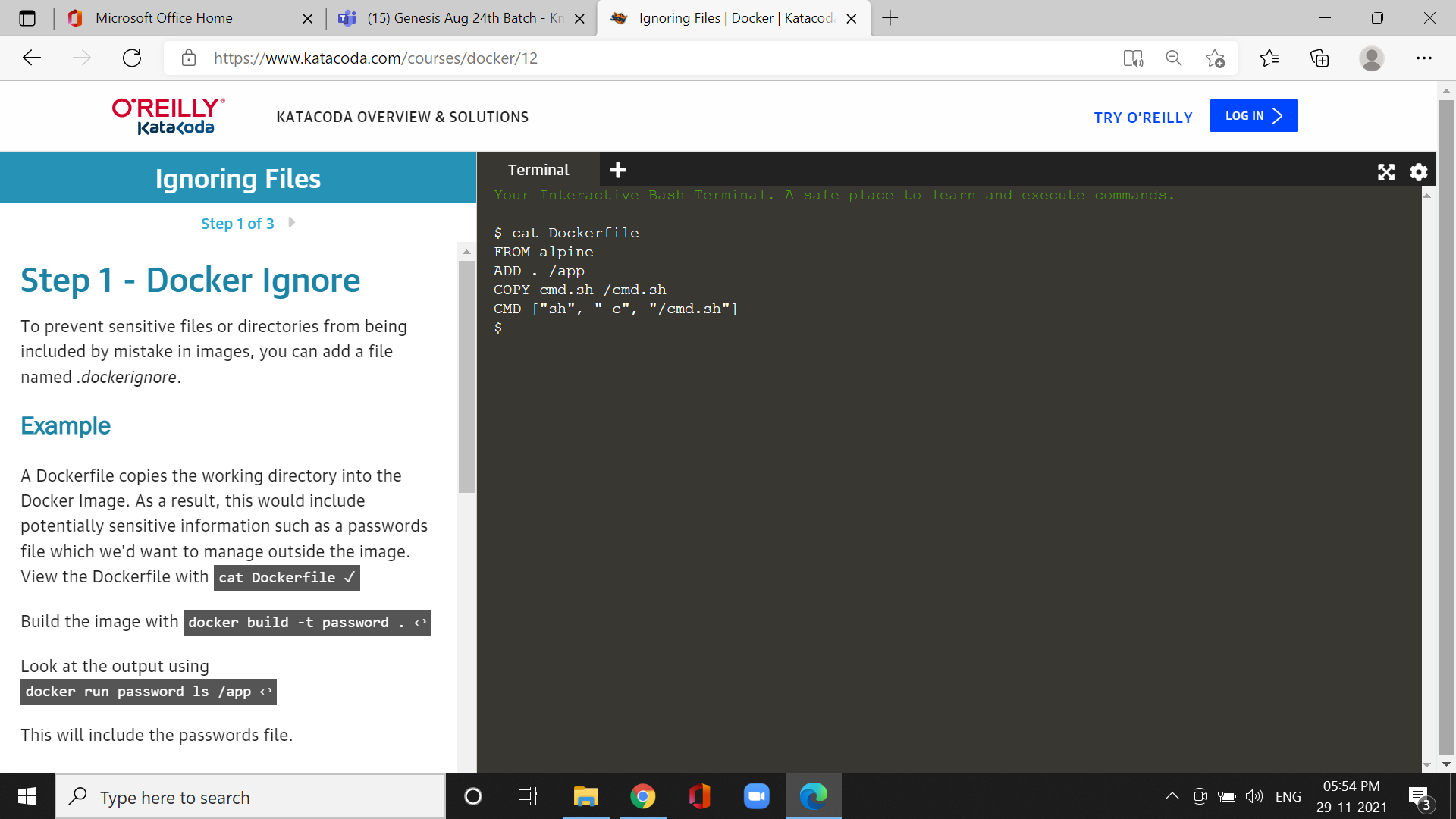
Scenario 6:

**Ignoring Files:**

**Step1: docker ignore**

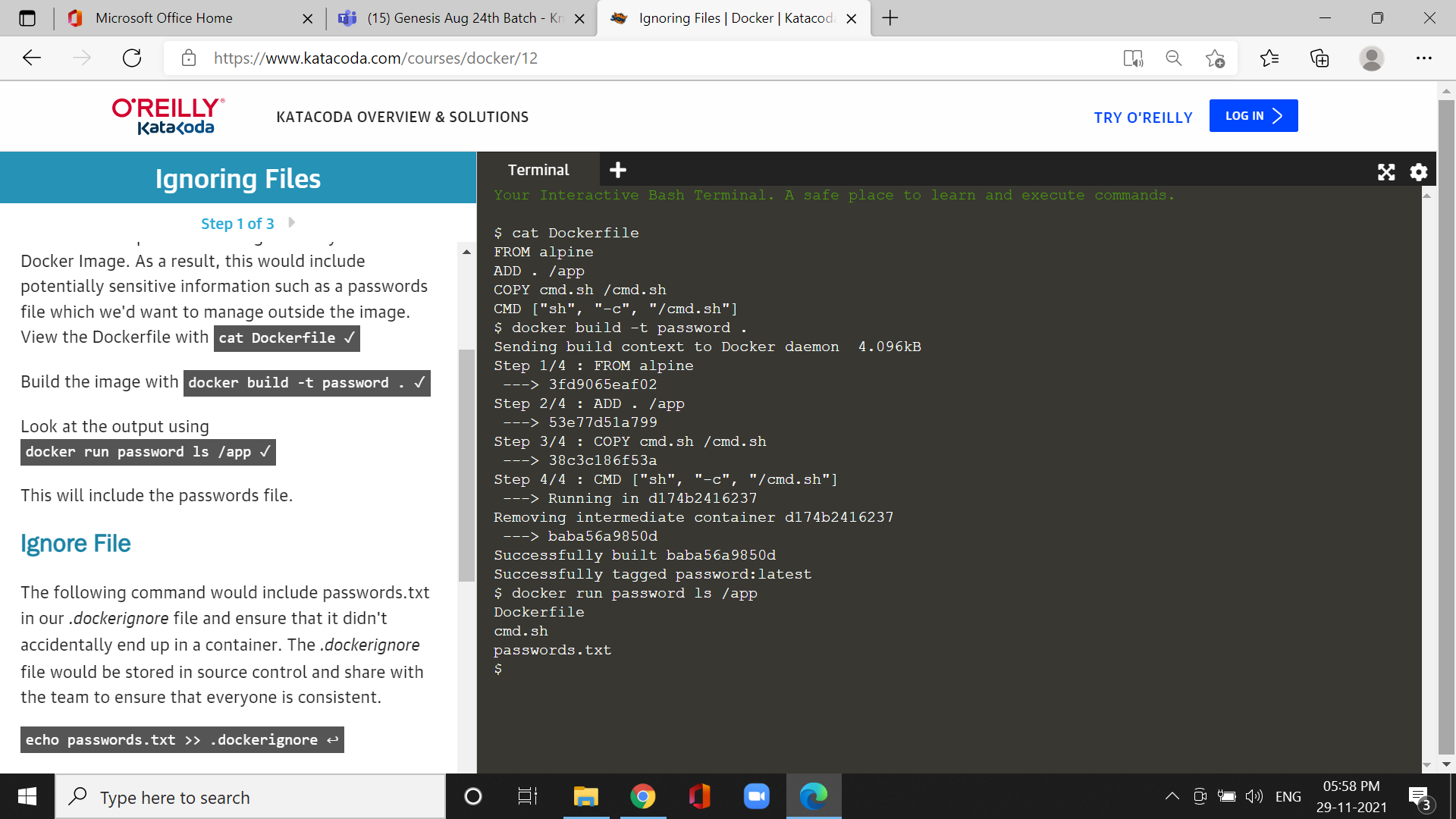
\*A Dockerfile copies the working directory into the Docker Ima**ge.**

