We will start by creating our python app in a file called app.py

TP3 > app.py > main

```python
import os
import platform
import subprocess
from colorama import Fore, Style, init
import random
import time

def set_terminal_size(width, height):
    if platform.system() == "Windows":
        command = f"mode {width},{height}"
        os.system(command)
    else:
        command = f"printf '\e[8;{height};{width}t'"
        subprocess.run(command, shell=True)

init(autoreset=True)  # Initialize colorama

def get_random_color():
    # Returns a random color from colorama Fore
    colors = [Fore.RED, Fore.GREEN, Fore.YELLOW, Fore.BLUE, Fore.MAGENTA, Fore.CYAN]
    return random.choice(colors)

```

(No spoilers for what this app does)

The only external library is colorama, so we create a requirements.txt file like this:

TP3 > requirements.txt

```
colorama==0.4.6

```

Now we will create the Dockerfile. We first move the app.py and requirements.txt file to a directory called Docker for organization purposes. We the create a Dockerfile in this directory and fill it like this:

TP3 > Docker > Dockerfile

```dockerfile
# Use an official Python runtime as a parent image
FROM python:3.9-alpine

# Set the working directory in the container
WORKDIR /usr/src/app

# Copy the current directory contents into the container at /usr/src/app
COPY . .

# Install any needed packages specified in requirements.txt
RUN pip install -r requirements.txt

# Run app.py when the container launches
CMD ["python", "./app.py"]
```

Now that all the files are created, we can start by building our image from the local files.
We navigate to our Docker directory and run the following command:

```
(.venv) cflorval@Clements-MacBook-Pro Docker % docker build -t tp3-alpine .

[+] Building 0.5s (9/9) FINISHED                                                              docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                          0.0s
 => => transferring dockerfile: 429B                                                                          0.0s
 => [internal] load .dockerignore                                                                             0.0s
 => => transferring context: 2B                                                                               0.0s
 => [internal] load metadata for docker.io/library/python:3.9-alpine                                          0.5s
 => [1/4] FROM docker.io/library/python:3.9-alpine@sha256:974669b59630f8d7224a9c92e212d0b05f1b8b8030ff79ed6e3c70f718916409  0.0s
 => [internal] load build context                                                                             0.0s
 => => transferring context: 93B                                                                              0.0s
 => CACHED [2/4] WORKDIR /usr/src/app                                                                         0.0s
 => CACHED [3/4] COPY . .                                                                                     0.0s
 => CACHED [4/4] RUN pip install -r requirements.txt                                                          0.0s
 => exporting to image                                                                                        0.0s
 => => exporting layers                                                                                       0.0s
 => => writing image sha256:42738e9333b80c9600044c3cbfe4f856bb5c9bbd6a250dd66d0efae30f2b2ad9                  0.0s
 => => naming to docker.io/library/tp3-alpine                                                                 0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/dt8s0d303j171hs2qj9op9xha
```

We can see that the build happened without issue!

Now what's left is to publish it to DockerHub.

```
(.venv) cflorval@Clements-MacBook-Pro Docker % docker tag tp3-alpine chargeutile/tp3-alpine

(.venv) cflorval@Clements-MacBook-Pro Docker % docker push chargeutile/tp3-alpine
Using default tag: latest
The push refers to repository [docker.io/chargeutile/tp3-alpine]
9bbea86613a9: Pushed
7ba492ca190e: Pushed
503c6afad766: Pushed
b6811176f364: Mounted from library/python
ed7b5a56a058: Mounted from library/python
d4f73424950b: Mounted from library/python
f94bd46a158c: Mounted from library/python
3ce819cc4970: Mounted from library/python
latest: digest: sha256:f0bf8bca612226eca993d794844cc02293f7e48f683e797087fbe3d2ba3c28b4 size: 1994
```

Now that that's done, we can try and see it it runs without issue…

```
cflorval@Clements-MacBook-Pro ~ % docker run -it --rm --name running-app tp3-alpine

Completez l'expression: C'est vraiment n'importe xxxx !
```

