CS 5573/ EE 5523/IS 6973
Cloud Computing

Docker Containers

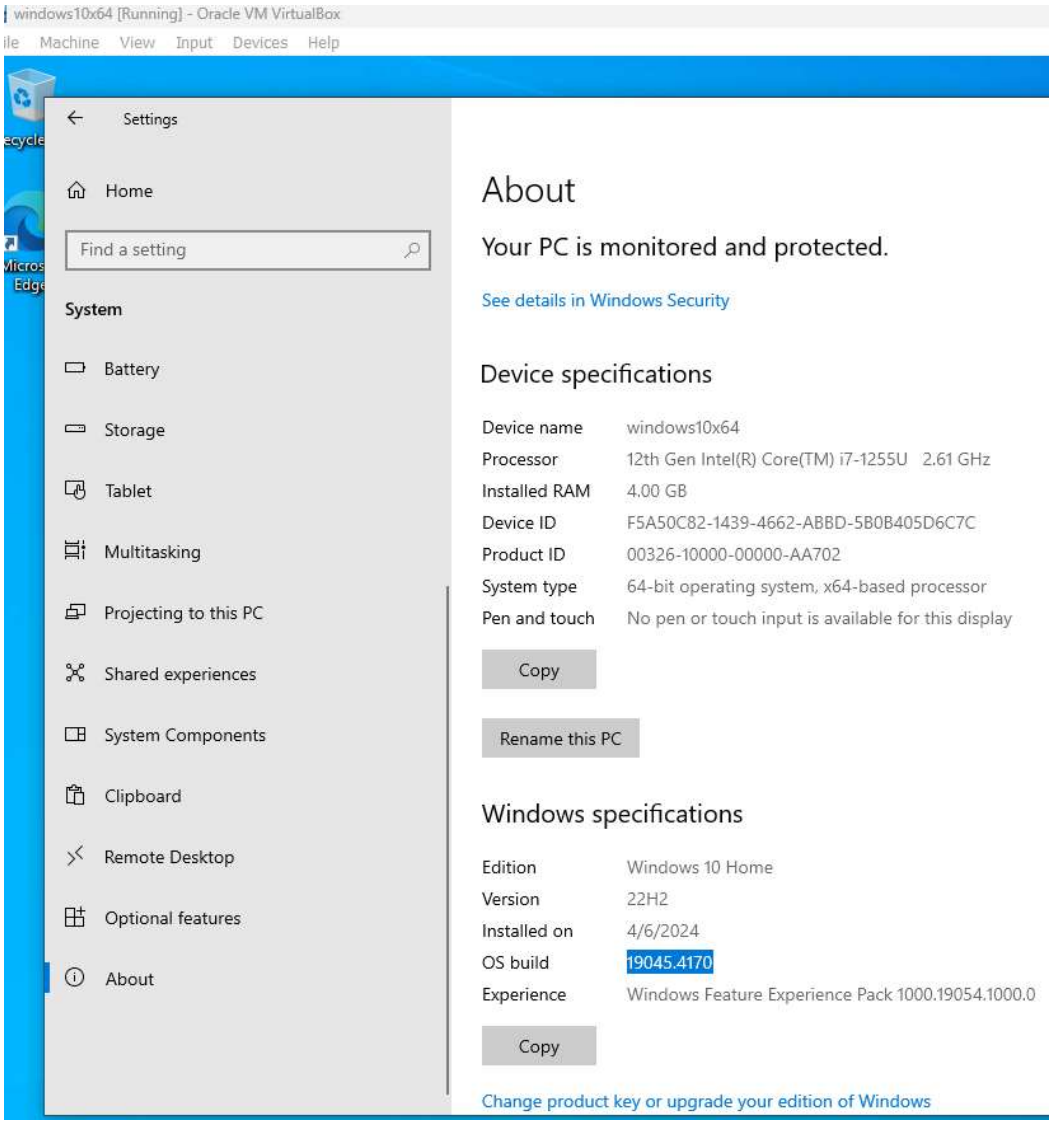
1. [10 pts] Docker Installation

(a) *Option 1*: If you choose to use Docker on a local Windows machine, install Docker Desktop using the following link. [Install Docker Desktop on Windows | Docker Documentation]

Note that Docker Desktop is supported only by the following versions of Windows.

