

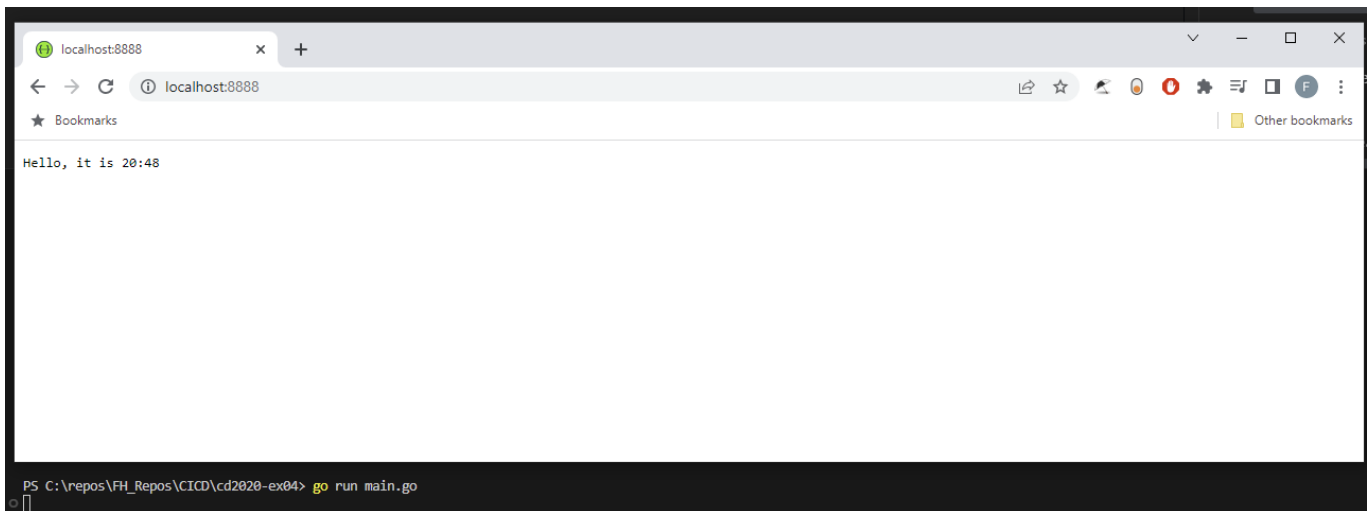
Exercise 4

Part 1

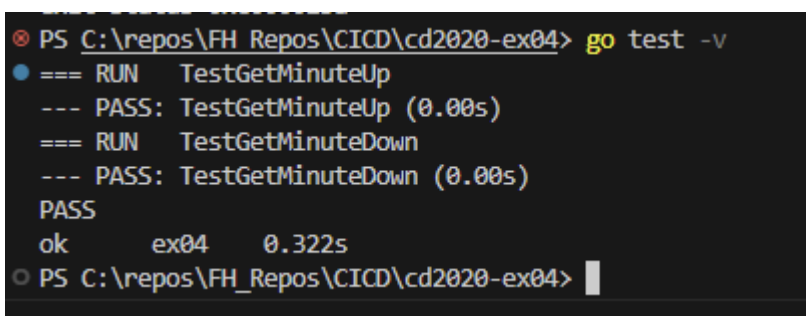
1. Check if the go program runs locally

First create a go.mod file (go init [NAME]).

```
go run main.go
```



```
go test -v
```



2. Modify the dockerfile in the repo

```
FROM golang:1.20-alpine

# Set maintainer label: maintainer=[YOUR-EMAIL]
LABEL maintainer="[s2210455001@fhooe.at]"

# Set working directory: `/src`
WORKDIR /src
```

```
# Copy local file `main.go` to the working directory
COPY *.go go.* ./

# List items in the working directory (ls)
RUN ls -al

# Execute mod init
# RUN go mod init ex04

# Execute tests
RUN CGO_ENABLED=0 go test -v

# Build the GO app as myapp binary and move it to /usr/
RUN CGO_ENABLED=0 go build -o /usr/myapp
# Expose port 8888
EXPOSE 8888

# Run the service myapp when a container of this image is launched
CMD ["/usr/myapp"]
```

