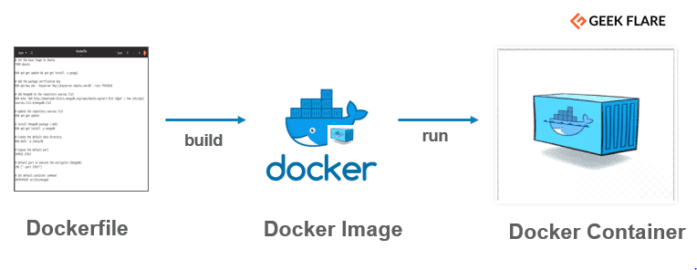
**DOCKERFILE**



**Dockerfile:**

**Dockerfile is a simple text file that consists of instructions to build Docker images.**

**Mentioned below is the syntax of a Dockerfile:**

**Syntax**

**# comments**

**command argument argument1...**

**Example**

**# Print "Get Certified. Get Ahead"**

**Run echo "Get Certified. Get Ahead"**

**Now, let's have a look at how to build a Docker image using a dockerfile.**

**Dockerfile consists of specific commands that guide you on how to build a specific Docker image.**

**The specific commands you can use in a dockerfile are:**

**FROM, PULL, RUN, and CMD**

**FROM - Creates a layer from the ubuntu:18.04**

**PULL - Adds files from your Docker repository**

**RUN - Builds your container**

**CMD - Specifies what command to run within the container**

**Mentioned below is an example of