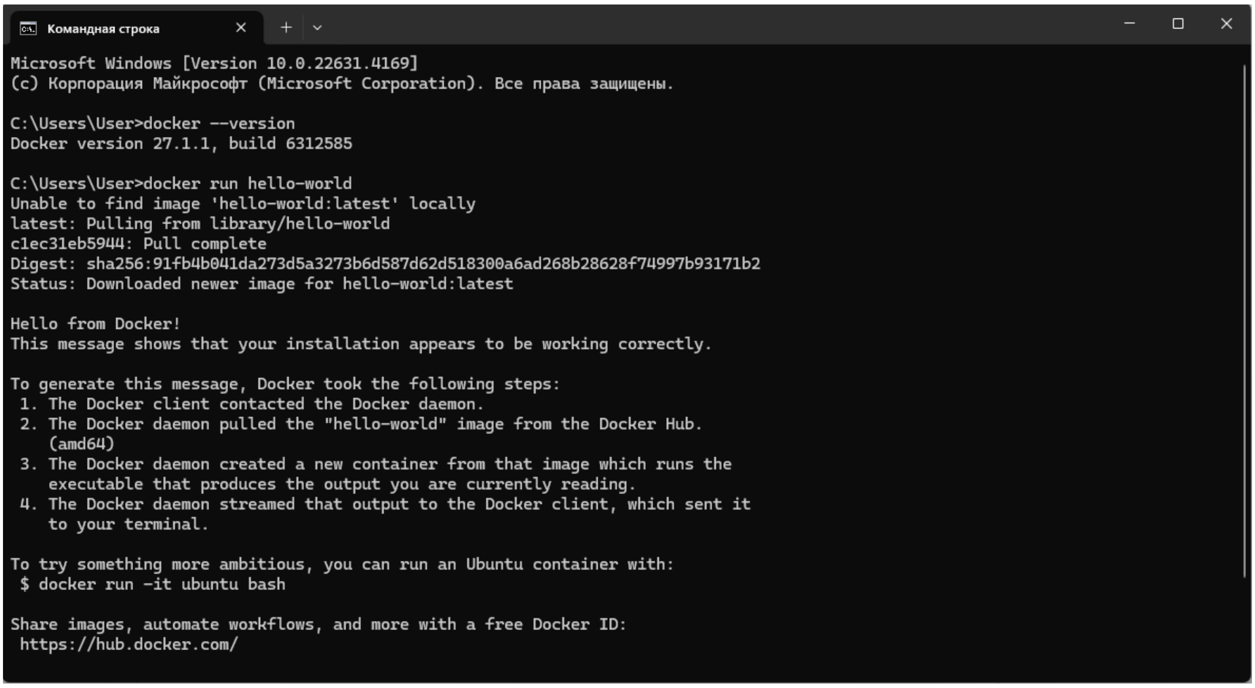
Assignment 1, Web Application Development

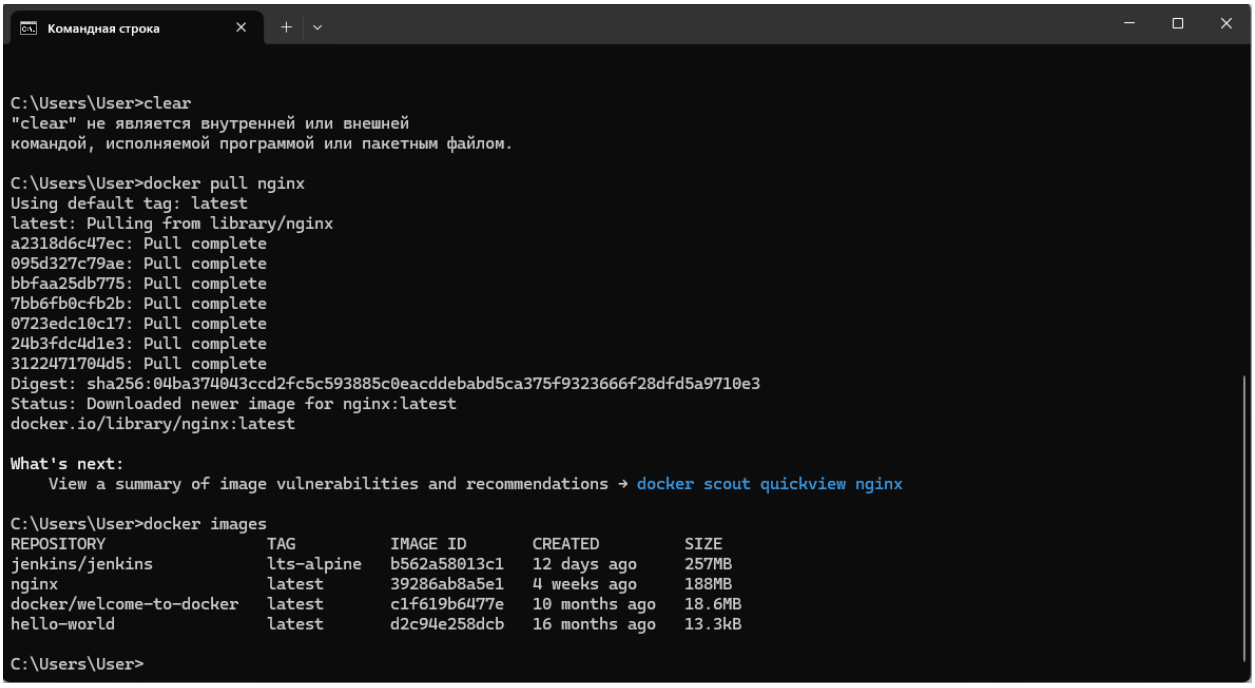
Amanzhol Temirbolat (23MD0382)

Exercise 1: Installing Docker

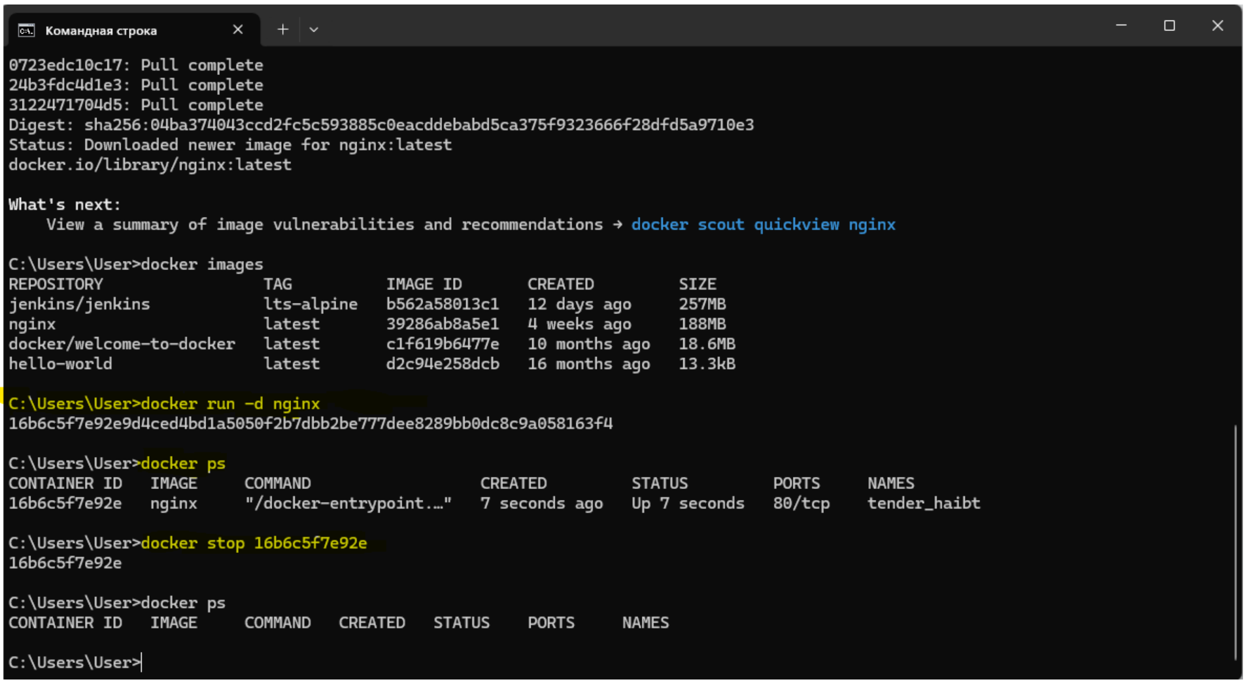


```
Microsoft Windows [Version 10.0.22631.4169]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\User>docker --version
Docker version 27.1.1, build 6312585

C:\Users\User>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:91fb4b041da273d5a3273b6d587d62d518300a6ad268b28628f74997b93171b2
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/
```