3. Build a docker image based on your dockerfile

```
docker image build -f Dockerfile -t dejavu99/my-first-image:0.0.1 ./
```

```
PS C:\repos\FH_Repos\CICD\cd2020-ex04> docker image build -f Dockerfile -t dejavu99/my-first-image:0.0.1 ./
[+] Building 21.3s (12/12) FINISHED
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 659B
=> [internal] load metadata for docker.io/library/golang:1.20-alpine
=> [auth] library/golang:pull token for registry-1.docker.io
=> [1/6] FROM docker.io/library/golang:1.20-alpine@sha256:913de96707b0460bcfdfe422796bb6e559fc300f6c53286777805a9a3010a5ea
=> => resolve docker.io/library/golang:1.20-alpine@sha256:913de96707b0460bcfdfe422796bb6e559fc300f6c53286777805a9a3010a5ea
=> => sha256:85791d961cd3578144eb3d64a9716fa35c198c4ae9202b07eb0f2de9127a7907 284.81kB / 284.81kB
=> => sha256:1b68059151f910b7f3b17d63a4798e99b3f190fd39f5a9539ed8529b3b00a7e0 100.88MB / 100.88MB
=> => sha256:3ecf7abccccfebb18597cfaec86e4554f2fecdc572ae6626544a18db40302e8f1 156B / 156B
=> => sha256:913de96707b0460bcfdfe422796bb6e559fc300f6c53286777805a9a3010a5ea 1.65kB / 1.65kB
=> => sha256:4ee203ff3933e7a6f18d3574fd6661a73b58c60f028d2927274400f4774aaa41 1.16kB / 1.16kB
=> => sha256:12d7310e12ea74b4e79765c1ad6340d039100cd01b2d292969a9756ffd629dc4 5.11kB / 5.11kB
=> => extracting sha256:85791d961cd3578144eb3d64a9716fa35c198c4ae9202b07eb0f2de9127a7907
=> => extracting sha256:1b68059151f910b7f3b17d63a4798e99b3f190fd39f5a9539ed8529b3b00a7e0
=> => extracting sha256:3ecf7abccccfebb18597cfaec86e4554f2fecdc572ae6626544a18db40302e8f1
=> [internal] load build context
=> => transferring context: 116B
=> [2/6] WORKDIR /src
=> [3/6] COPY *.go go.* ./
=> [4/6] RUN ls -al
=> [5/6] RUN CGO_ENABLED=0 go test -v
=> [6/6] RUN CGO_ENABLED=0 go build -o /usr/myapp
=> exporting to image
=> => exporting layers
=> => writing image sha256:ceea8a5a7ae507a707392c46d4081445d373bcfc15e3c6cbda3565027b8ad319
=> => naming to docker.io/dejavu99/my-first-image:0.0.1
PS C:\repos\FH_Repos\CICD\cd2020-ex04> |
```

4. List all images that are stored in your local registry


```
docker images
```


```
PS C:\repos\FH_Repos\CICD\cd2020-ex04> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
dejavu99/my-first-image  0.0.1              cee8a8a5a7ae5      2 minutes ago      333MB
```


- 5. Authenticate to the container registry
- 6. Push the created image to your DockerHub account

```
docker image push dejavu99/my-first-image:0.0.1
```



- 7. Verify deployed image

 **dejavu99 / my-first-image**

Description
This repository does not have a description 

 Last pushed: 2 days ago

Tags
This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
 0.0.1		Image	2 days ago	2 days ago

[See all](#)[Go to Advanced Image Management](#)

Part 2

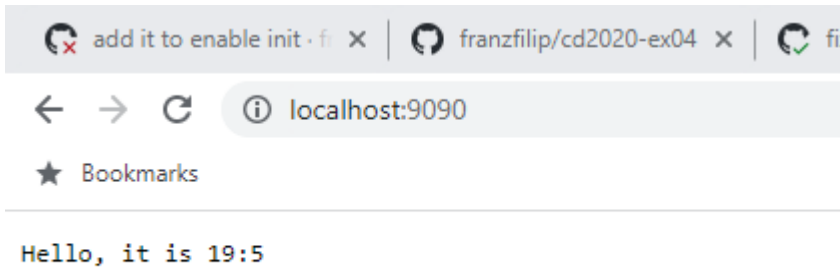
- 1. Create image from the provided Dockerfile

```
docker image build -t dejavu99/myhello:0.0.1 ./
```

- 2. Run image

```
docker run --rm -p 9090:8888 dejavu99/myhello:0.0.1
```

3. Check if application is running on localhost:9090



4. See your container running on your local Docker daemon

```
docker ps
```

```
PS C:\Users\Franz-Filip> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
b7bf00044aa3   dejavu99/myhello:0.0.1  "/usr/myapp"           5 seconds ago  Up 4 seconds  0.0.0.0:9090->8888/tcp    sweet_lovelace
PS C:\Users\Franz-Filip>
```

5. Stop your container

```
docker stop b7bf00044aa3
```

Part 3

Created with github actions in separate steps:

- build: Run go tests
- docker: Build image and deploy to docker hub

```
name: Docker Image CI

on:
  push:
    branches: [ "master" ]
  pull_request:
    branches: [ "master" ]

jobs:
  build:
    name: Build and test application
    runs-on: ubuntu-latest
    steps:
      - uses: actions/setup-go@v4
        with:
```

```
    go-version: 'stable'
  - name: Checkout
    uses: actions/checkout@v3
  - name: Run tests
    run: go test -v

docker:
  name: Build and push Docker image
  needs: build
  runs-on: ubuntu-latest
  steps:
    - name: Checkout
      uses: actions/checkout@v3
    - name: Build and tag Docker image
      run: docker image build -t dejavu99/myhello:${{ github.sha }} -t
dejavu99/myhello:latest ./
    - name: Push Docker image
      uses: docker/login-action@v2
      with:
        username: ${ secrets.DOCKER_USERNAME }
        password: ${ secrets.DOCKER_TOKEN }
    - run: docker image push -a dejavu99/myhello
```

```
docker image pull dejavu99/myhello
```

```
PS C:\Users\Franz-Filip> docker image pull dejavu99/myhello
Using default tag: latest
latest: Pulling from dejavu99/myhello
f56be85fc22e: Already exists
85791d961cd3: Already exists
1b68059151f9: Already exists
3ecf7abcccfe: Already exists
f62c789f8dcc: Pull complete
7619afb3143b: Pull complete
0cf0dfe62ea2: Pull complete
08f09f1c45e9: Pull complete
Digest: sha256:d8f6372514fbe485ef978121e9010ce4b61e04714b810c6a3e4dbf1002920a65
Status: Downloaded newer image for dejavu99/myhello:latest
docker.io/dejavu99/myhello:latest
PS C:\Users\Franz-Filip> |
```

Part 4

Scan docker and code for vulnerabilities.

```
trivy-docker:
  name: "Scan for docker vulnerabilities"
  needs: docker
  runs-on: ubuntu-latest
```

```

steps:
  - name: Run scanner
    uses: aquasecurity/trivy-action@master
    with:
      image-ref: "dejavu99/myhello:latest"
      format: "table"
      exit-code: "1"
      ignore-unfixed: true
      skip-files: "*.go"
      vuln-type: "os,library"
      severity: "CRITICAL"

```



```

trivy-repo:
  name: "Scan for code vulnerabilities"
  runs-on: ubuntu-latest
  steps:
    - name: Checkout
      uses: actions/checkout@v3
    - name: Run scanner
      uses: aquasecurity/trivy-action@master
      with:
        scan-type: "fs"
        scan-ref: "."
        format: "sarif"
        output: "trivy-results.sarif"
        severity: "CRITICAL,HIGH"

```

Successful Pipeline

Triggered via push 25 minutes ago

	Status	Total duration	Artifacts
 franzfilip pushed  a56fbcf master	Success	1m 49s	—

docker-publish.yml
on: push

```

graph LR
    A[✓ Build and test application 22s] --> B[✓ Build and push Docker image 38s]
    B --> C[✓ Scan for docker vulnerabilit... 24s]
    D[✓ Scan for code vulnerabilities 20s]