the dockerfile with the important commands**

**FROM ubuntu:18.04**

**PULL. /file**

**RUN make /file**

**CMD python /file/file.py**

**ENTRYPOINT allows specifying a command along with the parameters**

**Syntax**

**ENTRYPOINT application "arg, arg1".**

**Example**

**ENTRYPOINT echo "Hello, $name".**

**ADD command helps in copying data into a Docker image**

**Syntax**

**ADD /[source]/[destination]**

**Example**

**ADD /root\_folder/test\_folder**

**ENV provides default values for variables that can be accessed within the container**

**Syntax**

**ENV key value**

**Example**

**ENV value\_1**

**MAINTAINER declares the author field of the images**

**Syntax**

**MAINTAINER [name]**

**Example**

**MAINTAINER author\_name**

**Example**

This is a sample Image

FROM ubuntu

MAINTAINER demousr@gmail.com

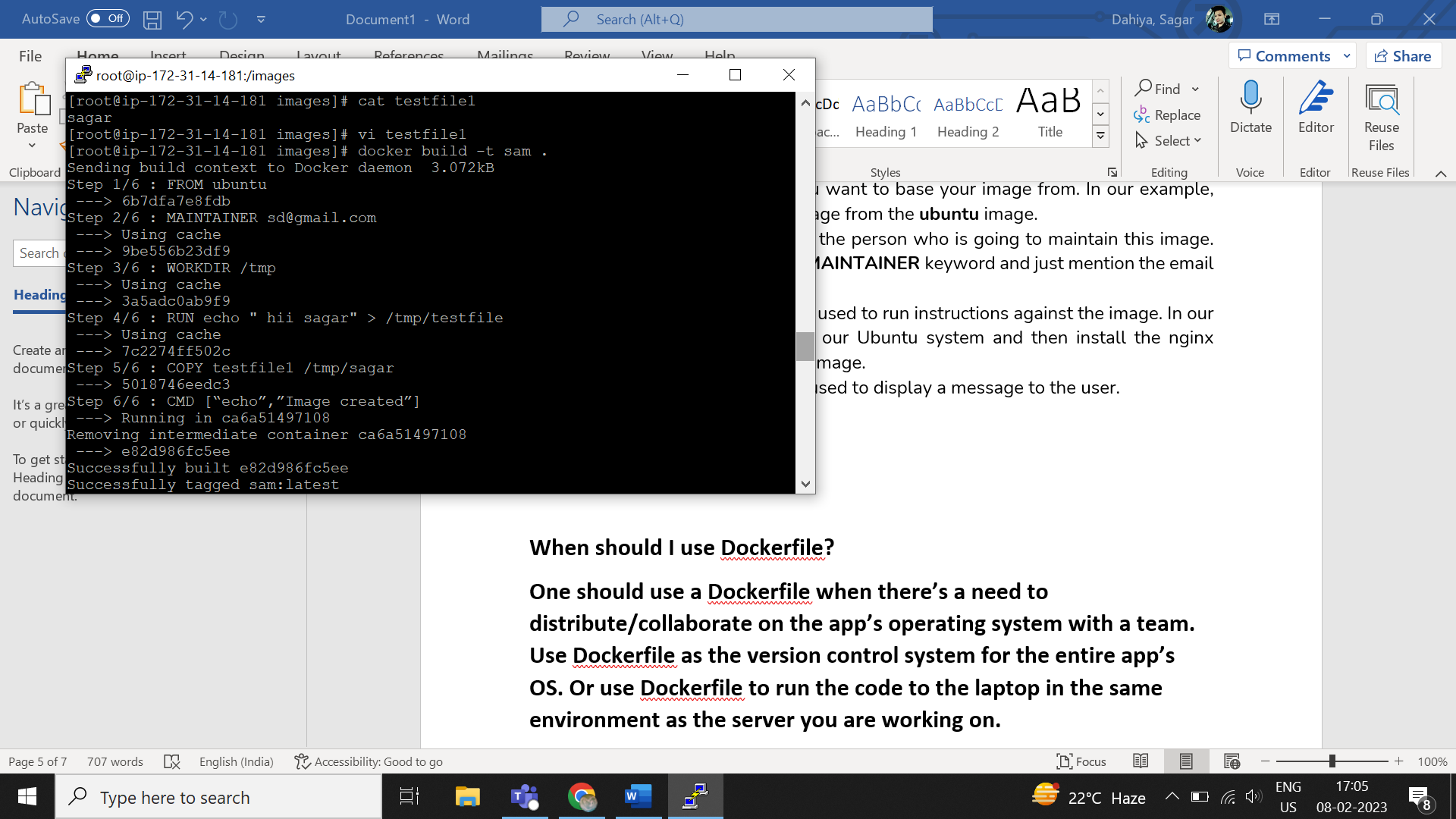
RUN apt-get update

