

**Explanation :**

**Containerisation suns multiple applications that share a single OS on a VM or physical server thus reducing the requirement of multiple operating systems.**

**Quiz - Basics of Container and Docker - Introduction to Docker**

Q1 of 2

State whether the following statement is true or false.

Docker allows to cluster containers across different physical servers in the network.

* True
* False

**Option True is correct**

Q2 of 2

Creating an environment is as easy as running an executable file in?

* Virtualization
* Traditional deployment
* *Containerization*
* Microservices

**Option Containerization is correct**

Quiz - Basics of Container and Docker - Lifecycle of a container

Q1 of 7

docker run -it centos

A container launched by executing the above command is currently in which state?

* *Running*
* Exited
* Paused
* Stopped

**Option Running is correct**

**Explanation :**

**-it option launches the container with a terminal session. So, the container will currently be in running state.**

Q2 of 7

Which of the following commands have to be used to stop a background container immediately?

* *docker kill*
* docker stop

**Option docker kill is correct**

**Explanation :**

**docker kill kills a container immediately.**

Q3 of 7

How do you check the number of running containers on the host?

* *docker ps*
* docker ps -a
* docker ps -l
* docker ps -r

**Option docker ps is correct**

Q4 of 7

Detaching from a container

* Stops the container
* *Keeps the container running in the background*
* Removes the container
* Stops and removes the container

**Option Keeps the container running in the background is correct**

**Explanation :**

**Detaching from a container brings your back to the host console without stopping the container. It keeps it running in the background.**

Q5 of 7

State whether the following statements is true or false:

Multiple attachments can be made to a container from different sessions.

* True
* False

**Option True is correct**

**Explanation :**

**Attaching to a container is multiple sessions is possible.**

Q6 of 7

State whether the following statements is true or false:

Any stopped container can be attached to by executing the "docker attach" command.

* True
* False

**Option False is correct**

**Explanation :**

**A "stopped" container must be started before attaching to it.**

Q7 of 7

"docker logs" command can retrieve the logs of

* Only running foreground containers
* Only running background containers
* Both running foreground and background containers
* *All containers irrespective of the state*

**Option All containers irrespective of the state is correct**

**Explanation :**

**The logs of every container is stored by Docker.**

**Quiz - Basics of Container and Docker - Docker images**

Q1 of 8

A Docker image is

* *Immutable*
* Mutable

**Option Immutable is correct**

**Explanation :**

**Changes cannot be made to docker images as such. That is why they are immutable.**

Q2 of 8

The thin writable layer which stores all the changes made to a container during its runtime is called

* Image layer
* Base layer
* *Container layer*
* Parent layer

**Option Container layer is correct**

**Explanation :**

**Container layer is where the changes to a docker image are written.**

Q3 of 8

Which is the default registry contacted by the Docker engine when pulling an image?

* *DockerHub*
* Self-hosted registry
* Google registry
* AWS EC2 Container registry

**Option DockerHub is correct**

**Explanation :**

**Docker engine tries to access DockerHub registry by default.**

Q4 of 8

docker run -it centos

Assuming that there are no centos images present in the Docker host, what happens when the above command is executed?

* Error. Docker cannot launch a container without the image being present.
* *Docker pulls the image from the registry and launches the container.*

**Option Docker pulls the image from the registry and launches the container. is correct**

**Explanation :**

**Yes. Docker pulls the iamge from the registry first and then runs the image to launch a container.**

Q5 of 8

When an image is aliased using the "docker tag" command with no tag specified, what tag does the image take by default?

* Default
* Latest
* 1
* No tag

**Option Latest is correct**

**Explanation :**

**Which tag will be pulled from executing "docker pull centos"?**

**centos:latest**

Q6 of 8

Which command is used to remove the tags of an image?

* docker rm
* *docker rmi*

**Option docker rmi is correct**

**Explanation :**

**docker rmi removes the tag of images.**

Q7 of 8

Which command is used to build a new docker image interactively?

* *docker commit*
* docker build

**Option docker commit is correct**

**Explanation :**

**"docker commit" is used to build a docker image interactively.**

Q8 of 8

Which keyword is used in Dockerfile to specify the command to be executed once the container is run?

* RUN
* *CMD*

Option CMD is correct

Explanation :

CMD is used to specify the command that has to be run once the container is launched.

**Quiz - Basics of Container and Docker - DockerHub Registry**

Q1 of 4

How many types of repositories can be created in a DockerHub registry?

* One
* *Two*
* Three
* Four

**Option Two is correct**

**Explanation :**

**Based on the visibility of a repository, private and public repositories can be created.**

Q2 of 4

Before pushing a Docker image to a DockerHub repository, the image must be named as?