Windows 11 64-bit: Home or Pro version 21H2 or higher, or Enterprise or Education version 21H2 or higher.

Windows 10 64-bit: Home or Pro 2004 (build 19041) or higher, or Enterprise or Education 1909 (build 18363) or higher.

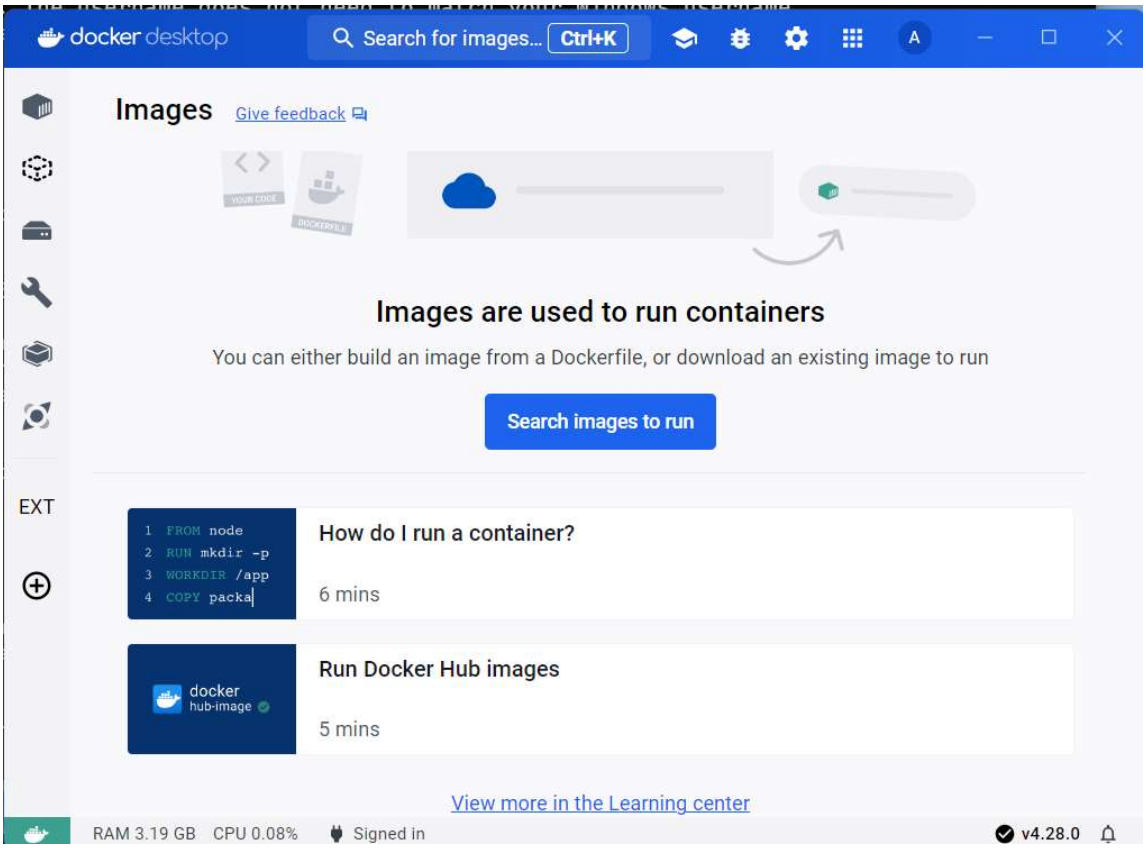


download & install docker desktop (4.28.0) for windows

==========

tried installing on windows VM, but facing issue while setting up wsl, so installed it on Windows physical box.

a) installed docker desktop 4.28.0 -reboot
b) update to latest WSL 2 from MS
c) sign-in into docker hub
d) starting docker engine
e) installed wsl for ubuntu and kali linux from MS appstore and configured using docker settings - > resources -> wsl integration - enabled both.



