Congratulations! You passed!

Grade received 100%
Latest Submission Grade 100%
To pass 70% or higher
Retake the assignment in **7h 57m**
Go to next item

1. Do containers have to be on a specific operating system or platform? **1 / 1 point**

- (●) No, containers work on many types of operating systems and platforms.
- ( ) Yes, containers only work on Windows-based operating systems and platforms.
- ( ) Yes, containers only work on Linux-based operating systems and platforms.
- ( ) No, containers don't run on operating systems or platforms.

Correct
Correct! Containers can be used across multiple types of operating systems and platforms.

2. How are digital containers like shipping containers? **1 / 1 point**

- ( ) Digital containers must be welded to computer motherboards, just like shipping containers must be welded.
- ( ) Digital containers are not like shipping containers.
- (●) Container size and specs are standardized, which makes them easier to move around.
- ( ) Both digital and shipping containers have to travel internationally.

Correct
Correct! Digital containers make software portable so applications can run on multiple platforms.

3. What applications are a good fit for Docker? **1 / 1 point**

- ( ) Applications with rich GUI features
- (●) Applications that require flexible scaling and portability
- ( ) Applications that need a lot of security
- ( ) Applications with high performance requirements

Correct
Correct! Docker became popular with developers because of its simple architecture, high scalability, and easy portability.

4. What are three parts of Docker's underlying technology? **1 / 1 point**

- [x] Linux kernel features

Correct
Correct! Linux kernel features are a part of Docker's underlying technology.

- [x] The Go programming language

Correct
Correct! The Go programming language is a part of Docker's underlying technology.

- [x] Namespaces

Correct
Correct! Namespaces are a part of Docker's underlying technology.

- [ ] GitHub

5. What Docker features create a container image? **1 / 1 point**

- ( ) The run command and a Dockerfile
- (●) The build command and a Dockerfile
- ( ) The image command and a Dockerfile
- ( ) The copy command and an existing image

Correct

Correct! The build command is used with a Dockerfile to build a container image.

6. What are the steps used to create and run containers? **1 / 1 point**

- ○ Create a container image, use it to create a Dockerfile, and then use the Dockerfile to create a running container.
- ◉ Create a Dockerfile, use it to create a container image, and then use the container image to create a running container.
- ○ Create a Dockerfile and use it with the pull command to create a running container.
- ○ Input the container image name and tag.

Correct
Correct! The proper sequence of steps to create and run containers is to create a Dockerfile, use it to create a container image, and then use the container image to create a running container.

7. What is the function of the Docker 'run' command? **1 / 1 point**

- ○ Stores images in a configured registry
- ◉ Creates a container from an image
- ○ Lists all images, repositories, tags, and sizes
- ○ Retrieves images from a configured registry

Correct
Correct! The Docker 'run' command creates a container from an image.

8. What is a Docker container? **1 / 1 point**

- ○ A read-only template
- ○ A method of isolating communication
- ◉ A runnable instance of an image
- ○ A persistent set of data that can be transferred

Correct
Correct! A Docker container is defined as a runnable instance of an image.

9. What are volumes and bind mounts used for in Docker? **1 / 1 point**

- ○ Erasing data
- ○ Connecting to external storage platforms
- ○ Isolating communication
- ◉ Persisting data

Correct
Correct! Volumes and bind mounts are used to persist data in Docker.

10. What does the Docker client-server architecture provide? **1 / 1 point**

- ○ A communication channel
- ◉ An application environment
- ○ Code checking
- ○ Cloud storage

Correct
Correct! The Docker client-server architecture provides a complete application environment.