* *<hub\_id>/<repo\_name>:<tag>*
* <hub\_id>/<repo\_name>:<image\_name>
* <hub\_id>/<repo\_name>:<container\_name>
* <repo\_name>/<hub\_id>:<tag>

**Option <hub\_id>/<repo\_name>:<tag> is correct**

Q3 of 4

State whether the following statement is true or false:

Docker images can be pushed to DockerHub repositories without logging into the DockerHub account.

* True
* False

**Option False is correct**

**Explanation :**

**DockerHub needs the credentials of the registry before pushing an image.**

Q4 of 4

When a Docker image is pushed to the repository with no tag specified, what tag does it get assigned to by default?

* Default
* 1
* *Latest*
* No tag is assigned

**Option Latest is correct**

**Explanation :**

**Latest is the default tag that gets assigned.**

**Kubernates Lex Quizzes**

Q1 of 3

Which of the following are feature of Kubernetes?

* Scheduling
* Load balancing
* Container creation
* Scaling

Explanation :

Kubernetes schedules pods to nodes

Explanation :

Based on load Kubernetes distributes traffic.

Explanation :

Kubernetes does auto scaling of pods.

Q2 of 3

Jhon wants to create containers, which of the following service's will help him?

* Docker
* Docker Swarm
* Kubernetes
* Apache Mesos

Explanation :

Docker is an container engine.

Q3 of 3

Jack wants to automatically scale instances of an application based on demand from clients. What is the best solution?

* Setup Kubernetes cluster, create deployment and configure Horizontal scalingontal
* Take API requests and launch the required number of pods
* Setup Docker engine create the required instances using "docker run"
* All of the above

Explanation :

Deployments helps in managing instances and Horizontal scaling will scale up and down.

**Quiz - Kubernetes Architecture and Basic Components**

Q1 of 3

Which of the following are Kubernetes node components?

* Kubelet
* Scheduler
* Kube-proxy
* API server

Option Kubelet is correct

Explanation :

Its an node agent.

Option Kube-proxy is correct

Explanation :

Its a networking component

Q2 of 3

What is the job of Scheduler?

* To make sure entire data store is available on every node in a cluster.
* It’s an initial stage in Interaction with client.
* Make sure that Pods are mapped to the Nodes.
* Monitoring the status of application.

Option Make sure that Pods are mapped to the Nodes. is correct

Explanation :

Scheduler makes sure that pods are placed in right nodes.

Q3 of 3

Which kubernetes component does coordination of tasks such as configuration, load balancing, health monitoring, deployment, service discovery and job scheduling across cluster?

* Scheduler
* Controller
* etcd
* Kubelet

Option etcd is correct

Explanation :

Etcd is the data store that fulfills this requirement.

**Quiz - Kubernetes Objects – Pods**

Q1 of 3

You are required to host some web application on Kubernetes system. You need to create a container named "nginx-web" using nginx image.

Which of the following method(s) will help in creating the required nginx pod?

**Method 1:**

*[root@k8s-master|/]# kubectl run nginx-web --image=nginx*

**Method 2:**

*[root@k8s-master|/]# kubectl create deploy nginx-web --image=nginx --dry-run -o yaml > nginx-web-defn.yaml*

*[root@k8s-master|/]# kubectl apply -f nginx-web-defn.yaml*

* Only Method 1
* Either Method 1 or Method 2
* Only Method 2
* Neither Method 1 nor Method 2

**Option Only Method 1 is correct**

**Explanation :**

Pod definition file can not be generated using imperative object method. Hence, only method 1 can be followed.

Q2 of 3

As per the business requirement, pod named "nginx-web" using the below kubectl command for testing some web application deployment.

*[root@k8s-master|/]# kubectl run nginx-web --image=nginx*

*deployment.apps/nginx-web created*

As the testing activity is over, now you need to delete the pod. Which of the following command will delete the same?

(i) kubectl delete deploy nginx-web

(ii) kubectl delete pod nginx-web

* Only (ii)
* Either (i) or (ii)
* Only (i)
* Neither (i) nor (ii)

**Option Only (i) is correct**

Q3 of 3

You executed the below kubectl command to create the pods. The output of the command shows that it has created 2 pods named "centos-pod" and "simplesrv-pod".

*[root@k8s-master|/root/yamls]# kubectl apply -f pod*

*pod/simplesrv-pod created*

*pod/centos-pod created*

What do you infer from command's output?