Note: after installing 'docker desktop for windows' on my laptop, I realised that it WSL was consuming 70% of RAM. So finally I uninstalled it and continued my work on my ubuntu 22.04 server.

(b) *Option 2*: If you choose to use Docker on a Linux machine, login to one of the Cloud VMs assigned to your group and run the following commands.

```
curl -fsSL https://get.docker.com -o get-docker.sh

sudo sh get-docker.sh

sudo usermod -aG docker $USER

sudo curl -L
"https://github.com/docker/compose/releases/download/1.29.2/docker-
```

```
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose
```

I installed docker packages by adding docker's apt repository (deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu jammy stable) in local server's apt sources.list.d. Intention was to get latest apt packages directly from docker website, downloaded packages list (list (download.docker.com_linux_ubuntu_dists_jammy_stable_binary-amd64_Packages)) in /var/lib/apt folder.

```
agupta@hl001:~$ sudo apt list --installed | grep 'docker\|compose'

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

docker-buildx-plugin/jammy,now 0.13.1-1~ubuntu.22.04~jammy amd64 [installed,automatic]
docker-ce-cli/jammy,now 5:26.0.1-1~ubuntu.22.04~jammy amd64 [installed]
docker-ce-rootless-extras/jammy,now 5:26.0.1-1~ubuntu.22.04~jammy amd64 [installed,automatic]
docker-ce/jammy,now 5:26.0.1-1~ubuntu.22.04~jammy amd64 [installed]
docker-compose-plugin/jammy,now 2.26.1-1~ubuntu.22.04~jammy amd64 [installed,automatic]
agupta@hl001:~$
```

2. [90 pts] Build and run a simple web application as a Docker container.

(a) Clone the following GitHub repository which contains a Dockerfile, a Python program (app.py) and HTML templates. On Windows, you need to use Git Bash. On Linux, use the terminal.

```
git clone https://github.com/lamapalden/mysimpleapp.git
```

```
agupta@hl001:~$ git clone https://github.com/lamapalden/mysimpleapp.git git_repo_dockerapp
Cloning into 'git_repo_dockerapp'...
remote: Enumerating objects: 22, done.
remote: Counting objects: 100% (22/22), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 22 (delta 6), reused 22 (delta 6), pack-reused 0
Receiving objects: 100% (22/22), done.
Resolving deltas: 100% (6/6), done.
agupta@hl001:~$
```

```
agupta@hl001:~/git_repo_dockerapp$ cat app.py
from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def index():
    return render_template('index.html')

@app.route("/c2f/<value>")
def convert_temperature(value):
    try:
       fahrenheit = float(value) * 9 / 5 + 32
       fahrenheit = round(fahrenheit, 3)  # Round to three decimal places
    except:
       return render_template('index.html')

    return render_template('convert1.html', var1=value, var2=fahrenheit)

if __name__ == '__main__':
        app.run()

agupta@hl001:~/git_repo_dockerapp$ python3 app.py
Traceback (most recent call last):
  File "/home/agupta/git_repo_dockerapp/app.py", line 1, in <module>
    from flask import Flask, render_template
ModuleNotFoundError: No module named 'flask'
agupta@hl001:~/git_repo_dockerapp$
```

```
agupta@hl001:~/git_repo_dockerapp$ pip install flask
Defaulting to user installation because normal site-packages is not writeable
Collecting flask
  Downloading flask-3.0.3-py3-none-any.whl (101 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 101.7/101.7 KB 392.8 kB/s eta 0:00:00
Collecting itsdangerous>=2.1.2
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Jinja2>=3.1.2
  Downloading Jinja2-3.1.3-py3-none-any.whl (133 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 133.2/133.2 KB 1.3 MB/s eta 0:00:00
Collecting click>=8.1.3
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 97.9/97.9 KB 1.1 MB/s eta 0:00:00
Collecting Werkzeug>=3.0.0
  Downloading werkzeug-3.0.2-py3-none-any.whl (226 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 226.8/226.8 KB 1.9 MB/s eta 0:00:00
Collecting blinker>=1.6.2
  Downloading blinker-1.7.0-py3-none-any.whl (13 kB)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/lib/python3/dist-packages (from Jinja2>=3.1.2->flask) (2.0.1)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-2.1.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, flask
  WARNING: The script flask is installed in '/home/agupta/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.2 blinker-1.7.0 click-8.1.7 flask-3.0.3 itsdangerous-2.1.2
agupta@hl001:~/git_repo_dockerapp$ python3 app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [12/Apr/2024 23:22:42] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:22:42] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [12/Apr/2024 23:26:50] "GET /c2f/200 HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:27:12] "GET /c2f/200 HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:27:20] "GET /c2f/20 HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:27:29] "GET /c2f/24 HTTP/1.1" 200 -
```