Exercise 2: Basic Docker Commands

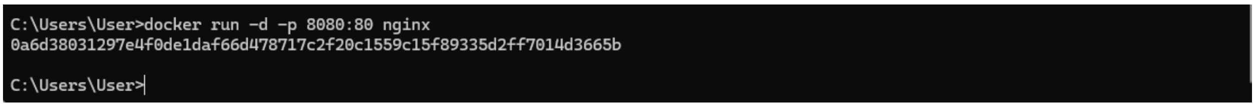


```
C:\Users\User>clear
"clear" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

C:\Users\User>docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
a2318d6c47ec: Pull complete
095d327c79ae: Pull complete
bbfaa25db775: Pull complete
7bb6fb0cfb2b: Pull complete
0723edc10c17: Pull complete
24b3fdc4d1e3: Pull complete
3122471704d5: Pull complete
Digest: sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview nginx

C:\Users\User>docker images
REPOSITORY                 TAG          IMAGE ID       CREATED         SIZE
jenkins/jenkins            lts-alpine   b562a58013c1   12 days ago     257MB
nginx                      latest       39286ab8a5e1   4 weeks ago     188MB
docker/welcome-to-docker   latest       c1f619b6477e   10 months ago   18.6MB
hello-world                latest       d2c94e258dcb   16 months ago   13.3kB

C:\Users\User>
```

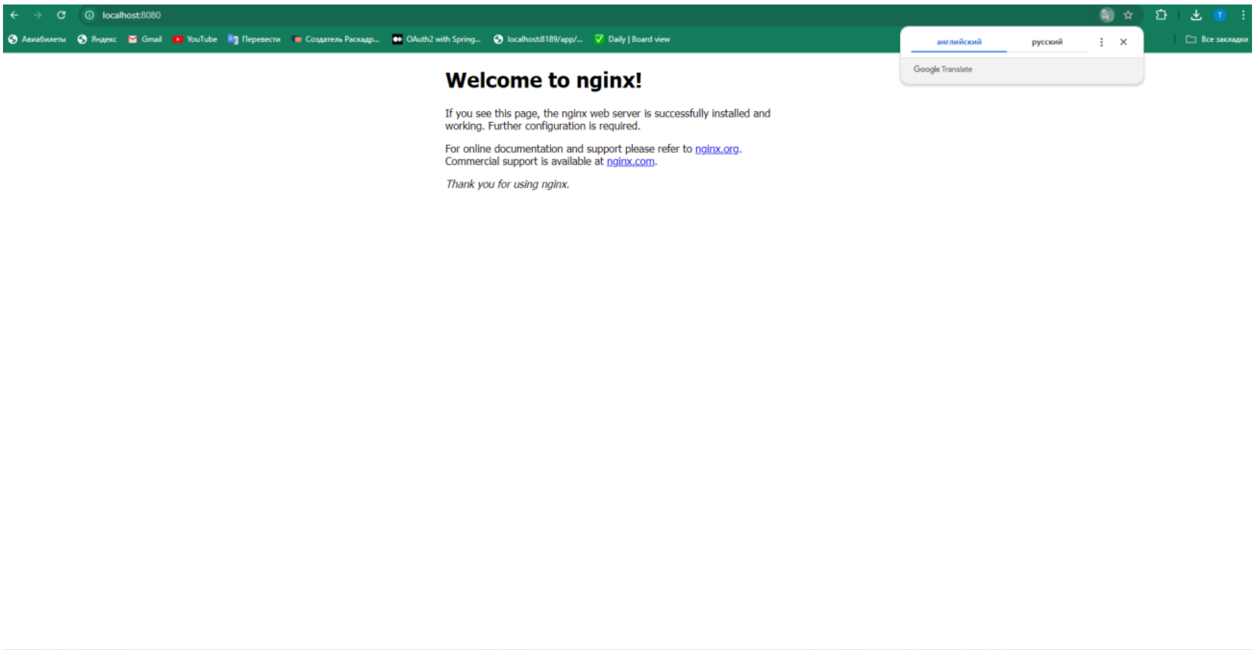


```
0723edc10c17: Pull complete
24b3fdc4d1e3: Pull complete
3122471704d5: Pull complete
Digest: sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview nginx

C:\Users\User>docker images
REPOSITORY               TAG          IMAGE ID       CREATED         SIZE
jenkins/jenkins          lts-alpine   b562a58013c1   12 days ago     257MB
nginx                    latest       39286ab8a5e1   4 weeks ago     188MB
docker/welcome-to-docker latest       c1f619b6477e   10 months ago   18.6MB
hello-world              latest       d2c94e258dcb   16 months ago   13.3kB

C:\Users\User>docker run -d nginx
16b6c5f7e92e9d4ced4bd1a5050f2b7dbb2be777dee8289bb0dc8c9a058163f4

C:\Users\User>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS     NAMES
16b6c5f7e92e   nginx     "/docker-entrypoint.…"   7 seconds ago   Up 7 seconds   80/tcp    tender_haibt

C:\Users\User>docker stop 16b6c5f7e92e
16b6c5f7e92e

C:\Users\User>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

C:\Users\User>
```

Exercise 3: Working with Docker Containers



```
C:\Users\User>docker run -d -p 8080:80 nginx
0a6d38031297e4f0de1daf66d478717c2f20c1559c15f89335d2ff7014d3665b

C:\Users\User>
```

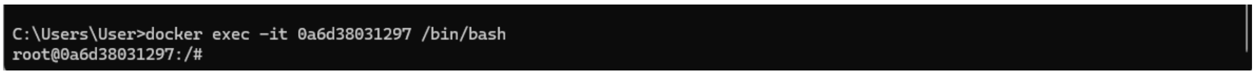


Welcome to nginx!

