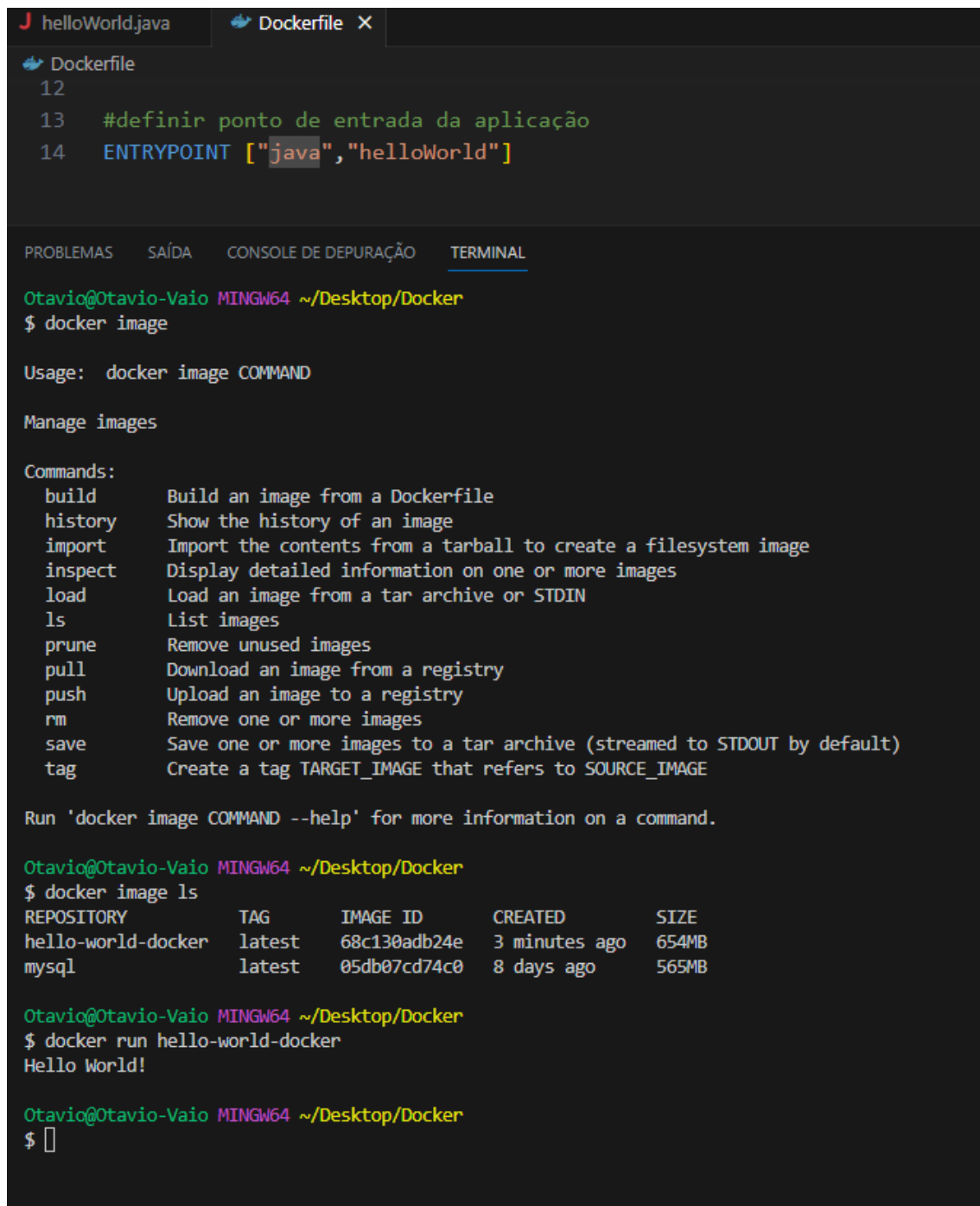


[M3S04] – EX4

“Dockerize” sua aplicação “hello world” em java.

1. Crie uma imagem de sua aplicação;
2. Teste sua execução localmente:



The screenshot shows a code editor with two tabs: 'helloWorld.java' and 'Dockerfile'. The 'Dockerfile' tab is active, showing the following content:

```
12
13 #definir ponto de entrada da aplicação
14 ENTRYPOINT ["java", "helloWorld"]
```

Below the code editor, there is a terminal window with the following content:

```
PROBLEMAS  SAÍDA  CONSOLE DE DEPURAÇÃO  TERMINAL

Otavio@Otavio-Vaio MINGW64 ~/Desktop/Docker
$ docker image

Usage:  docker image COMMAND

Manage images

Commands:
  build      Build an image from a Dockerfile
  history    Show the history of an image
  import     Import the contents from a tarball to create a filesystem image
  inspect    Display detailed information on one or more images
  load       Load an image from a tar archive or STDIN
  ls         List images
  prune      Remove unused images
  pull       Download an image from a registry
  push       Upload an image to a registry
  rm         Remove one or more images
  save       Save one or more images to a tar archive (streamed to STDOUT by default)
  tag        Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE

Run 'docker image COMMAND --help' for more information on a command.

Otavio@Otavio-Vaio MINGW64 ~/Desktop/Docker
$ docker image ls
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
hello-world-docker  latest      68c130adb24e  3 minutes ago  654MB
mysql               latest      05db07cd74c0  8 days ago    565MB

Otavio@Otavio-Vaio MINGW64 ~/Desktop/Docker
$ docker run hello-world-docker
Hello World!

Otavio@Otavio-Vaio MINGW64 ~/Desktop/Docker
$
```

Aluno: Otávio Nascimento