And it does! The app works as expected (no spoilers again…)

You can try it yourself; the repository name is `chargeutile/tp3-alpine`
(Mind you, I am on Mac Silicon, so x86 might cause problems…)

Bonus tasks:

- Have the smallest possible image size: I used python:3.9-alpine to achieve a very small image size of 59MB.
- Run a linter on the Dockerfile: I ran it on Internet and there was no issue found!
- Difference between ADD and COPY:
  - COPY is used to copy files from the local file system into the container.
  - ADD has all capabilities of COPY but also can handle remote URLs and auto-extract compressed files.
- The container runs without sudo rights because of the myuser user created in the Dockerfile.

- Secure scan: I used trivy to run a secure scan:

```
(.venv) cflorval@Clements-MacBook-Pro Docker % trivy image chargeutile/tp3-alpine
2024-01-22T19:13:01.005+0100    INFO    Need to update DB
2024-01-22T19:13:01.005+0100    INFO    DB Repository: ghcr.io/aquasecurity/trivy-db
2024-01-22T19:13:01.005+0100    INFO    Downloading DB...
42.45 MiB / 42.45 MiB [-------------------------------------------------------------------------------------------------------------------------------------------------------] 100.00% 6.42 MiB p/s 6.8s
2024-01-22T19:13:08.639+0100    INFO    Vulnerability scanning is enabled
2024-01-22T19:13:08.639+0100    INFO    Secret scanning is enabled
2024-01-22T19:13:08.639+0100    INFO    If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2024-01-22T19:13:08.639+0100    INFO    Please see also https://aquasecurity.github.io/trivy/v0.48/docs/scanner/secret/#recommendation for faster secret detection
2024-01-22T19:13:09.497+0100    INFO    Detected OS: alpine
2024-01-22T19:13:09.497+0100    WARN    This OS version is not on the EOL list: alpine 3.19
2024-01-22T19:13:09.497+0100    INFO    Detecting Alpine vulnerabilities...
2024-01-22T19:13:09.499+0100    INFO    Number of language-specific files: 1
2024-01-22T19:13:09.499+0100    INFO    Detecting python-pkg vulnerabilities...

chargeutile/tp3-alpine (alpine 3.19.0)

Total: 4 (UNKNOWN: 0, LOW: 0, MEDIUM: 4, HIGH: 0, CRITICAL: 0)

┌──────────────┬────────────────┬──────────┬────────┬───────────────────┬───────────────┬──────────────────────────────────────────────────────────────┐
│   Library    │ Vulnerability  │ Severity │ Status │ Installed Version │ Fixed Version │                            Title                             │
├──────────────┼────────────────┼──────────┼────────┼───────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ libcrypto3   │ CVE-2023-6129  │ MEDIUM   │ fixed  │ 3.1.4-r2          │ 3.1.4-r3      │ openssl: POLY1305 MAC implementation corrupts vector         │
│              │                │          │        │                   │               │ registers on PowerPC                                         │
│              │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2023-6129                    │
│              ├────────────────┤          │        │                   ├───────────────┼──────────────────────────────────────────────────────────────┤
│              │ CVE-2023-6237  │          │        │                   │ 3.1.4-r4      │ openssl: Excessive time spent checking invalid RSA public    │
│              │                │          │        │                   │               │ keys                                                         │
│              │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2023-6237                    │
├──────────────┼────────────────┤          │        │                   ├───────────────┼──────────────────────────────────────────────────────────────┤
│ libssl3      │ CVE-2023-6129  │          │        │                   │ 3.1.4-r3      │ openssl: POLY1305 MAC implementation corrupts vector         │
│              │                │          │        │                   │               │ registers on PowerPC                                         │
│              │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2023-6129                    │
│              ├────────────────┤          │        │                   ├───────────────┼──────────────────────────────────────────────────────────────┤
│              │ CVE-2023-6237  │          │        │                   │ 3.1.4-r4      │ openssl: Excessive time spent checking invalid RSA public    │
│              │                │          │        │                   │               │ keys                                                         │
│              │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2023-6237                    │
└──────────────┴────────────────┴──────────┴────────┴───────────────────┴───────────────┴──────────────────────────────────────────────────────────────┘
2024-01-22T19:13:09.503+0100    INFO    Table result includes only package filenames. Use '--format json' option to get the full path to the package file.

Python (python-pkg)

Total: 2 (UNKNOWN: 0, LOW: 0, MEDIUM: 1, HIGH: 1, CRITICAL: 0)

┌──────────────────────┬────────────────┬──────────┬────────┬───────────────────┬───────────────┬──────────────────────────────────────────────────────────────┐
│       Library        │ Vulnerability  │ Severity │ Status │ Installed Version │ Fixed Version │                            Title                             │
├──────────────────────┼────────────────┼──────────┼────────┼───────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ pip (METADATA)       │ CVE-2023-5752  │ MEDIUM   │ fixed  │ 23.0.1            │ 23.3          │ pip: Mercurial configuration injectable in repo revision     │
│                      │                │          │        │                   │               │ when installing via pip                                      │
│                      │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2023-5752                    │
├──────────────────────┼────────────────┤          │        ├───────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ setuptools (METADATA)│ CVE-2022-40897 │ HIGH     │        │ 58.1.0            │ 65.5.1        │ pypa-setuptools: Regular Expression Denial of Service        │
│                      │                │          │        │                   │               │ (ReDoS) in package_index.py                                  │
│                      │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2022-40897                   │
└──────────────────────┴────────────────┴──────────┴────────┴───────────────────┴───────────────┴──────────────────────────────────────────────────────────────┘
```