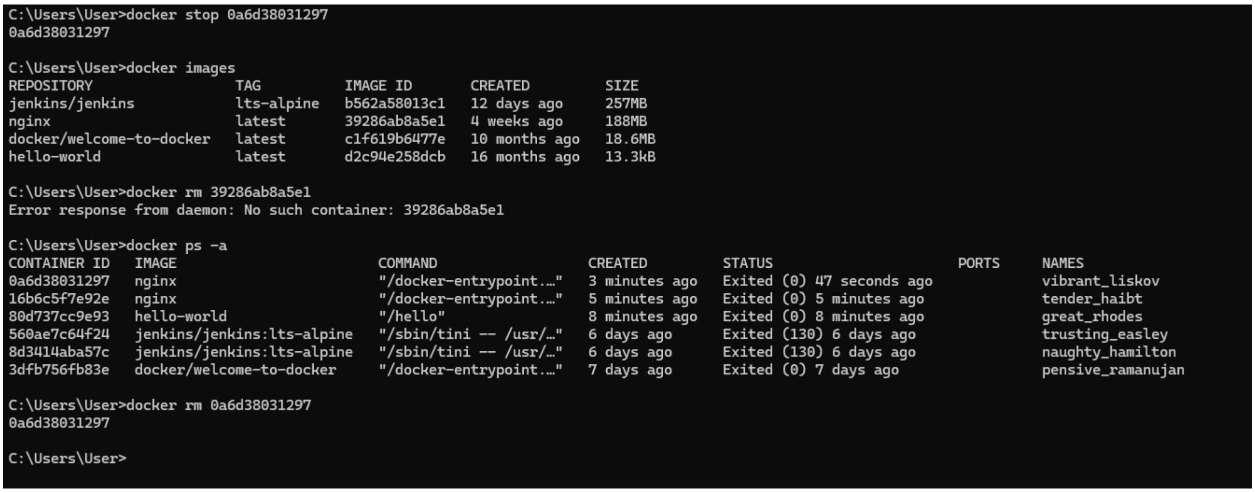
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



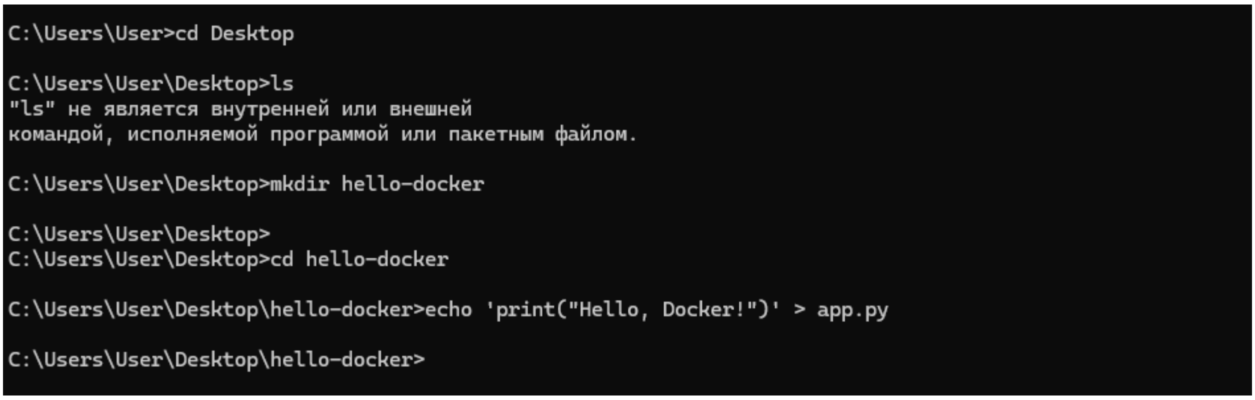
```
C:\Users\User>docker exec -it 0a6d38031297 /bin/bash
root@0a6d38031297:/#
```



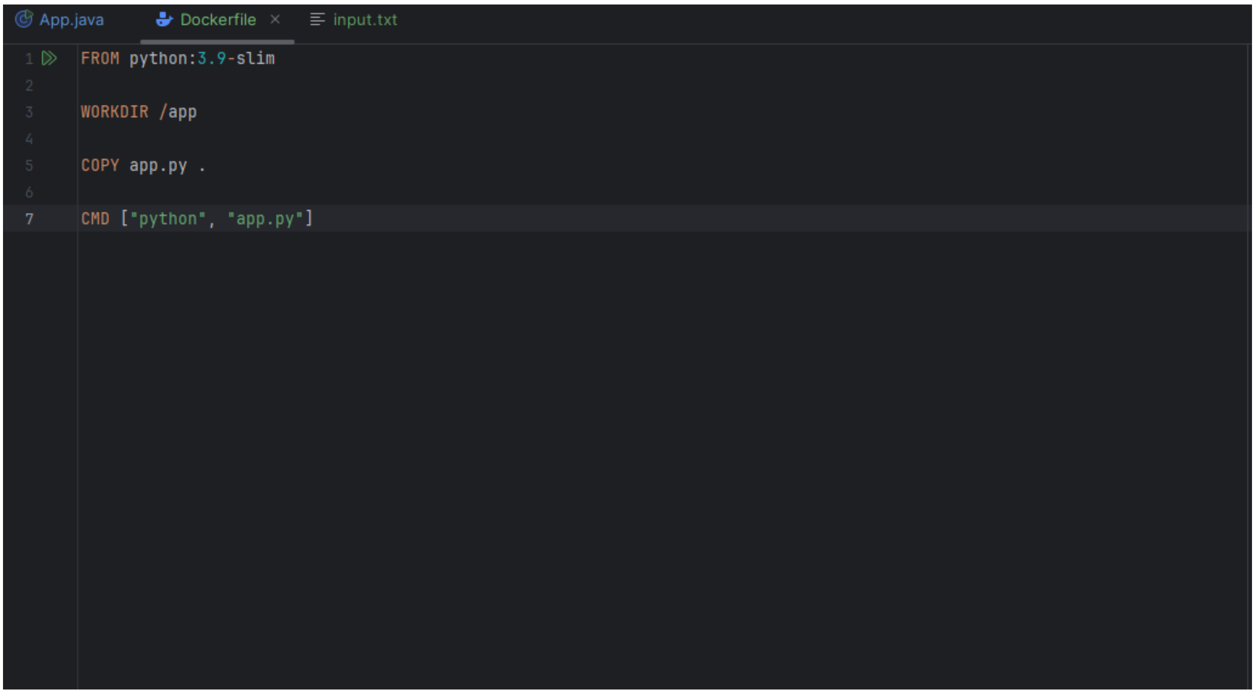
```
C:\Users\User>docker stop 0a6d38031297
0a6d38031297

C:\Users\User>docker images
REPOSITORY                 TAG          IMAGE ID       CREATED         SIZE
jenkins/jenkins            lts-alpine   b562a58013c1   12 days ago     257MB
nginx                      latest       39286ab8a5e1   4 weeks ago     188MB
docker/welcome-to-docker   latest       c1f619b6477e   10 months ago   18.6MB
hello-world                latest       d2c94e258dcb   16 months ago   13.3kB

C:\Users\User>docker rm 39286ab8a5e1
Error response from daemon: No such container: 39286ab8a5e1

C:\Users\User>docker ps -a
CONTAINER ID   IMAGE                        COMMAND                  CREATED         STATUS                        PORTS      NAMES
0a6d38031297   nginx                        "/docker-entrypoint.…"   3 minutes ago   Exited (0) 47 seconds ago                vibrant_liskov
16b6c5f7e92e   nginx                        "/docker-entrypoint.…"   5 minutes ago   Exited (0) 5 minutes ago                 tender_haibt
80d737cc9e93   hello-world                  "/hello"                 8 minutes ago   Exited (0) 8 minutes ago                 great_rhodes
560ae7c64f24   jenkins/jenkins:lts-alpine   "/sbin/tini -- /usr/…"   6 days ago      Exited (130) 6 days ago                  trusting_easley
8d3414aba57c   jenkins/jenkins:lts-alpine   "/sbin/tini -- /usr/…"   6 days ago      Exited (130) 6 days ago                  naughty_hamilton
3dfb756fb83e   docker/welcome-to-docker     "/docker-entrypoint.…"   7 days ago      Exited (0) 7 days ago                    pensive_ramanujan

C:\Users\User>docker rm 0a6d38031297
0a6d38031297

C:\Users\User>
```