* "pod" in the command is a name of the directory which has 2 pod definition files
* "pod" in the command is a yaml file which has 2 container definitions in it
* "pod" in the command is a yaml file which has 2 pod definitions in it

**Option "pod" in the command is a name of the directory which has 2 pod definition files is correct**

**Explanation :**

"pod" is the name of the directory, hence kubectl apply command processes all the yaml files in it.

**Quiz - Kubernetes Objects - Labels and Selectors**

Q1 of 3

You need to create a test container instance for running some web application on nginx. You write a below yaml definition file to create a Pod.

What will be the OUTPUT of the below command?

*[root@k8s-master|/]# kubectl create -f test\_nginx\_pod.yaml*

* It will be executed and creating a Pod successfully in running state
* It will throw an error
* It will be executed and creating a Pod successfully in containerCreating state
* It will be executed and creating a Pod successfully in pending state

**Option It will throw an error is correct**

**Explanation :**

**Labels fields can not be a part of Pod specification in yaml file.**

Q2 of 3

You have created multiple pods for various purpose such as backend, frontend, discovery, logging, etc. As there are hundreds of pods running in your cluster environment, you need to view the specific pods alone. You execute the below command to filter few pods.

[root@k8s-master|/]# kubectl get pods --selector 'tier in (frontend, backend)'

What will be the OUTPUT of the above command?

* It shows the pods that have the label value "frontend" and "backend" for the key "tier"
* It shows the pods that have the label value "frontend" for the key "tier"
* It shows the pods that have the label value "frontend" or "backend" for the key "tier"
* It shows the pods that have the label value "backend" for the key "tier"

**Option It shows the pods that have the label value "frontend" or "backend" for the key "tier" is correct**

**Explanation :**

**Operator "in" in selector will perform logical OR operation. Hence, it includes both the values.**

Q3 of 3

Below pod named "label-nginx-test" is created successfully with the labels and annotations defined.

*[root@k8s-master|/root/yaml]# cat test\_nginx\_pod.yaml*

*apiVersion: v1*

*kind: Pod*

*metadata:*

*name: label-nginx-test*

*labels:*

*environment: test*

*app: nginx*

*annotations:*

*buildVersion: 1.25*

*contact: cisrun@example.com*

*spec:*

*.....*

Now you wish to view the pods with the specific condition using "selector" field as shown below. What will be the OUTPUT of the below kubectl command?

*[root@k8s-master|/]# kubectl get pods --selector app=nginx,buildVersion=1.25*

* It will show the pods that have the labels app=nginx and buildVersion=1.25
* It will show "No resources found in default namespace."
* It will show the pods that have the labels app=nginx or buildVersion=1.25
* It will show the pods that have the labels app=nginx

**Option It will show "No resources found in default namespace." is correct**

**Explanation :**

Annotations can not be used to select the objects. You can use labels only.

Quiz - Kubernetes Objects - Services

Q1 of 3

Adam is working on an application which as a database pods running in back-end and web-server pods running as front-end. He as a Kubernetes Cluster administrator, wants the web-server pods to interact with database pods without exposing the back-end pods outside the cluster.

Which Kubernetes Service should he use?

* NodePort
* ClusterIP
* LoadBalancer
* External Name

**Option ClusterIP is correct**

**Explanation :**

Correct, ClusterIP is used to connect pods internally within the cluster.

Q2 of 3

Which of the following statements are TRUE regarding Service specification file?

* Service specification file without a type field under spec will create service of type Cluster IP by default
* The node port range should be between 30000 to 32767
* Service specification file without a type field under spec will create service of type NodePort by default
* Target Port refers to the port number set on node machine.

**Option Service specification file without a type field under spec will create service of type Cluster IP by default is correct**

**Explanation :**

Correct, by default ClusterIP service is created.

**Option The node port range should be between 30000 to 32767 is correct**

**Explanation :**

Correct, port number 30000 to 32767 is used by node machine

Q3 of 3

From the below options select the situation when the client needs to use service manifest file without selector filed configured inside it.

* When client wants to map/point a particular service created to another service located in different namespaces or cluster.
* When client wants to create a service mesh spanning over the whole cluster
* During migrating workload into Kubernetes, the user wants to work only on a small proportion/sets of pods.
* When client requires to use external database for production environment, but uses his own database while testing

**Option When client wants to map/point a particular service created to another service located in different namespaces or cluster. is correct**

**Explanation :**

Correct, service without selectors helps to map services between different namespaces.

**Option During migrating workload into Kubernetes, the user wants to work only on a small proportion/sets of pods. is correct**

**Explanation :**

Correct, its used during partial pod migration from one environment to other.

**Option When client requires to use external database for production environment, but uses his own database while testing is correct**

**Explanation :**

Correct, it's used when the client wants to use his own test cases for testing in pods.