And I updated the Dockerfile to consider all vulnerabilities.

TP3 > Docker > Dockerfile

```dockerfile
# Use an official Python runtime as a parent image
# Alpine is used to reduce the size of the image
FROM python:3.13.0a3-alpine3.19

# Update packages and install security updates
RUN apk update && apk upgrade && \
    pip install --upgrade pip setuptools

# Create a new user to run the application
RUN adduser -D myuser
USER myuser

# Set the working directory in the container
WORKDIR /usr/src/app

# Copy the current directory contents into the container at /usr/src/app
COPY . .

# Install any needed packages specified in requirements.txt
RUN pip install --no-cache-dir -r requirements.txt

# Run app.py when the container launches
CMD ["python", "./app.py"]

```

I reran a trivy scan, and no vulns were found!

The new image is available under: `chargeutile/tp3-python-novuln`

```
(.venv) cflorval@Clements-MacBook-Pro Docker % trivy image tp3-python-novuln
2024-01-22T19:28:26.574+0100    INFO    Vulnerability scanning is enabled
2024-01-22T19:28:26.575+0100    INFO    Secret scanning is enabled
2024-01-22T19:28:26.575+0100    INFO    If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2024-01-22T19:28:26.575+0100    INFO    Please see also https://aquasecurity.github.io/trivy/v0.48/docs/scanner/secret/#recommendation for faster secret detection
2024-01-22T19:28:27.203+0100    INFO    Detected OS: alpine
2024-01-22T19:28:27.203+0100    WARN    This OS version is not on the EOL list: alpine 3.19
2024-01-22T19:28:27.203+0100    INFO    Detecting Alpine vulnerabilities...
2024-01-22T19:28:27.208+0100    INFO    Number of language-specific files: 1
2024-01-22T19:28:27.208+0100    INFO    Detecting python-pkg vulnerabilities...

tp3-python-novuln (alpine 3.19.0)

Total: 0 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 0, CRITICAL: 0)
```

- Put the code in a Github repo: The code will be on: https://github.com/ClementFrvl/Contenerization/tree/main/TP3