(b) Update the Python program (app.py) to add a new feature. The new feature should enable the web application to convert a given temperature from Fahrenheit to Celsius and display the converted temperature when you open a web browser on your local machine using the appropriate URL as described below.

If you are using Docker on local Windows machine, use the following URL:

localhost/f2c/<temperature>

For example,
localhost/f2c/32

If you are using Docker on the Cloud VM (Linux machine), use the following URL:

<VM's public IP address>/f2c/<temperature>
For example,
129.114.27.107/f2c/32

```
^Cagupta@hl001:~/git_repo_dockerapp$ cat app.py
from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def index():
    return render_template('index.html')

@app.route("/c2f/<value>")
def convert_temperature(value):
    try:
       fahrenheit = float(value) * 9 / 5 + 32
       fahrenheit = round(fahrenheit, 3)  # Round to three decimal places
    except:
       return render_template('index.html')

    return render_template('convert1.html', var1=value, var2=fahrenheit)



@app.route("/f2c/<value>")
def convert_temperaturef2c(value):
    try:
       celcius = (float(value) - 32) * 5 / 9
       celcius = round(celcius, 3)  # Round to three decimal places
    except:
       return render_template('index.html')

    return render_template('convert2.html', var1=value, var2=celcius)



if __name__ == '__main__':
        app.run()

agupta@hl001:~/git_repo_dockerapp$
```

```
agupta@hl001:~/git_repo_dockerapp$ python3 app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [12/Apr/2024 23:42:19] "GET /f2c/76 HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:42:27] "GET /c2f/24 HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2024 23:42:32] "GET / HTTP/1.1" 200 -
```

(c) Use appropriate command to build the Docker image of your web application. (Take a screenshot of the command and its output)

```
agupta@hl001:~/git_repo_dockerapp$ sudo docker build -t arti/pyflasksamp:v2 .
[+] Building 4.4s (9/9) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 195B
 => [internal] load metadata for docker.io/library/python:3.9-slim-bullseye
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/4] FROM docker.io/library/python:3.9-slim-bullseye@sha256:4699b511fd5a0bf93f8067749285f040d5937
 => [internal] load build context
 => => transferring context: 2.08kB
 => CACHED [2/4] WORKDIR /code
 => CACHED [3/4] RUN pip install flask
 => CACHED [4/4] COPY . .
 => exporting to image
 => => exporting layers
 => => writing image sha256:47dfda7342c03cae1ea7dc562f5e326afa996071cf513ae0c85e87116a624b24
 => => naming to docker.io/arti/pyflasksamp:v2
agupta@hl001:~/git_repo_dockerapp$ sudo docker images
REPOSITORY          TAG       IMAGE ID       CREATED          SIZE
arti/pyflasksamp    v2        47dfda7342c0   30 minutes ago   135MB
hello-world         latest    d2c94e258dcb   11 months ago    13.3kB
agupta@hl001:~/git_repo_dockerapp$
```