**Quiz - Kubernetes Objects – ReplicaSet**

Q1 of 3

You wrote a configuration file for creating a ReplicaSet. As per the below yaml file, it should create 3 replicas of nginx container.

A screenshot of a computer program

Description automatically generated

Before creating a ReplicaSet, you have executed "kubectl get pods" command to view the pods that are already running.

*[root@k8s-master|/]# kubectl get pods --show-labels*

*NAME READY STATUS RESTARTS AGE LABELS*

*prod-nginx-1 1/1 Running 0 10m app=nginx,environment=prod*

*prod-nginx-2 1/1 Running 0 10m app=nginx,environment=prod*

How many pods will be created by the below kubectl command using the ReplicaSet definition file?

[root@k8s-master|/]# kubectl apply -f prod\_nginx\_rs.yaml

* 1
* 3
* 0
* 2

**Option 1 is correct**

Q2 of 3

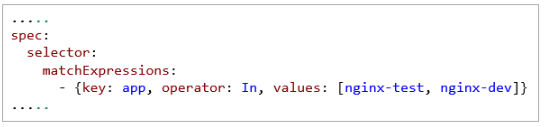
You have created a replication controller and replicaset instances in your Kubernetes cluster environment. Below are the selector labels used in the definition files.

Selector 1:

A white background with red and blue text

Description automatically generated

Selector 2:



Identify the type of the Kubernetes object that can include these selectors in its definition file.

* Selector 1: ReplicationController, Selector 2: ReplicaSet
* Selector 1: ReplicaSet, Selector 2: ReplicationController
* Selector 1: ReplicaSet, Selector 2: Pod
* Selector 1: Pod, Selector 2: ReplicationController

**Option Selector 1: ReplicationController, Selector 2: ReplicaSet is correct**

Explanation :

Equality-based selector can be used with ReplicationController and set-based selector can be used with ReplicaSet.

Q3 of 3

There are 3 test replica instances running using MySQL image with backend application. The definition file name is mysql-rs.yaml. Now due to the load on the database access, you need to scale the ReplicaSet with 2 more instances.

Which of the following method(s) can be used to scale the ReplicaSet?

(i) Edit the definition file and run kubectl replace -f mysql-rs.yaml

(ii) Edit the definition file and run kubectl apply -f mysql-rs.yaml

(iii) Run kubectl scale --replicas=5 -f mysql-rs.yaml

* Only (i) and (iii)
* Only (ii) and (iii)
* All the methods
* Only (iii)

**Option All the methods is correct**

**Quiz - Kubernetes Objects – Deployments**

Q1 of 3

As part of deploying the backend application on the MariaDB container, below instance is running.

*[root@k8s-master|/root/yaml]# kubectl get deploy*

*NAME READY UP-TO-DATE AVAILABLE AGE*

*mariadb-deploy 1/1 1 1 6s*

What will happen after the execution of the below kubectl command?

*[root@k8s-master|/]# kubectl delete pod mariadb-deploy*

* kubectl will create a new pod automatically after deleting the current one
* *Error "pods mariadb-deploy not found"*
* Error "Deployment can not be deleted using kubectl delete pod"
* It will delete the current one and won't create any new deployment

**Option Error "pods mariadb-deploy not found" is correct**

**Explanation :**

**Deployment can be deleted using "kubectl delete deploy" command.**

Q2 of 3

You need to view the ReplicaSet status that are already created in your Kubernetes cluster environment. You executed the below "kubectl" command to view the same. You can see that there are 2 different ReplicaSet exist with the same name as shown below.

*[root@k8s-master|/root/yamls]# kubectl get rs*

*NAME DESIRED CURRENT READY AGE*

*simpleservice-deploy-68bbcbc6bf 0 0 0 22m*

*simpleservice-deploy-c7c8898bf 4 4 4 73m*

What does the above OUTPUT imply?

* Old deployment "simpleservice-deploy-68bbcbc6bf" has been deleted and the new deployment "simpleservice-deploy-c7c8898bf" is created manually
* Old deployment "simpleservice-deploy-68bbcbc6bf" has been deleted manually and the new deployment "simpleservice-deploy-c7c8898bf" is created manually automatically
* Another set of new instances have been created manually
* *A new of version of the container image has been used to deploy the instances*

**Option A new of version of the container image has been used to deploy the instances is correct**

**Explanation :**

**Once the new version of definition file is used for creating a deployment, it will delete the existing deployment.**

Q3 of 3

As part of scaling the existing deployments that are used for hosting a web application, you execute the below command to scale the number of instances running in the deployment simpleservice-deploy. There are 3 instances already running

*[root@k8s-master|/root/yamls]# kubectl get deploy*

*NAME READY UP-TO-DATE AVAILABLE AGE*

*simpleservice-deploy 3/3 3 3 35m*

*[root@k8s-master|/root/yamls]# kubectl scale deployment.apps/simpleservice-deploy --replicas=5*

What does the above command imply?

