If you don't want to hardcode the numbers and instead want to provide them dynamically when running the Docker container, you can use environment variables. Here's how you can modify the calculator.py script and the Dockerfile to achieve this:

1. Update the calculator.py script:

python

# calculator.py

import os

import sys

def add(x, y):

return x + y

def subtract(x, y):

return x - y

def multiply(x, y):

return x \* y

def divide(x, y):

if y != 0:

return x / y

else:

return "Cannot divide by zero"

if \_\_name\_\_ == "\_\_main\_\_":

num1 = os.getenv("NUM1")

num2 = os.getenv("NUM2")

if not num1 or not num2:

print("Please provide values for NUM1 and NUM2 environment variables.")

sys.exit(1)

num1 = float(num1)

num2 = float(num2)

print("Sum:", add(num1, num2))

print("Difference:", subtract(num1, num2))

print("Product:", multiply(num1, num2))

print("Quotient:", divide(num1, num2))

1. Update the Dockerfile:

Dockerfile

# Dockerfile

FROM python:3.8-slim

WORKDIR /app

COPY . /app

EXPOSE 80

ENV NAME World

CMD ["python", "calculator.py"]

Now, when you run the Docker container, you can provide values for NUM1 and NUM2 as environment variables. For example:

bash

docker run -p 4000:80 -e NUM1=5 -e NUM2=3 calculator-app

Adjust the values of NUM1 and NUM2 as needed. This way, you can dynamically pass the numbers to the Python script without hardcoding them in the Dockerfile.