(d) Use appropriate command to run the web application as a Docker container. (Take a screenshot of the command and its output)

sudo docker run -p 80:8080 arti/pyflasksamp:v2

```
agupta@hl001:~/git_repo_dockerapp$ sudo docker build -t arti/pyflasksamp:v2 .
[+] Building 4.4s (9/9) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 195B
 => [internal] load metadata for docker.io/library/python:3.9-slim-bullseye
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/4] FROM docker.io/library/python:3.9-slim-bullseye@sha256:4699b511fd5a0bf93f8067749285
 => [internal] load build context
 => => transferring context: 2.08kB
 => CACHED [2/4] WORKDIR /code
 => CACHED [3/4] RUN pip install flask
 => CACHED [4/4] COPY . .
 => exporting to image
 => => exporting layers
 => => writing image sha256:47dfda7342c03cae1ea7dc562f5e326afa996071cf513ae0c85e87116a624b24
 => => naming to docker.io/arti/pyflasksamp:v2
agupta@hl001:~/git_repo_dockerapp$ sudo docker images
REPOSITORY          TAG       IMAGE ID       CREATED          SIZE
arti/pyflasksamp    v2        47dfda7342c0   30 minutes ago   135MB
hello-world         latest    d2c94e258dcb   11 months ago    13.3kB
agupta@hl001:~/git_repo_dockerapp$ sudo docker run -p 80:8080 arti/pyflasksamp:v2
 * Serving Flask app 'app.py'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a produc
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8080
 * Running on http://172.17.0.2:8080
Press CTRL+C to quit
172.17.0.1 - - [13/Apr/2024 05:25:38] "GET /f2c/76 HTTP/1.1" 200 -
172.17.0.1 - - [13/Apr/2024 05:25:48] "GET /c2f/76:80 HTTP/1.1" 200 -
172.17.0.1 - - [13/Apr/2024 05:25:54] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [13/Apr/2024 05:26:04] "GET /c2f/76 HTTP/1.1" 200 -
172.17.0.1 - - [13/Apr/2024 05:26:30] "GET /c2f/24 HTTP/1.1" 200 -
```

```
^Cagupta@hl001:~/git_repo_dockerapp$ sudo docker run -p 80:8080 -d arti/pyflasksamp:v2
4ed932b473394d6d306bc49da03e9c50e658b76ee0b01d677a606b7b50393024
agupta@hl001:~/git_repo_dockerapp$
```

(e) Open a web browser and use appropriate URL to convert any temperature from Fahrenheit to Celsius.

(Take a screenshot of the web page)

(f) Use appropriate command to find the name of the running container. (Take a screenshot of the command and its output)

```
agupta@hl001:~/git_repo_dockerapp$ sudo docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED              STATUS              PORTS                                     NAMES
4ed932b47339   arti/pyflasksamp:v2    "flask run --host=0.…"   About a minute ago   Up About a minute   0.0.0.0:80->8080/tcp, :::80->8080/tcp     zen_jang
agupta@hl001:~/git_repo_dockerapp$
```

(g) Use appropriate command to stop the running container. (Take a screenshot of the command and its output)

```
agupta@hl001:~/git_repo_dockerapp$ sudo docker stop 4ed932b47339
4ed932b47339
agupta@hl001:~/git_repo_dockerapp$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
agupta@hl001:~/git_repo_dockerapp$
```

Hints: Formula to convert Fahrenheit to Celsius is as follows:
Celsius = (Fahrenheit - 32) × 5/9

Submission Policy and Deliverables

Assignment must be submitted as a PDF file on Canvas. The file should include :

- Description of how the work was divided among your group members. It is critical to accurately describe the contribution made by each member. If the contribution is not significant, only partial credits will be provided.
- Representative Screenshots as specified in the assignment description.
- Include the contents of the Python code (app.py) in the report.