Phase-5 Practice Project: Assisted Practice

1. **Build a Custom Docker Image**.

✓ Clone Git repository

● First, clone the Git repository on a Docker host using the command below:

> git clone https://github.com/siri10080/Docker.git

✓ Docker Build

● Then, proceed with the docker build command to build a custom docker image.

```
root@ip-172-31-25-208:~/Docker# docker run -d -p 80:80 --name phpcode phpcode
406f301bc231926332d263572e34cc55b70ca94157177b9dfb7bc5a598d07241
root@ip-172-31-25-208:~/Docker# docker ps
CONTAINER ID      IMAGE           COMMAND            CREATED        STATUS        PORTS
406f301bc231      phpcode         "/usr/sbin/apache2 -â¦"  2 seconds ago   Up 1 second   0.0.0.0:80->80/tcp
root@ip-172-31-25-208:~/Docker# curl localhost
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
        <title>Simple PHP App</title>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link href="assets/css/bootstrap.min.css" rel="stylesheet">
        <style>body {margin-top: 40px; background-color: #333;}</style>
        <link href="assets/css/bootstrap-responsive.min.css" rel="stylesheet">
        <!--[if lt IE 9]><script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script><![endif]-->
    </head>

    <body>
        <div class="container">
            <div class="hero-unit">
                <h1>Simple PHP App</h1>
                <h2>Congratulations</h2>
                <p>Your PHP application is now running on a container in Amazon ECS.</p>
                <p>The Kubernetes Docker container is running PHP version 7.0.32-0ubuntu0.16.04.1.</p>
                root@ip-172-31-25-208:~/Docker#
```

cd Docker

dockerbuild -t phpcode . -f Dockerfile

```
root@docker:~/Docker# docker build -t phpcode .
Sending build context to Docker daemon  337.9kB
Step 1/14 : FROM ubuntu
 ---> 93fd78260bd1
Step 2/14 : ENV DEBIAN_FRONTEND=non-interactive
 ---> Using cache
 ---> b21eb69f632a
Step 3/14 : RUN apt-get update -y
 ---> Using cache
 ---> d2e4866734b9
Step 4/14 : RUN apt-get install -y git curl apache2 php libapache2-mod-php php-mysql
 ---> Using cache
 ---> 85f084edfc0b
Step 5/14 : RUN rm -rf /var/www/html/*
 ---> Using cache
 ---> b56166da0f16
Step 6/14 : ADD src /var/www/html/
 ---> Using cache
 ---> ba9e5c5c651c
Step 7/14 : RUN a2enmod rewrite
 ---> Using cache
 ---> cff3e4bb8c42
Step 8/14 : RUN chown -R www-data:www-data /var/www/html
 ---> Using cache
 ---> 7a4314c7b69b
Step 9/14 : ENV APACHE_RUN_DIR /var/www/html
 ---> Using cache
 ---> 663a68663f90
```

- Once the image is built, check the image using docker run command and then run it to initialize custom container on Docker host. docker images

```
root@ip-172-31-25-208:~/Docker# docker run -d -p 80:80 --name phpcode phpcode
406f301bc231926332d263572e34cc55b70ca94157177b9dfb7bc5a598d07241
root@ip-172-31-25-208:~/Docker# docker ps
CONTAINER ID    IMAGE         COMMAND              CREATED        STATUS        PORTS
406f301bc231    phpcode       "/usr/sbin/apache2 -å¦"  2 seconds ago  Up 1 second   0.0.0.0:80->80/tcp
root@ip-172-31-25-208:~/Docker# curl localhost
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
        <title>Simple PHP App</title>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link href="assets/css/bootstrap.min.css" rel="stylesheet">
        <style>body {margin-top: 40px; background-color: #333;}</style>
        <link href="assets/css/bootstrap-responsive.min.css" rel="stylesheet">
        <!--[if lt IE 9]><script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script><![endif]-->
    </head>

    <body>
        <div class="container">
            <div class="hero-unit">
                <h1>Simple PHP App</h1>
                <h2>Congratulations</h2>
                <p>Your PHP application is now running on a container in Amazon ECS.</p>
                <p>The Kubernetes Docker container is running PHP version 7.0.32-0ubuntu0.16.04.1.</p>
                root@ip-172-31-25-208:~/Docker#
```

docker run -d --name phpcode -p 80:80 phpcode

```
Step 10/14 : ENV APACHE_RUN_USER www-data
 ---> Using cache
 ---> 2915860f4df7
Step 11/14 : ENV APACHE_RUN_GROUP www-data
 ---> Using cache
 ---> c83847b098df
Step 12/14 : ENV APACHE_LOG_DIR /var/log/apache2
 ---> Using cache
 ---> c4428c1a14db
Step 13/14 : EXPOSE 80
 ---> Using cache
 ---> 7374debb3213
Step 14/14 : CMD ["/usr/sbin/apache2", "-D",  "FOREGROUND"]
 ---> Using cache
 ---> d35edda13537
Successfully built d35edda13537
Successfully tagged phpcode:latest
root@ip-172-31-25-208:~/Docker#
root@ip-172-31-25-208:~/Docker# docker images
REPOSITORY            TAG                IMAGE ID            CREATED            SIZE
phpcode               latest             d35edda13537        20 minutes ago     291MB
ubuntu                16.04              a51debf7e1eb        7 days ago         116MB
root@ip-172-31-25-208:~/Docker#
```

- Once the container is up and running, validate the connectivity using the **curl** command to see if **php code** is running on port 80 or not.