Dockerfile

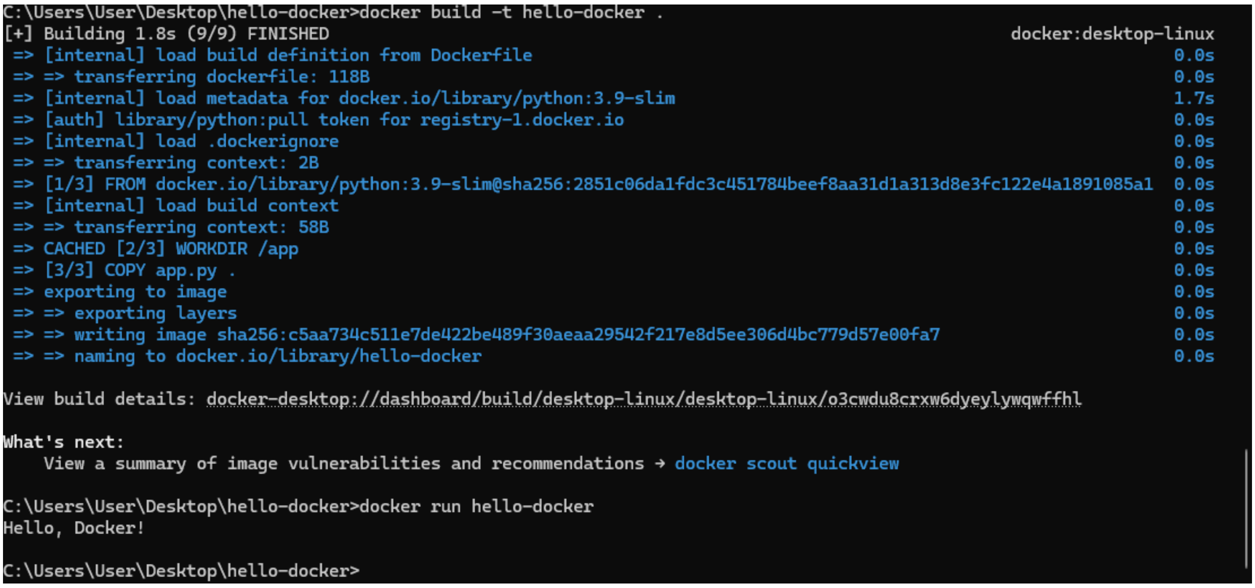
Exercise 1: Creating a Simple Dockerfile



```
C:\Users\User>cd Desktop

C:\Users\User\Desktop>ls
"ls" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

C:\Users\User\Desktop>mkdir hello-docker

C:\Users\User\Desktop>
C:\Users\User\Desktop>cd hello-docker

C:\Users\User\Desktop\hello-docker>echo 'print("Hello, Docker!")' > app.py

C:\Users\User\Desktop\hello-docker>
```

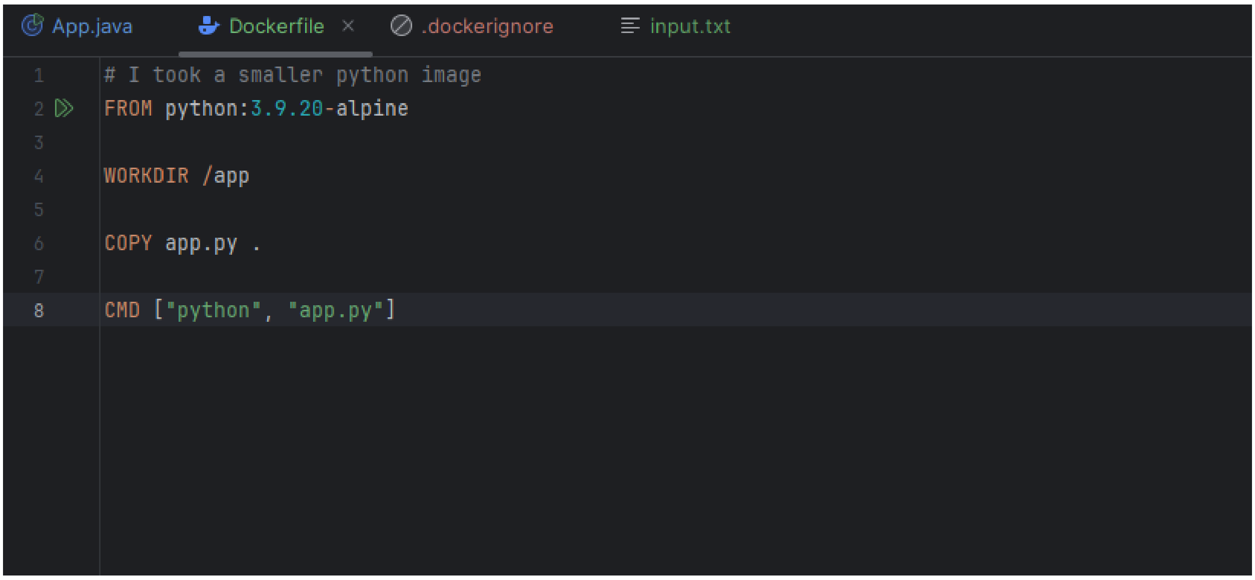


```
FROM python:3.9-slim

WORKDIR /app

COPY app.py .

CMD ["python", "app.py"]
```

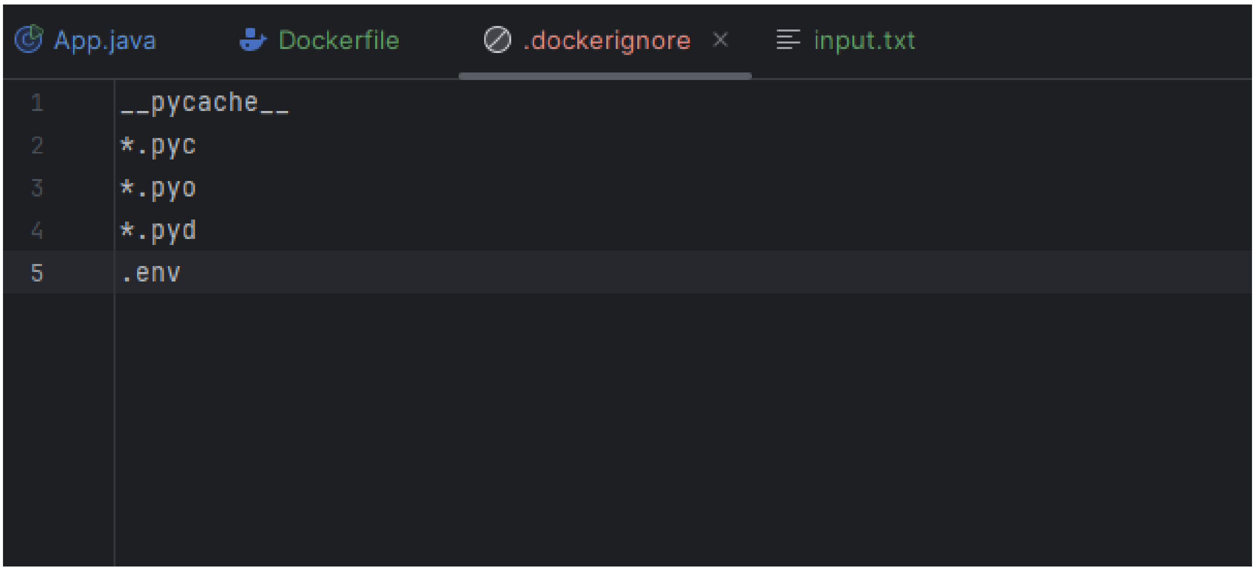


```
C:\Users\User\Desktop\hello-docker>docker build -t hello-docker .
[+] Building 1.8s (9/9) FINISHED                                                                  docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                              0.0s
 => => transferring dockerfile: 118B                                                                              0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim                                                1.7s
 => [auth] library/python:pull token for registry-1.docker.io                                                     0.0s
 => [internal] load .dockerignore                                                                                 0.0s
 => => transferring context: 2B                                                                                   0.0s
 => [1/3] FROM docker.io/library/python:3.9-slim@sha256:2851c06da1fdc3c451784beef8aa31d1a313d8e3fc122e4a1891085a1 0.0s
 => [internal] load build context                                                                                 0.0s
 => => transferring context: 58B                                                                                  0.0s
 => CACHED [2/3] WORKDIR /app                                                                                     0.0s
 => [3/3] COPY app.py .                                                                                           0.0s
 => exporting to image                                                                                            0.0s
 => => exporting layers                                                                                           0.0s
 => => writing image sha256:c5aa734c511e7de422be489f30aeaa29542f217e8d5ee306d4bc779d57e00fa7                      0.0s
 => => naming to docker.io/library/hello-docker                                                                   0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/o3cwdu8crxw6dyeylywqwffhl

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\Users\User\Desktop\hello-docker>docker run hello-docker
Hello, Docker!

C:\Users\User\Desktop\hello-docker>
```