Cmd:docker file

****

**\***To manage outside image dockerfile.

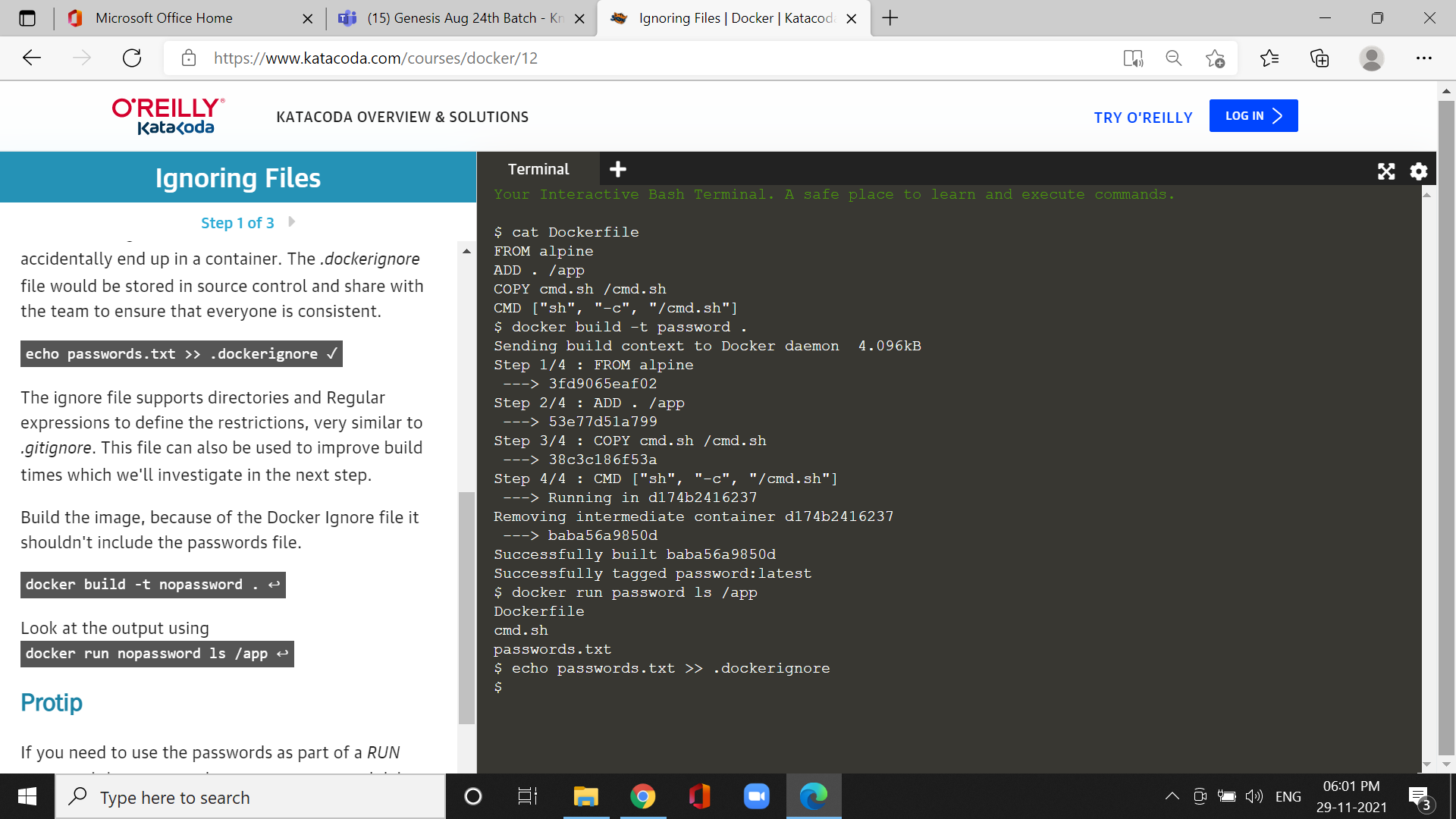
Cmd:docker run password ls/app



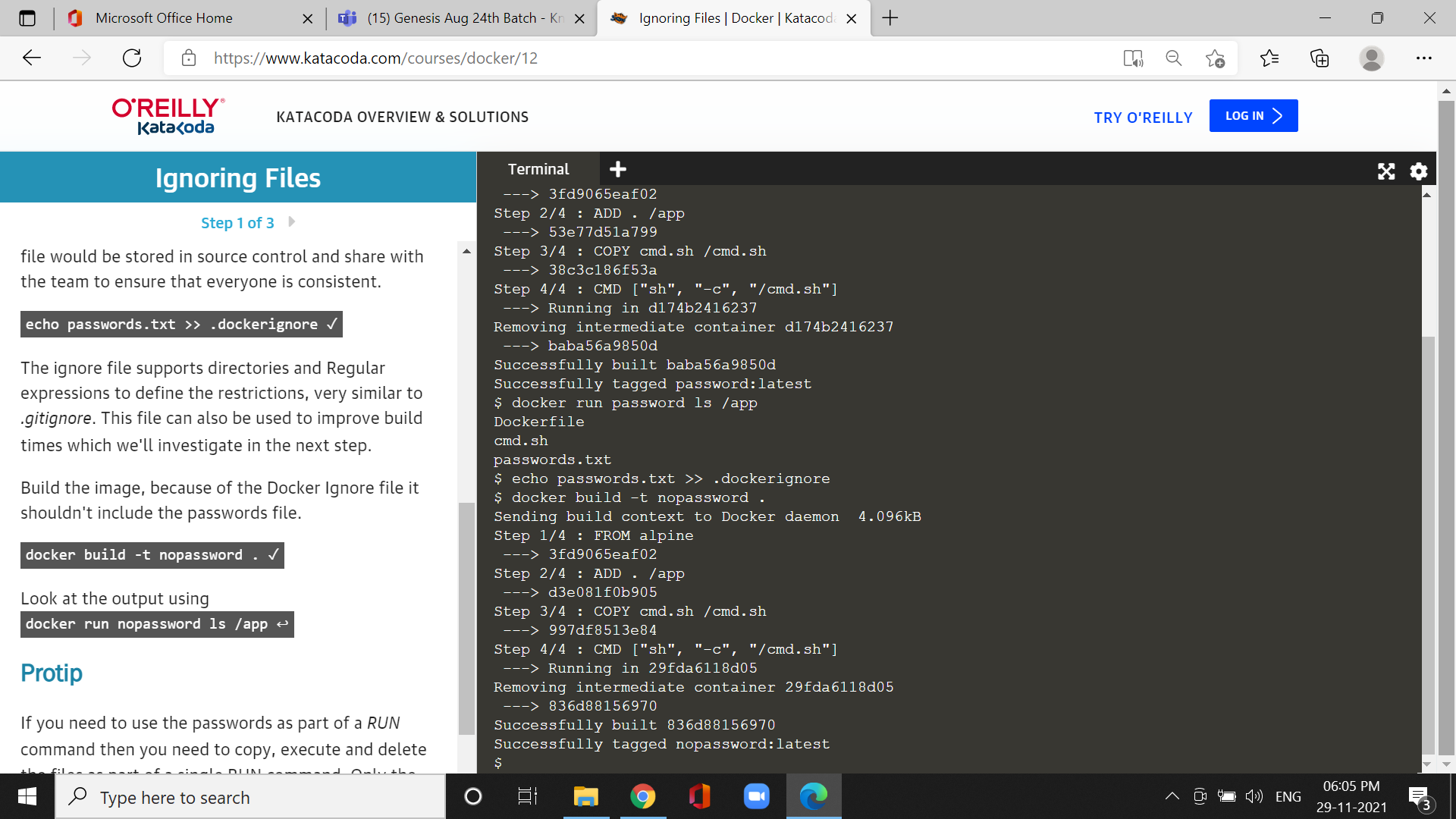
\*it will include the passwords file.

\*The *.dockerignore* file would be stored in source control and share with the team to ensure that everyone is consistent.

Cmd:echo passwords.txt >> .docker ignore



Cmd:docker build -t nopassword.