```

Scan for docker vulnerabilites:

```
Scan for docker vulnerabilities
succeeded 25 minutes ago in 24s

Run scanner

1 ▶ Run aquasecurity/trivy-action@master
13 /usr/bin/docker run --name ed866eb9d813d5ba9845d0ab8003db5d0bc66d_723b42 --label ed866e --workdir /github/workspace --rm -e "INPUT_IMAGE-REF" -e "INPUT_FORMAT" -e "INPUT_EXIT-CODE" -e "INPUT_IGNORE-UNFIXED" -e
"INPUT_SKIP-FILES" -e "INPUT_VULN-TYPE" -e "INPUT_SEVERITY" -e "INPUT_SCAN-TYPE" -e "INPUT_INPUT" -e "INPUT_SCAN-REF" -e "INPUT_TEMPLATE" -e "INPUT_OUTPUT" -e "INPUT_SKIP-DIRS" -e "INPUT_CACHE-DIR" -e
"INPUT_TIMEOUT" -e "INPUT_IGNORE-POLICY" -e "INPUT_HIDE-PROGRESS" -e "INPUT_LIST-ALL-PKGS" -e "INPUT_SCANNERS" -e "INPUT_TRIVYIGNORES" -e "INPUT_ARTIFACT-TYPE" -e "INPUT_GITHUB-PAT" -e "INPUT_TRIVY-CONFIG" -e
"INPUT_LIMIT-SEVERITIES-FOR-SARIF" -e "HOME" -e "GITHUB_JOB" -e "GITHUB_REF" -e "GITHUB_SHA" -e "GITHUB_REPOSITORY" -e "GITHUB_REPOSITORY_OWNER" -e "GITHUB_REPOSITORY_OWNER_ID" -e "GITHUB_RUN_ID" -e
"GITHUB_RUN_NUMBER" -e "GITHUB_RETENTION_DAYS" -e "GITHUB_RUN_ATTEMPT" -e "GITHUB_REPOSITORY_ID" -e "GITHUB_ACTOR_ID" -e "GITHUB_ACTOR" -e "GITHUB_TRIGGERING_ACTOR" -e "GITHUB_WORKFLOW" -e "GITHUB_HEAD_REF" -e
"GITHUB_BASE_REF" -e "GITHUB_EVENT_NAME" -e "GITHUB_SERVER_URL" -e "GITHUB_API_URL" -e "GITHUB_GRAPHQL_URL" -e "GITHUB_REF_NAME" -e "GITHUB_REF_PROTECTED" -e "GITHUB_REF_TYPE" -e "GITHUB_WORKFLOW_REF" -e
"GITHUB_WORKFLOW_SHA" -e "GITHUB_WORKSPACE" -e "GITHUB_ACTION" -e "GITHUB_EVENT_PATH" -e "GITHUB_ACTION_REPOSITORY" -e "GITHUB_ACTION_REF" -e "GITHUB_PATH" -e "GITHUB_ENV" -e "GITHUB_STEP_SUMMARY" -e "GITHUB_STATE"
-e "GITHUB_OUTPUT" -e "RUNNER_OS" -e "RUNNER_ARCH" -e "RUNNER_NAME" -e "RUNNER_TOOL_CACHE" -e "RUNNER_TEMP" -e "RUNNER_WORKSPACE" -e "ACTIONS_RUNTIME_URL" -e "ACTIONS_RUNTIME_TOKEN" -e "ACTIONS_CACHE_URL" -e
GITHUB_ACTIONS=true -e CI=true -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/home/runner/work/_temp/_github_home":"/github/home" -v "/home/runner/work/_temp/_github_workflow":"/github/workflow" -v
"/home/runner/work/_temp/_runner_file_commands":"/github/file_commands" -v "/home/runner/work/cd2020-ex04/cd2020-ex04":"/github/workspace" ed866e:b9d813d5ba9845d0ab8003db5d0bc66d -a image -b table -c " -d 1"
-e true" -f os,library" -g CRITICAL" -h " -i dejavu99/myhello:latest" -j "." -k " -l " -m " -n " -o " -p " -q *.go" -r false" -s " -t " -u " -v " -z "
14 Running trivy with options: trivy image --format table --exit-code 1 --ignore-unfixed --vuln-type os,library --severity CRITICAL --skip-files *.go dejavu99/myhello:latest
15 Global options:
16 2023-05-11T18:45:48.451Z INFO Need to update DB
17 2023-05-11T18:45:48.451Z INFO DB Repository: ghcr.io/aquasecurity/trivy-db
18 2023-05-11T18:45:48.451Z INFO Downloading DB...
19 20.21 MiB / 36.76 MiB [----->] 54.98% > p/s 736.76 MiB / 36.76 MiB [----->] 100.00% > p/s 736.76 MiB /
36.76 MiB [----->] 100.00% 27.55 MiB p/s ETA 0s36.76 MiB / 36.76 MiB [----->] 100.00% 27.55 MiB p/s ETA 0s36.76 MiB / 36.76 MiB [----->] 100.00% 25.77 MiB p/s ETA 0s36.76 MiB / 36.76 MiB [----->] 100.00% 25.77 MiB p/s ETA 0s36.76 MiB / 36.76 MiB [----->] 100.00% 25.77 MiB p/s ETA 0s36.76 MiB / 36.76 MiB [----->] 100.00% 20.45 MiB p/s 2.0s2023-05-11T18:45:51.674Z INFO Vulnerability scanning is enabled
20 2023-05-11T18:45:51.674Z INFO Secret scanning is enabled
21 2023-05-11T18:45:51.674Z INFO If your scanning is slow, please try '--scanners vuln' to disable secret scanning
22 2023-05-11T18:45:51.674Z INFO Please see also https://aquasecurity.github.io/trivy/v0.49/docs/secret/scanning/#recommendation for faster secret detection
23 2023-05-11T18:45:56.118Z INFO Detected OS: alpine
24 2023-05-11T18:45:56.118Z INFO Detecting Alpine vulnerabilities...
25 2023-05-11T18:45:56.120Z INFO Number of language-specific files: 5
26 2023-05-11T18:45:56.120Z INFO Detecting gomod vulnerabilities...
27 2023-05-11T18:45:56.122Z INFO Detecting node-pkg vulnerabilities...
28
29 dejavu99/myhello:latest (alpine 3.17.3)
30 =====
31 Total: 0 (CRITICAL: 0)
32
> Complete job
```