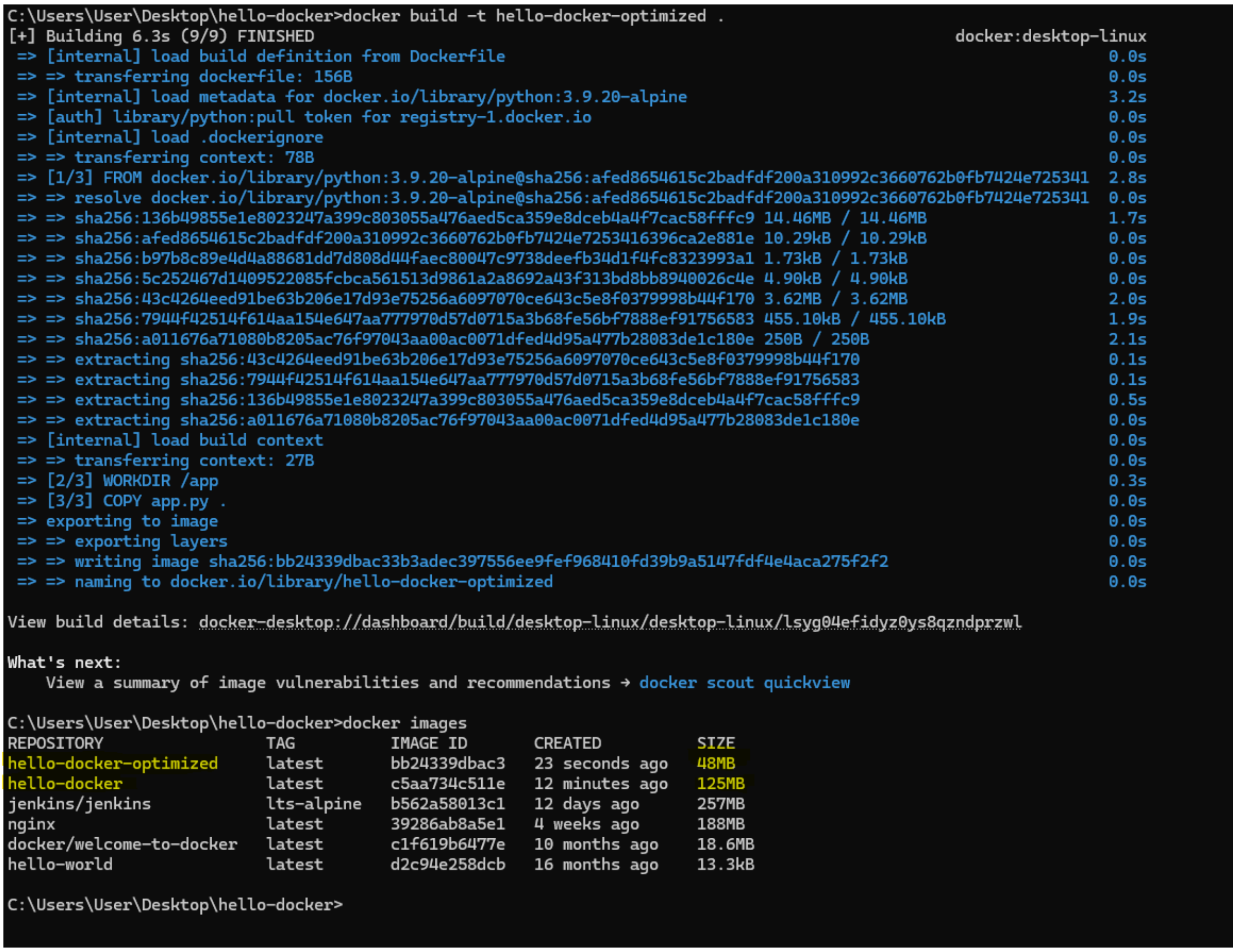
Exercise 2: Optimizing Dockerfile with Layers and Caching



```dockerfile
# I took a smaller python image
FROM python:3.9.20-alpine

WORKDIR /app

COPY app.py .

CMD ["python", "app.py"]
```



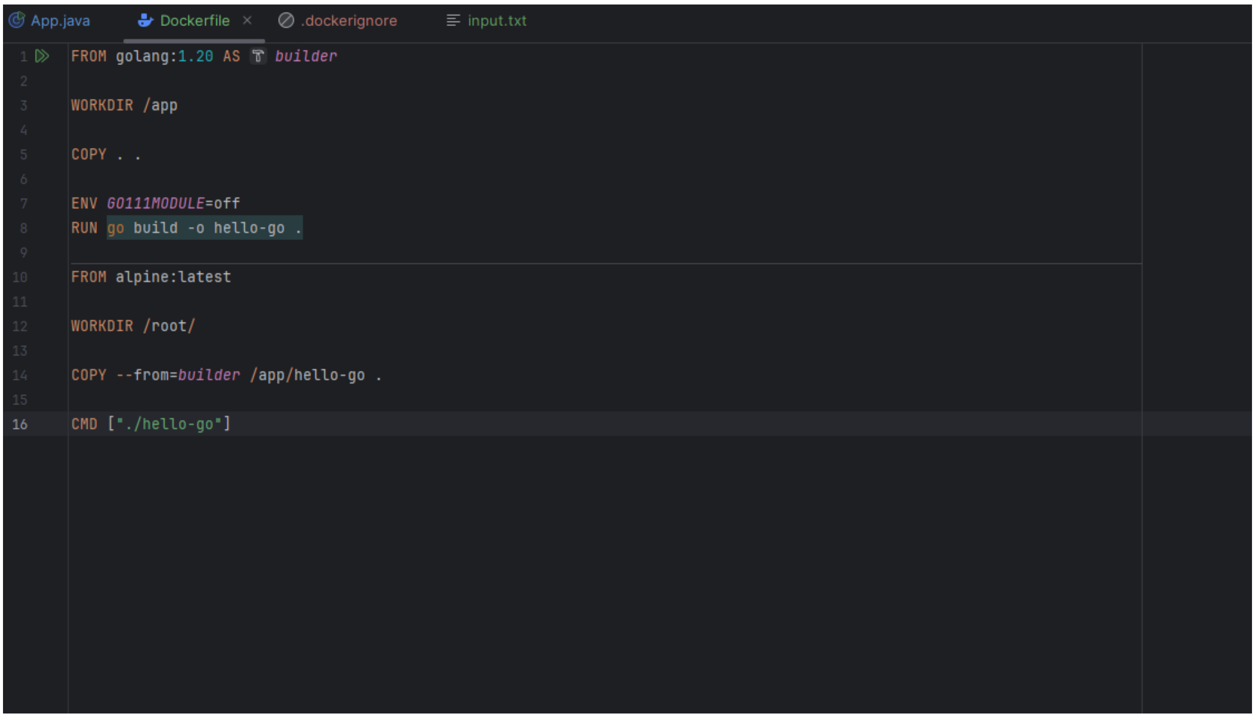
```
__pycache__
*.pyc
*.pyo
*.pyd
.env
```