RUN apt-get install –y nginx

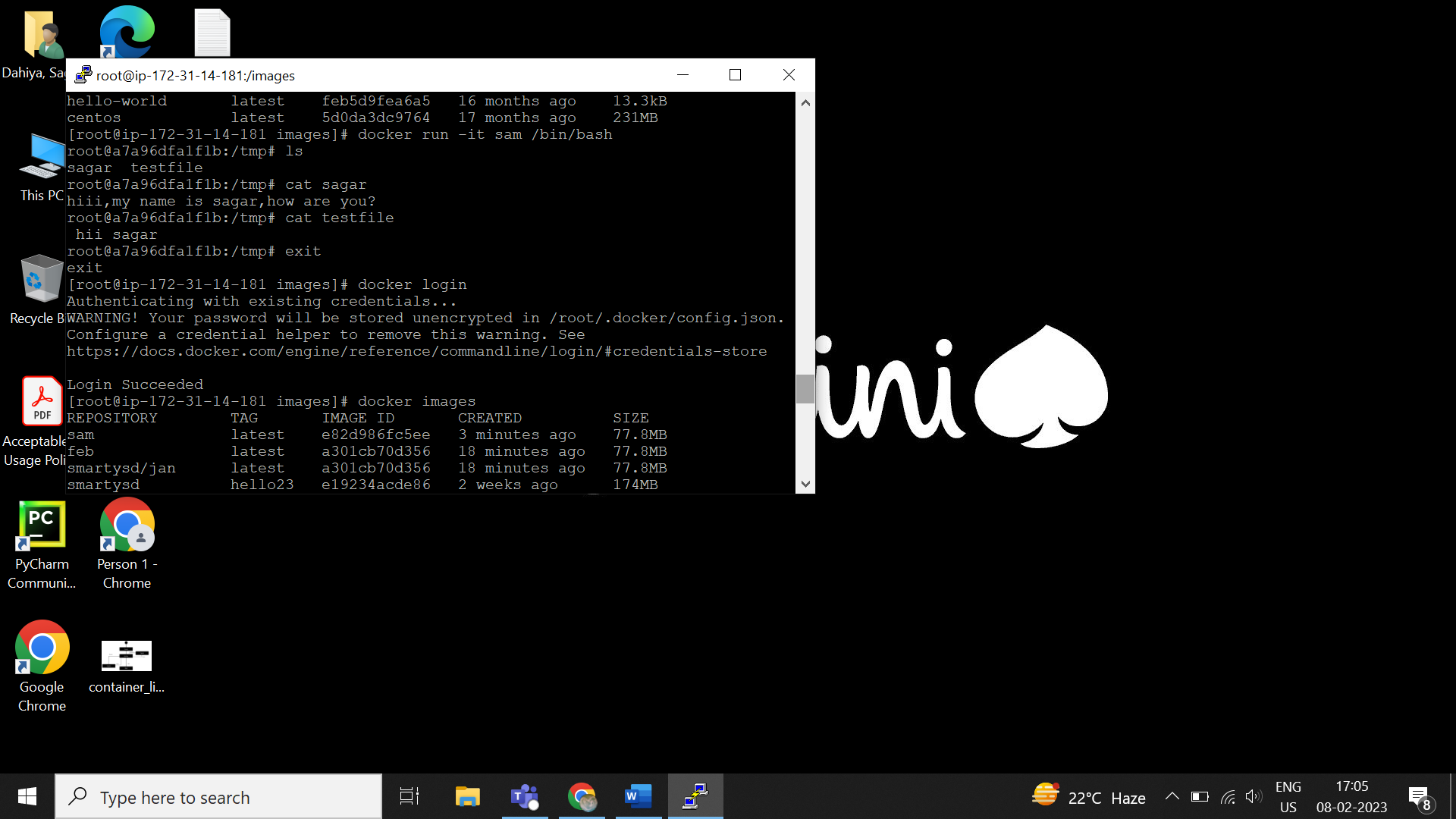
CMD [“echo”,”Image created”]

The following points need to be noted about the above file −

* The first line "#This is a sample Image" is a comment. You can add comments to the Docker File with the help of the **#** command
* The next line has to start with the **FROM** keyword. It tells docker, from which base image you want to base your image from. In our example, we are creating an image from the **ubuntu** image.
* The next command is the person who is going to maintain this image. Here you specify the **MAINTAINER** keyword and just mention the email ID.
* The **RUN** command is used to run instructions against the image. In our case, we first update our Ubuntu system and then install the nginx server on our **ubuntu** image.
* The last command is used to display a message to the user.