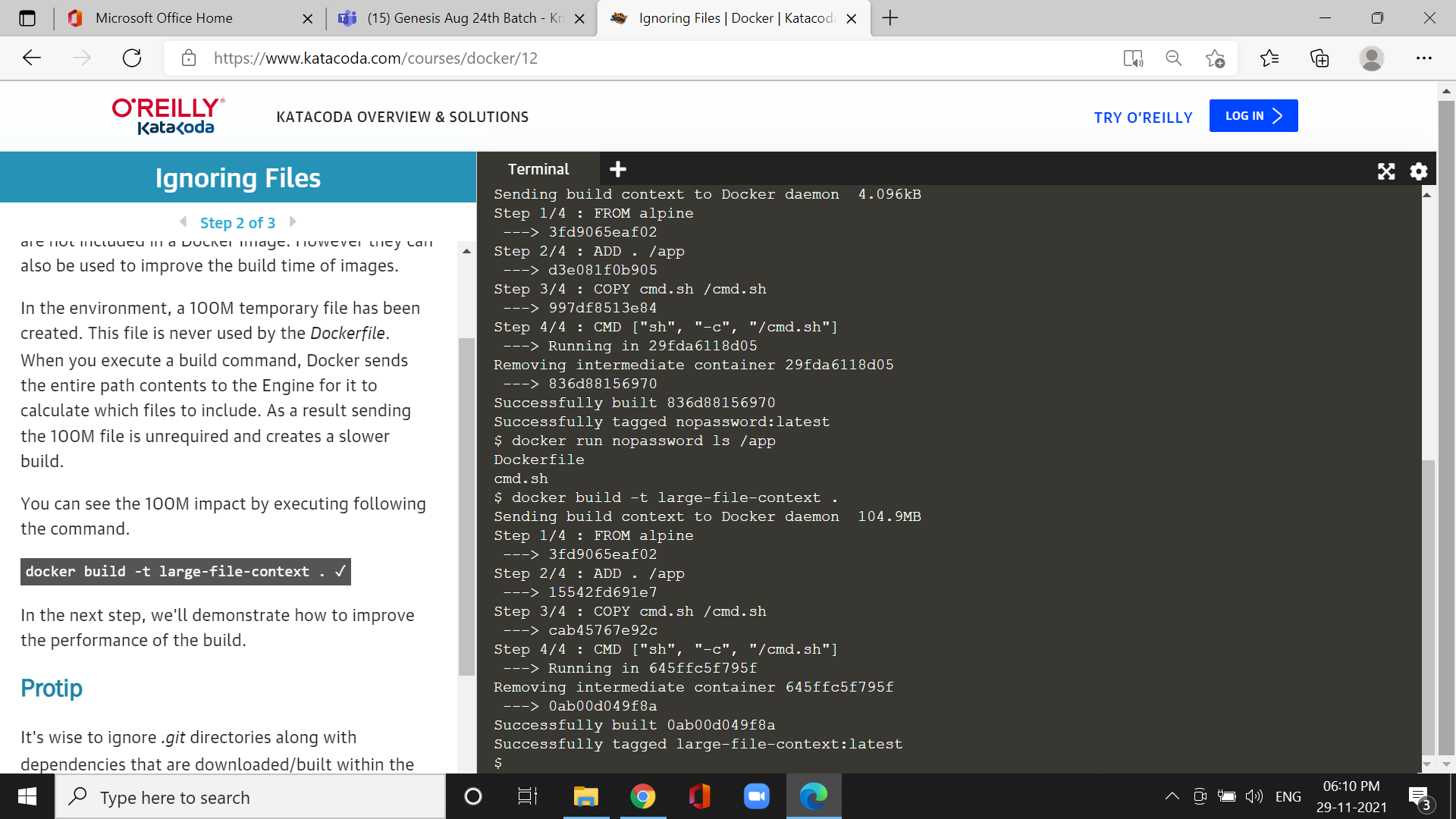


\*Docker Ignore file it shouldn't include the passwords file.