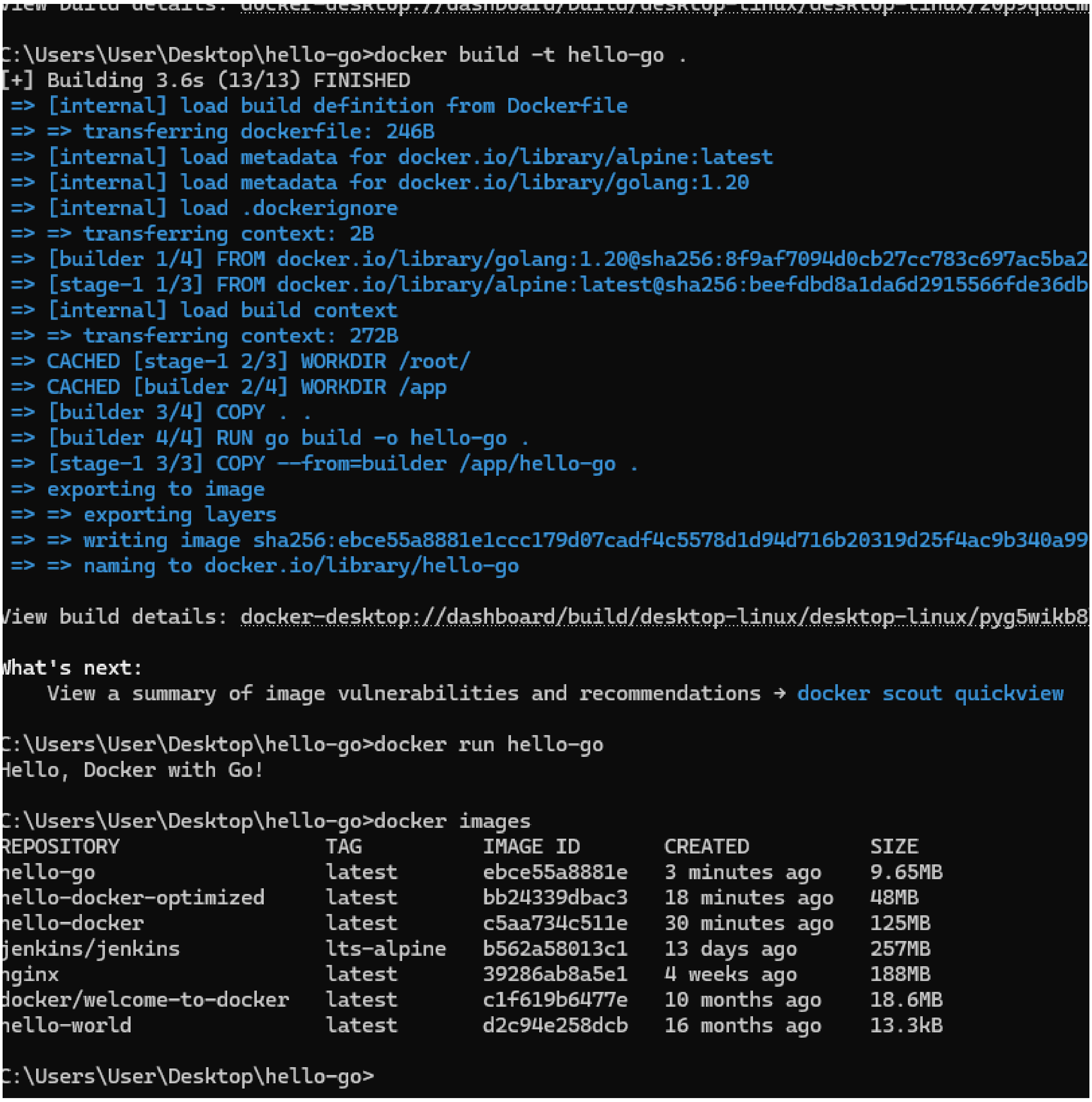
```
C:\Users\User\Desktop\hello-docker>docker build -t hello-docker-optimized .
[+] Building 6.3s (9/9) FINISHED                                                                  docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                              0.0s
 => => transferring dockerfile: 156B                                                                              0.0s
 => [internal] load metadata for docker.io/library/python:3.9.20-alpine                                           3.2s
 => [auth] library/python:pull token for registry-1.docker.io                                                     0.0s
 => [internal] load .dockerignore                                                                                 0.0s
 => => transferring context: 78B                                                                                  0.0s
 => [1/3] FROM docker.io/library/python:3.9.20-alpine@sha256:afed8654615c2badfdf200a310992c3660762b0fb7424e725341 2.8s
 => => resolve docker.io/library/python:3.9.20-alpine@sha256:afed8654615c2badfdf200a310992c3660762b0fb7424e725341 0.0s
 => => sha256:136b49855e1e8023247a399c803055a476aed5ca359e8dceb4a4f7cac58fffc9 14.46MB / 14.46MB                  1.7s
 => => sha256:afed8654615c2badfdf200a310992c3660762b0fb7424e7253416396ca2e881e 10.29kB / 10.29kB                  0.0s
 => => sha256:b97b8c89e4d4a88681dd7d808d44faec80047c9738deefb34d1f4fc8323993a1 1.73kB / 1.73kB                    0.0s
 => => sha256:5c252467d1409522085fcbca561513d9861a2a8692a43f313bd8bb8940026c4e 4.90kB / 4.90kB                    0.0s
 => => sha256:43c4264eed91be63b206e17d93e75256a6097070ce643c5e8f0379998b44f170 3.62MB / 3.62MB                    2.0s
 => => sha256:7944f42514f614aa154e647aa777970d57d0715a3b68fe56bf7888ef91756583 455.10kB / 455.10kB                1.9s
 => => sha256:a011676a71080b8205ac76f97043aa00ac0071dfed4d95a477b28083de1c180e 250B / 250B                        2.1s
 => => extracting sha256:43c4264eed91be63b206e17d93e75256a6097070ce643c5e8f0379998b44f170                         0.1s
 => => extracting sha256:7944f42514f614aa154e647aa777970d57d0715a3b68fe56bf7888ef91756583                         0.1s
 => => extracting sha256:136b49855e1e8023247a399c803055a476aed5ca359e8dceb4a4f7cac58fffc9                         0.5s
 => => extracting sha256:a011676a71080b8205ac76f97043aa00ac0071dfed4d95a477b28083de1c180e                         0.0s
 => [internal] load build context                                                                                 0.0s
 => => transferring context: 27B                                                                                  0.0s
 => [2/3] WORKDIR /app                                                                                            0.3s
 => [3/3] COPY app.py .                                                                                           0.0s
 => exporting to image                                                                                            0.0s
 => => exporting layers                                                                                           0.0s
 => => writing image sha256:bb24339dbac33b3adec397556ee9fef968410fd39b9a5147fdf4e4aca275f2f2                      0.0s
 => => naming to docker.io/library/hello-docker-optimized                                                         0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/lsyg04efidyz0ys8qzndprzwl

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\Users\User\Desktop\hello-docker>docker images
REPOSITORY                 TAG          IMAGE ID       CREATED          SIZE
hello-docker-optimized     latest       bb24339dbac3   23 seconds ago   48MB
hello-docker               latest       c5aa734c511e   12 minutes ago   125MB
jenkins/jenkins            lts-alpine   b562a58013c1   12 days ago      257MB
nginx                      latest       39286ab8a5e1   4 weeks ago      188MB
docker/welcome-to-docker   latest       c1f619b6477e   10 months ago    18.6MB
hello-world                latest       d2c94e258dcb   16 months ago    13.3kB

C:\Users\User\Desktop\hello-docker>
```

