# Hands-on labs Introduction to Docker

## Exercise 1: Run a docker container on Windows

In this exercise you will run a container to inspect the concepts of immutability.

Docker.

Using steps on the command-line

Verify that docker is correctly installed by running

docker version

Start by running a simple windows container with the command prompt in it.

docker run -it microsoft/windowsservercore cmd

Then inside the newly started container run the following commands

dir

md demo

cd demo

copy con myfile.txt

Hello this is in my docker container

ctrl+c

dir

Now you should see the text file you created in the steps above

Exit the container by running

ctrl+p ctrl+q

When back on the host run the following command

docker ps

Now you should see a container running with a generated name

Verify that you don’t see the file on your host by going to c: and running dir.

To stop the container run “docker stop name-of-the-container”. Then use

“docker ps” to view all running container and use “docker ps -a” to view all containers even stopped ones.

Now run “docker run -it microsoft/windowsservercore cmd” again. Inside the container run “dir”.

Verify that the folder and file your created earlier in not there anymore.

## Exercise 2: Build a new Container Image

In this exercise, you will go through the steps of creating a new Container image first from the command line using Docker commit and in a second step by using a docker file and the docker build command. Accompanying files can be found I the demo 1.2.4 folder. The complete video of this exercise can be found in the course.

Using steps on the command-line

docker run -it --name mycontainer microsoft/windowsservercore cmd

md builddemo

cd buidldemo

copy con myfile.txt

this is a text file

ctrl+c

exit

docker ps -a

docker commit mycontainerimage mycontainer

docker images

docker stop mycontainer

docker rm mycontainer

docker run -it --name mycontainer microsoft/windowsservercore cmd

dir

exit

docker run -it mycontainerimage cmd

dir

you see the demo folder again

#### Now using the docker build file

Ensure you are in the folder 1.2.4 that contains the docker file

Docker build -t mynewiiscontainer .

Docker images

Docker run -d -p 80:80 mynewiiscontainer

Docker ps -> find the running container ID

Docker inspect containerID

Find the IP address

Start a browser to the found IP address and see the new website running on port 80 serving the index.htm file that is in the current folder

## Exercise 3: Using docker-compose to orchestrate multiple containers

In this exercise, you will create multiple container instances that have a dependency. You will create a blog website that has a dependency on a database that also run’s in a container. The required files for this demo can be found in the folder demo 1.4.4

The complete video of this exercise can be found in the course.

In the folder demo 1.4.4 run the following command line:

Docker-compose up

Start a new command prompt to inspect the running containers

Inspect the running containers and find the ip address of the web frontend

Docker ps

Docker inspect contianerID

Use a browser to go to the blogsite running in the containers (<http://containerIP/blogsite>)

Ensure you are in the demo 1.4.4. folder so we can clean up the containers

Docker-compose down

Docker ps

Docker ps -a

## Exercise 4: Create a continuous delivery pipeline with VSTS

In this exercise you create a build and deployment pipeline to run your custom created docker image on a build machine for validation

The complete video of this exercise can be found in the course.

Prerequisites:

A Visual studio team services account

A virtual machine running windows server 2016 + a VSTS build agent

The folder Demo 1.6.4 that contains the docker file + iis homepage we replace

#### Creating a build

Create or use a Visual studio team Services account

Add the sources found in folder demo 1.6.4 to the git repository

Create a new build, that contains the following steps:

Get sources

Run docker image build task

Give the image a name of your linking + a tag that is the build number

Check the box on the task to also mark as latest

Run docker publish task

Publish the image to your container registry on docker-hub

Upload artifacts to VSTS

Add readme.md to the artifacts folder, this is required for the deployment to succeed. (it expects always at least one artifact published from a build)

#### Create a deployment

Add a new release definition

Rename environment to verify

In the verify environment add the following tasks:

Docker run

Specify the image you just created and uploaded to the docker hub

Specify a name we can use as reference later in the other tasks

Docker stop

Specify a docker command stop with the name of the image

Docker rm

Specify a docker rm command with the name of the image

Create a new release

Check the logs on completion

Check the build agent if it is still clean after a full run of the pipeline

If time permits:

Add a task to use curl to retrieve the home page that is part of the custom iis image