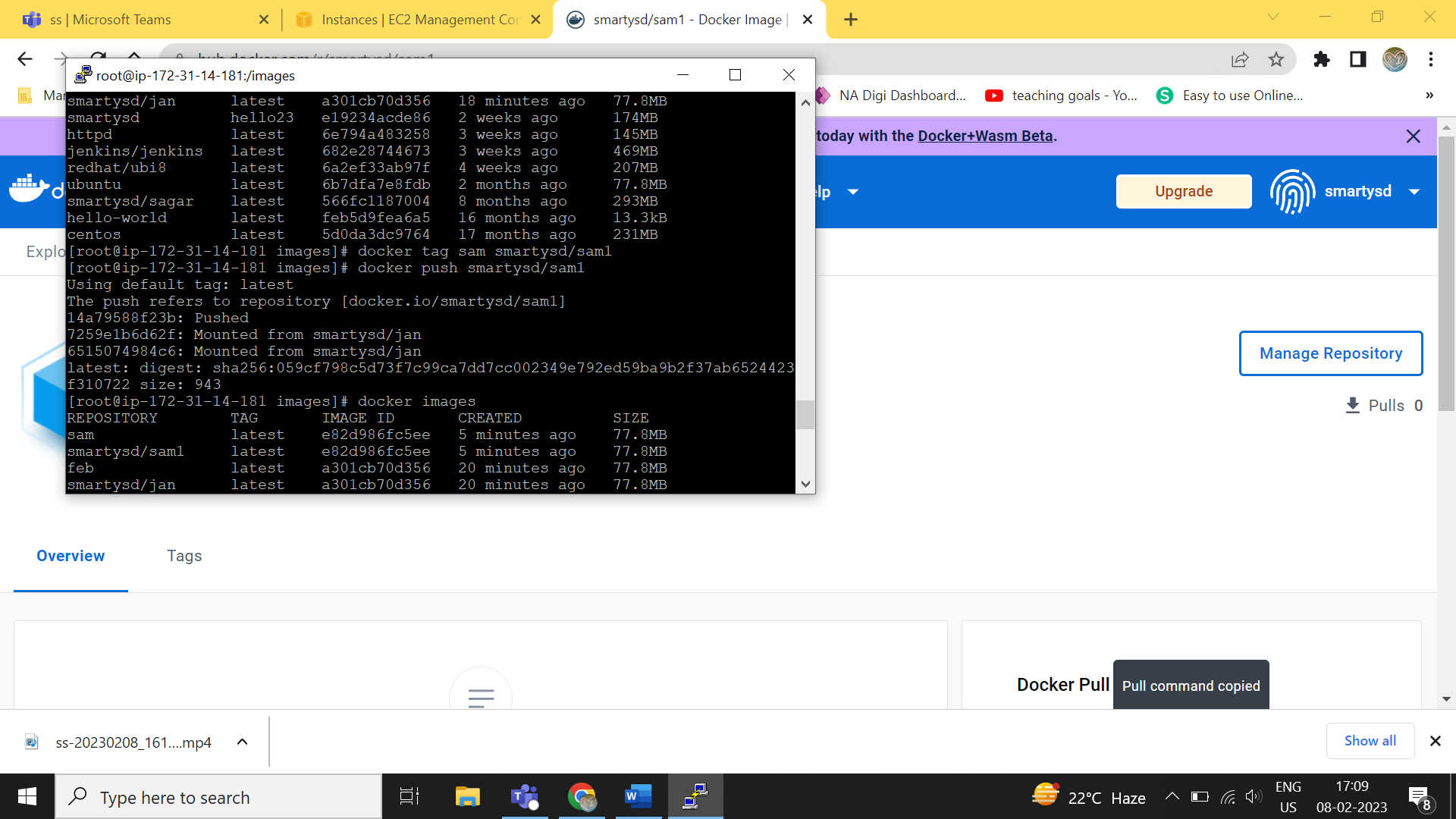


**Docker run -it imagename /bin/bash**

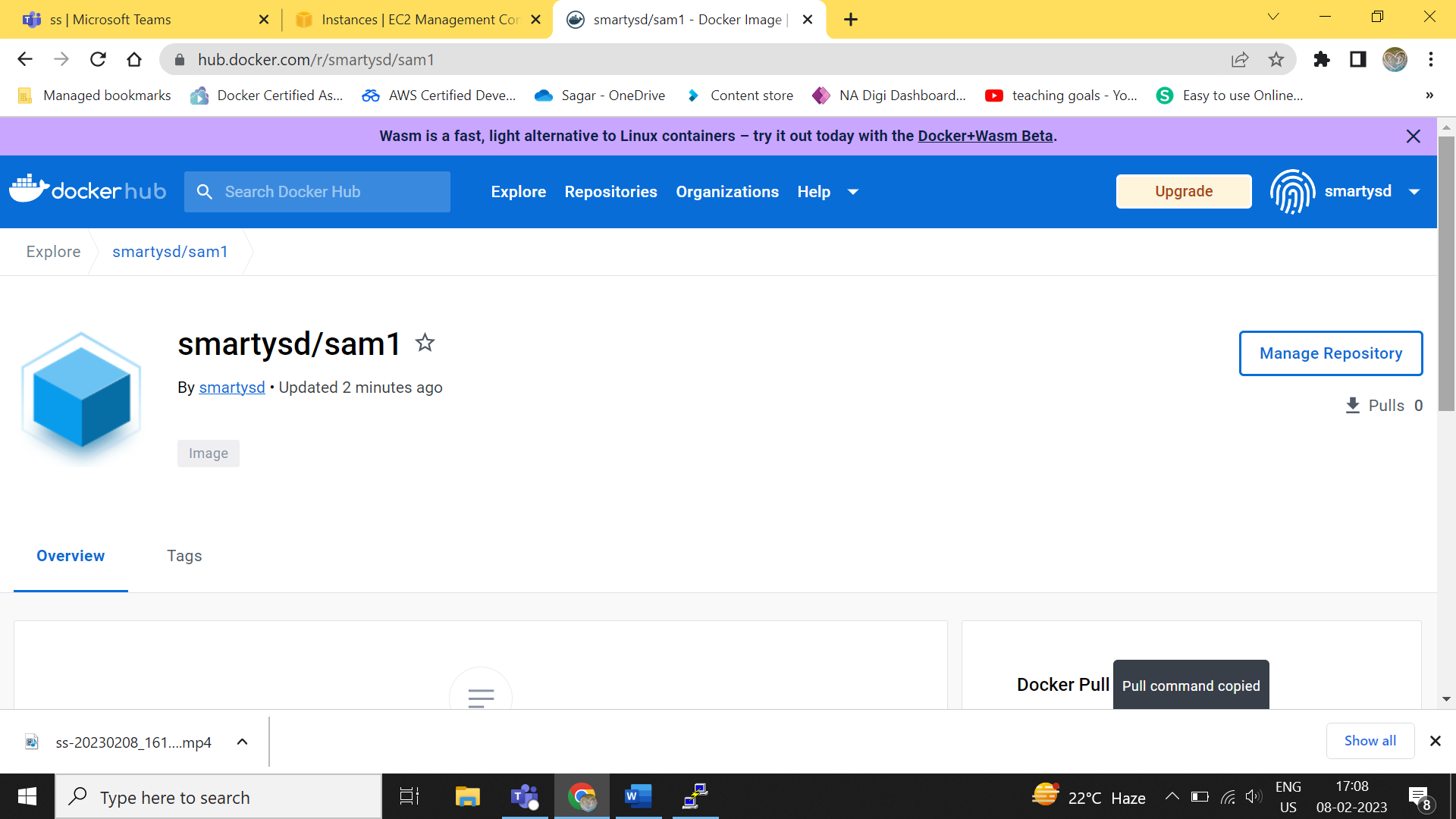


**You can see all files we copied using docker file.**

**Dockerhub : to make image available to anyone,we push it to dockerhub**

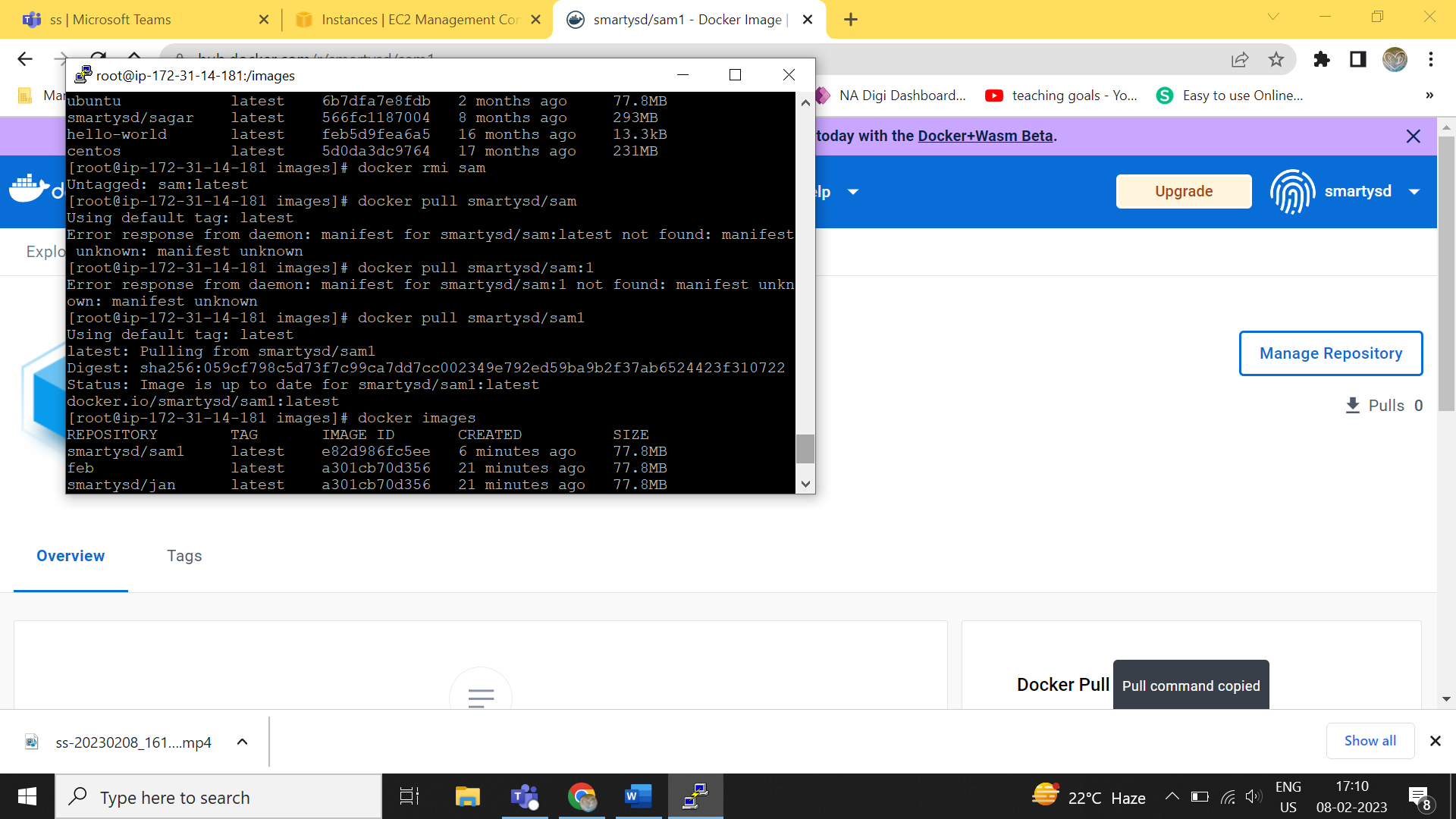


**Now check into dockerhub**



**To pull image,use**

**Docker pull smartysd/sam**



**When should I use Dockerfile?**

**One should use a Dockerfile when there’s a need to distribute/collaborate on the app’s operating system with a team. Use Dockerfile as the version control system for the entire app’s OS. Or use Dockerfile to run the code to the laptop in the same environment as the server you are working on.**

**Is Dockerfile a text file?**

**Dockerfile is a text document containing all the commands the user requires to call on the command line to assemble an image. With the help of a Dockerfile, users can create an automated build that executes several command-line instructions in succession.**

**How do I create a simple Dockerfile?**

**To create a Dockerfile, set up Docker and Docker Hub. Create the original Docker container and then create a file on it. Make changes to the container, and finally, create a new image.**

**What is Dockerfile language?**

**Go language is used to write Docker. A Dockerfile is a text file that contains collections of instructions and commands that will be automatically executed in sequence in the docker environment for building a new docker image.**

**What is Docker compose vs. Dockerfile?**

**The key difference between Docker compose, and Docker is that the Docker contents describe how to create and build a Docker image, while Docker compose is a command that runs Docker containers based on settings described in a docker-composed.yaml file.**

**What is a .dockerignore file?**

**A .dockerignore file allows you to specify a list of files or directories that Docker is to ignore during the build process. It is similar to a .gitignore file, which is used when you build Git repositories. You can specify the list of files and directories inside the .dockerignore file.**

**How do I commit a docker container?**

**First, you need to pull a docker image. Then deploy the container, modify it and commit the changes to the image. When you commit to changes, you essentially create a new image with an additional layer that modifies the base image layer.**