Exercise 3: Multi-Stage Builds

Multi-stage:

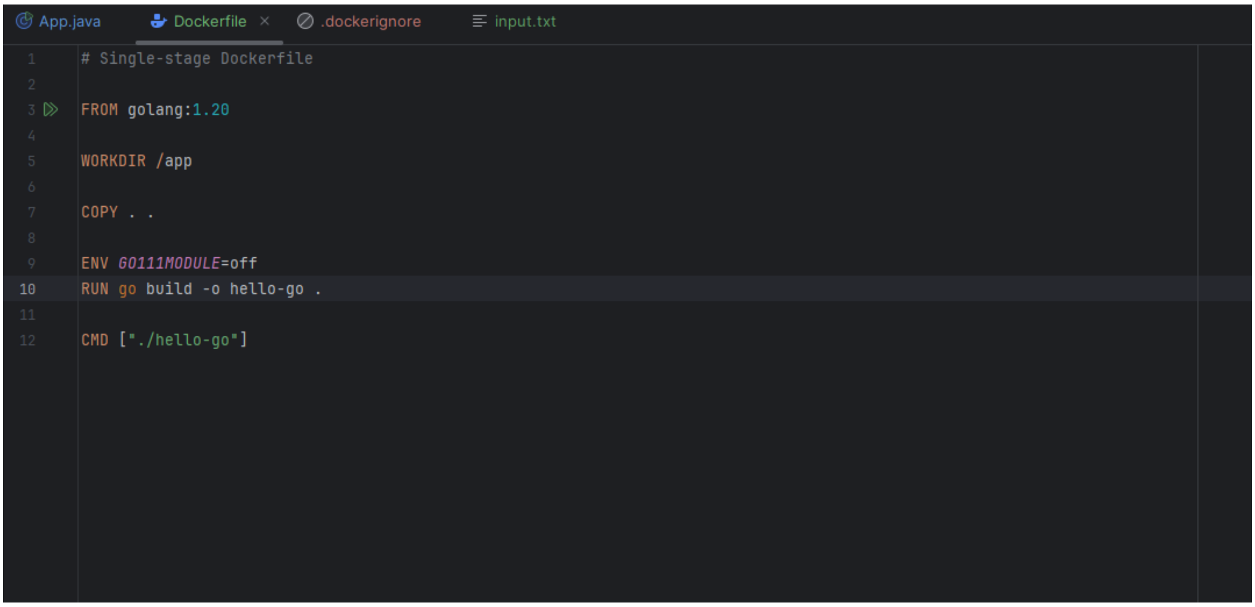


```dockerfile
FROM golang:1.20 AS builder

WORKDIR /app

COPY . .

ENV GO111MODULE=off
RUN go build -o hello-go .

FROM alpine:latest

WORKDIR /root/

COPY --from=builder /app/hello-go .

CMD ["./hello-go"]
```

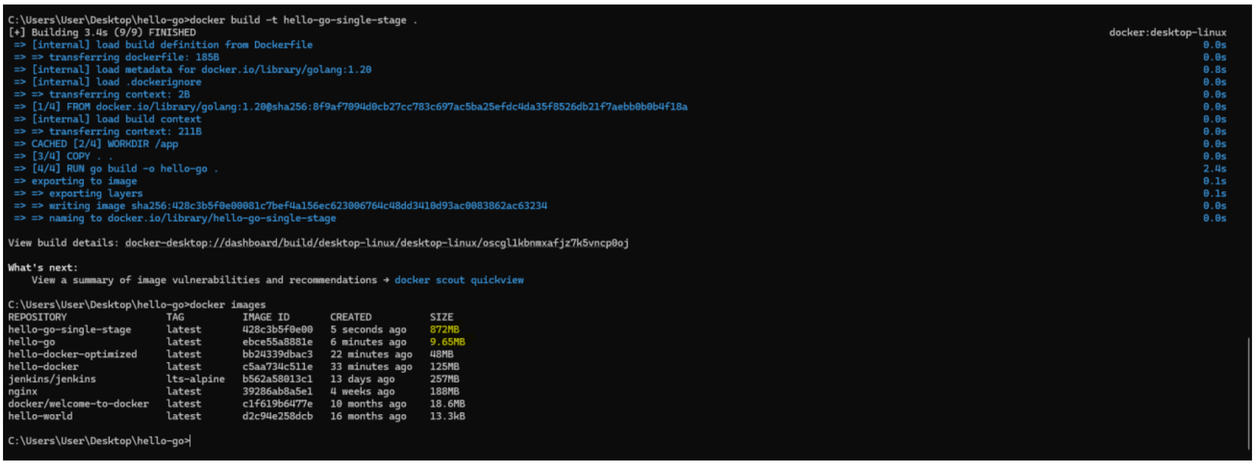


```
View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/z6p9qu8cm

C:\Users\User\Desktop\hello-go>docker build -t hello-go .
[+] Building 3.6s (13/13) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 246B
 => [internal] load metadata for docker.io/library/alpine:latest
 => [internal] load metadata for docker.io/library/golang:1.20
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [builder 1/4] FROM docker.io/library/golang:1.20@sha256:8f9af7094d0cb27cc783c697ac5ba2
 => [stage-1 1/3] FROM docker.io/library/alpine:latest@sha256:beefdbd8a1da6d2915566fde36db
 => [internal] load build context
 => => transferring context: 272B
 => CACHED [stage-1 2/3] WORKDIR /root/
 => CACHED [builder 2/4] WORKDIR /app
 => [builder 3/4] COPY . .
 => [builder 4/4] RUN go build -o hello-go .
 => [stage-1 3/3] COPY --from=builder /app/hello-go .
 => exporting to image
 => => exporting layers
 => => writing image sha256:ebce55a8881e1ccc179d07cadf4c5578d1d94d716b20319d25f4ac9b340a99
 => => naming to docker.io/library/hello-go

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/pyg5wikb8

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\Users\User\Desktop\hello-go>docker run hello-go
Hello, Docker with Go!

C:\Users\User\Desktop\hello-go>docker images
REPOSITORY                TAG          IMAGE ID       CREATED          SIZE
hello-go                  latest       ebce55a8881e   3 minutes ago    9.65MB
hello-docker-optimized    latest       bb24339dbac3   18 minutes ago   48MB
hello-docker              latest       c5aa734c511e   30 minutes ago   125MB
jenkins/jenkins           lts-alpine   b562a58013c1   13 days ago      257MB
nginx                     latest       39286ab8a5e1   4 weeks ago      188MB
docker/welcome-to-docker  latest       c1f619b6477e   10 months ago    18.6MB
hello-world               latest       d2c94e258dcb   16 months ago    13.3kB

C:\Users\User\Desktop\hello-go>
```

Single-stage:

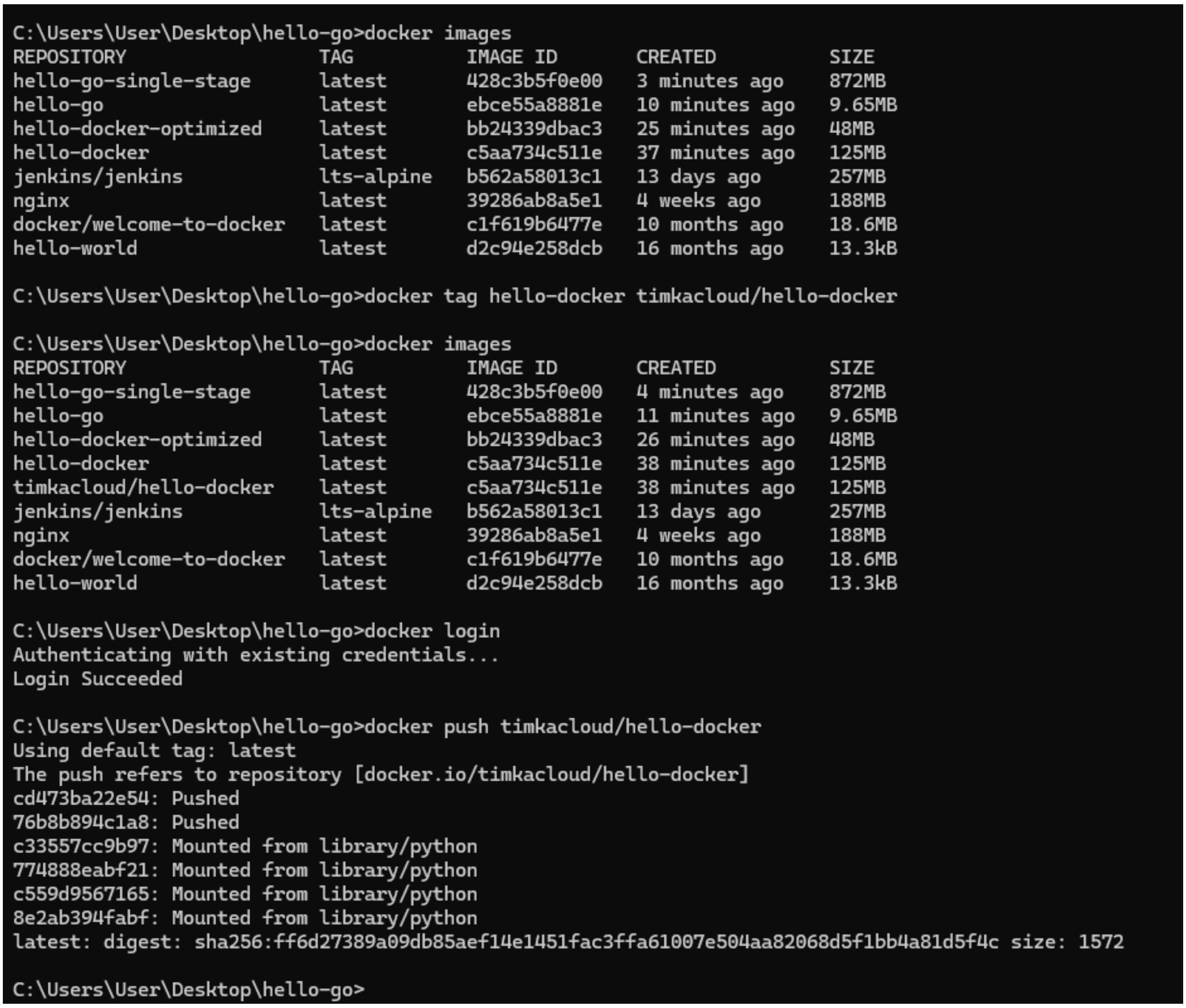


```
# Single-stage Dockerfile

FROM golang:1.20

WORKDIR /app

COPY . .

ENV GO111MODULE=off
RUN go build -o hello-go .

CMD ["./hello-go"]
```



```
C:\Users\User\Desktop\hello-go>docker build -t hello-go-single-stage .
[+] Building 3.4s (9/9) FINISHED                                                                          docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                      0.0s
 => => transferring dockerfile: 185B                                                                                      0.0s
 => [internal] load metadata for docker.io/library/golang:1.20                                                            0.8s
 => [internal] load .dockerignore                                                                                         0.0s
 => => transferring context: 2B                                                                                           0.0s
 => [1/4] FROM docker.io/library/golang:1.20@sha256:8f9af7094d0cb27cc783c697ac5ba25efdc4da35f8526db21f7aebb0b0b4f18a      0.0s
 => [internal] load build context                                                                                         0.0s
 => => transferring context: 211B                                                                                         0.0s
 => CACHED [2/4] WORKDIR /app                                                                                             0.0s
 => [3/4] COPY . .                                                                                                        0.0s
 => [4/4] RUN go build -o hello-go .                                                                                      2.4s
 => exporting to image                                                                                                    0.1s
 => => exporting layers                                                                                                   0.1s
 => => writing image sha256:428c3b5f0e00081c7bef4a156ec623006764c48dd3410d93ac0083862ac63234                              0.0s
 => => naming to docker.io/library/hello-go-single-stage                                                                  0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/oscgl1kbnmxafjz7k5vncp0oj

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\Users\User\Desktop\hello-go>docker images
REPOSITORY                TAG          IMAGE ID       CREATED          SIZE
hello-go-single-stage     latest       428c3b5f0e00   5 seconds ago    872MB
hello-go                  latest       ebce55a8881e   6 minutes ago    9.65MB
hello-docker-optimized    latest       bb24339dbac3   22 minutes ago   48MB
hello-docker              latest       c5aa734c511e   33 minutes ago   125MB
jenkins/jenkins           lts-alpine   b562a58013c1   13 days ago      257MB
nginx                     latest       39286ab8a5e1   4 weeks ago      188MB
docker/welcome-to-docker  latest       c1f619b6477e   10 months ago    18.6MB
hello-world               latest       d2c94e258dcb   16 months ago    13.3kB

C:\Users\User\Desktop\hello-go>
```

Exercise 4: Pushing Docker Images to Docker Hub



```
C:\Users\User\Desktop\hello-go>docker images
REPOSITORY                TAG          IMAGE ID       CREATED          SIZE
hello-go-single-stage     latest       428c3b5f0e00   3 minutes ago    872MB
hello-go                  latest       ebce55a8881e   10 minutes ago   9.65MB
hello-docker-optimized    latest       bb24339dbac3   25 minutes ago   48MB
hello-docker              latest       c5aa734c511e   37 minutes ago   125MB
jenkins/jenkins           lts-alpine   b562a58013c1   13 days ago      257MB
nginx                     latest       39286ab8a5e1   4 weeks ago      188MB
docker/welcome-to-docker  latest       c1f619b6477e   10 months ago    18.6MB
hello-world               latest       d2c94e258dcb   16 months ago    13.3kB

C:\Users\User\Desktop\hello-go>docker tag hello-docker timkacloud/hello-docker

C:\Users\User\Desktop\hello-go>docker images
REPOSITORY                TAG          IMAGE ID       CREATED          SIZE
hello-go-single-stage     latest       428c3b5f0e00   4 minutes ago    872MB
hello-go                  latest       ebce55a8881e   11 minutes ago   9.65MB
hello-docker-optimized    latest       bb24339dbac3   26 minutes ago   48MB
hello-docker              latest       c5aa734c511e   38 minutes ago   125MB
timkacloud/hello-docker   latest       c5aa734c511e   38 minutes ago   125MB
jenkins/jenkins           lts-alpine   b562a58013c1   13 days ago      257MB
nginx                     latest       39286ab8a5e1   4 weeks ago      188MB
docker/welcome-to-docker  latest       c1f619b6477e   10 months ago    18.6MB
hello-world               latest       d2c94e258dcb   16 months ago    13.3kB

C:\Users\User\Desktop\hello-go>docker login
Authenticating with existing credentials...
Login Succeeded

C:\Users\User\Desktop\hello-go>docker push timkacloud/hello-docker
Using default tag: latest
The push refers to repository [docker.io/timkacloud/hello-docker]
cd473ba22e54: Pushed
76b8b894c1a8: Pushed
c33557cc9b97: Mounted from library/python
774888eabf21: Mounted from library/python
c559d9567165: Mounted from library/python
8e2ab394fabf: Mounted from library/python
latest: digest: sha256:ff6d27389a09db85aef14e1451fac3ffa61007e504aa82068d5f1bb4a81d5f4c size: 1572

C:\Users\User\Desktop\hello-go>
```

https://hub.docker.com/repository/docker/timkacloud/hello-docker/general