Step2:Docker build context

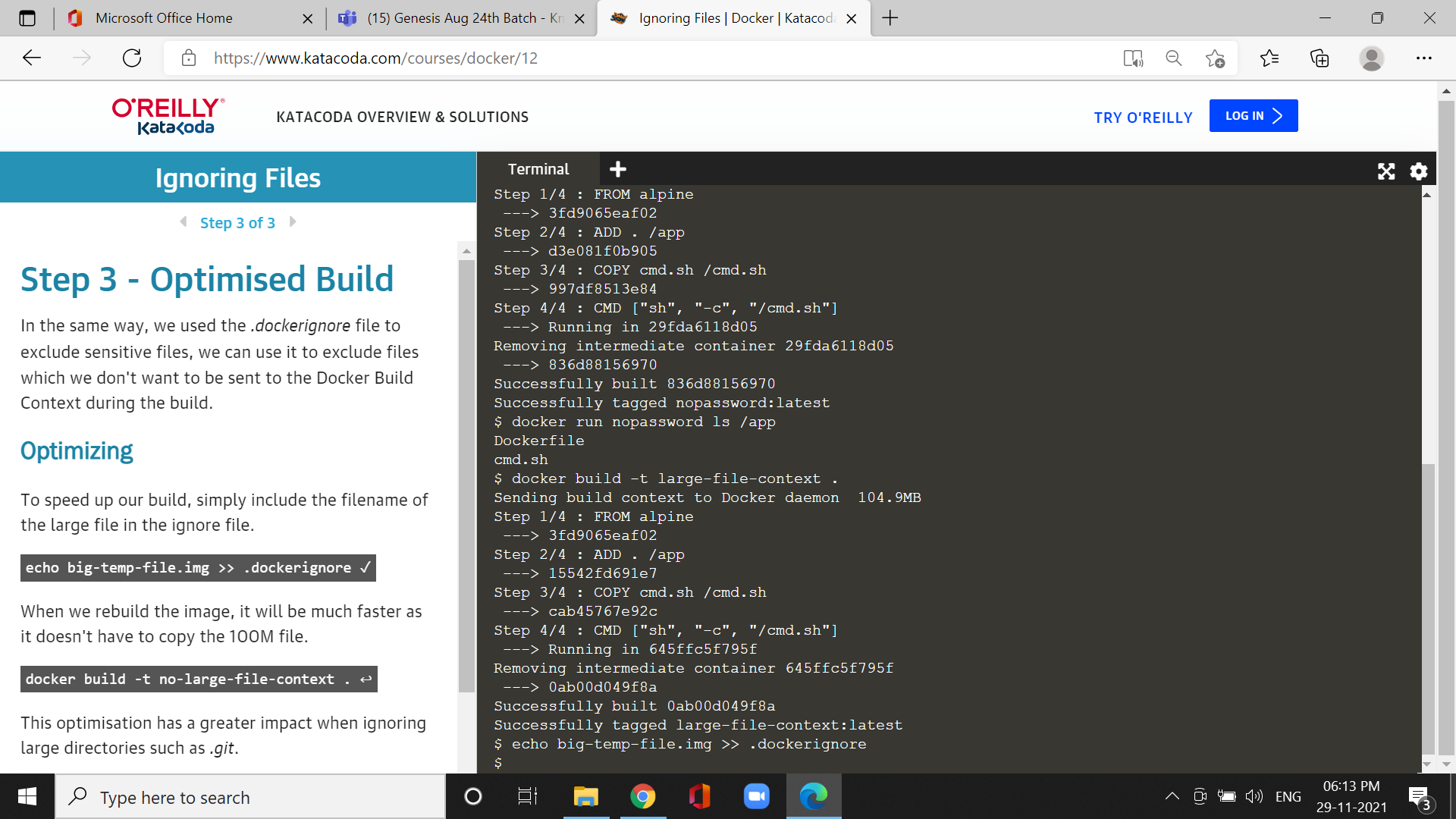
\*We can execute a build command, Docker sends the entire path contents to the Engine for it to calculate which files.

Cmd:docker build -t large-file-context.



\*The 100M file is unrequired and creates a slower build.

**Step3 :Optimized build**

Cmd :echo big-temp-file.img >>.dockerimage

\*it will rebuild the image, it will be much faster as it doesn't have to copy the 100M file.

Cmd:docker build -t no-large-file-context.