* It will create 5 more instances
* It will delete existing instances after creating 5 new instances
* It will create 2 more instances
* It will delete existing instances and create 5 new instances

**Option It will create 2 more instances is correct**

**Explanation :**

**"replicas" field is the kubectl command points total number of instances in the deployment.**

**Quiz - Kubernetes Objects – DaemonSet**

Q1 of 2

Which of the following are the best suited scenarios to use DaemonSet in Kubernetes? [Choose 2 options]

* Performance of the deployed application
* Log collection
* Monitoring
* Log store

**Option Log collection is correct**

**Explanation :**

**Common uses of a DaemonSet are,**

1. running a cluster storage daemon on every node

2. running a logs collection daemon on every node

3. running a node monitoring daemon on every node.

**Option Monitoring is correct**

**Explanation :**

**Common uses of a DaemonSet are,**

1. running a cluster storage daemon on every node

2. running a logs collection daemon on every node

3. running a node monitoring daemon on every node.

Q2 of 2

In a Kubernetes cluster environment, there are 3 nodes being used to deploy the web application. Due to the high demand on the application over the period, now you need to add few more nodes in the cluster to ensure consistency service.

As part of the web application, there is a log collection instance running in the existing nodes through DaemonSet. You need to collect the logs from the new instances as well that you will add.

Which of the following statement is TRUE with respect to log collection instance on the new nodes?

* It will be created in the new nodes manually
* It will be created in the new and existing nodes manually
* It will be automatically created in the new and existing nodes
* It will be automatically created in the new nodes

**Option It will be automatically created in the new nodes is correct**

**Quiz - Kubernetes Objects – ConfigMap**

Q1 of 3

You are using ConfigMap to store the environment details that are related to web application.

Which of the following are best suited info that can be stored using ConfigMap? [Choose 2 options]

* Database password
* Application properties
* Database configurations except password
* Host credentials

Q2 of 3

You have created a ConfigMap to store application and database related properties in key-value pairs. Now, you need to refer few ConfigMap variables in the Pod definition and use it in a container as a command-line argument.

Which of the following properties should be defined in the Pod definition to use the variables as a command-line argument? [Choose 3 options]

* *Env*
* envFrom
* configMapRef
* *configMapKeyRef*
* *valueFrom*

Q3 of 3

You have stored the database properties in a ConfigMap named "mysql-config". Below is the pod definition file in which you inject the ConfigMap as a Volume.

*[root@k8s-master|/root/yaml]# cat prod\_mysql\_pod.yaml*

*apiVersion: v1*

*kind: Pod*

*.....*

*spec:*

*.....*

*volumeMounts:*

*- name: "config-vol"*

*mountPath: "/root/config"*

*volumes:*

*- name: "config-volume"*

*configMap:*

*name: mysql-config*

What will be the OUTPUT of the above pod definition file?

* Pod will be created, but it is not running
* *It returns the error "Not found: config-vol"*
* It returns the error "missing required field"
* Pod will be created and running successfully

**Quiz - Kubernetes Objects – Secrets**

Q1 of 3

You want to store the environment details that are related to web application.

Which of the following are best suited details that can be stored using Secrets? [Choose 2 options]

* Application properties
* Database password
* Host credentials
* Database configurations except password

**Option Database password is correct**

**Explanation :**

**Credentials are stored in encrypted format in Secrets.**

**Option Host credentials is correct**

**Explanation :**

**Credentials are stored in encrypted format in Secrets.**

Q2 of 3

You got a file from development team in which the encrypted database username and password are stored. As part of testing the application, you need to view the decrypted username and password.

Which of the following function you can use to view the same as a plain text?

* base64 –encode
* base64 --decode
* base32 –decode
* base32 --encode

**Option base64 --decode is correct**

**Explanation :**

**"base64 --decode" decrypts the encrypted string.**

Q3 of 3

You have created a Secret to store host and database credentials. Now, you need to inject those details in the Pod definition and use it in a container as an environment variables.

Which of the following properties should be defined in the Pod definition file to use the variables as an environment variables? [Choose 2 options]

* secret
* secretName
* envFrom
* secretRef

**Option envFrom is correct**

**Explanation :**

**It can be used to inject Secrets as an environment variables.**

**Option secretRef is correct**

**Explanation :**

**It can be used to inject Secrets as an environment variables.**