**🚀 My Docker App**

This project contains a Dockerized application designed for easy build, deployment, and management using Docker.

**🛠 Prerequisites**

* Docker installed and running
* Optional: Docker Compose (for multi-container setups)
* Clone the repository:
* git clone https://github.com/your-username/your-repo.git

cd your-repo

**🔨 Build the Docker Image**

docker build -t logparser-blockchain .

To build with a specific tag/version:

docker build -t logparser-blockchain:v1.0 .

**🚀 Run the Container**

docker run -d --name logparser -p 8080:80 logparser-blockchain

* -d: Run in detached mode
* --name: Name the container
* -p: Map host port to container port

**📦 Install Dependencies (If applicable)**

Dependencies are typically installed during build using your Dockerfile:

RUN pip install -r requirements.txt # For Python

You can also exec into the container:

docker exec -it logparser bash

**🔄 Restart the Container**

docker restart logparser

Or stop and start:

docker stop logparser

docker start logparser

**🔑 Key Commands**

|  |  |
| --- | --- |
| **Purpose** | **Command** |
| Build image | docker build -t logparser-blockchain . |
| Run container | docker run -d --name logparser logparser-blockchain |
| Stop container | docker stop logparser |
| Restart container | docker restart logparser |
| View logs | docker logs -f logparser |
| Exec into container | docker exec -it logparser bash |
| Remove container | docker rm -f logparser |
| Remove image | docker rmi logparser-blockchain |

**📂 Docker Compose (Optional)**

If using [docker-compose.yml](https://docker-compose.yml):

docker-compose up -d --build

To restart:

docker-compose restart

To shut down:

docker-compose down

[**docker-compose.yml**](https://docker-compose.yml)

version: '3.8'

services:

logparser:

build:

context: .

dockerfile: Dockerfile

container\_name: logparser

ports:

- "8080:80"

volumes:

- ./data:/app/data

environment:

- ENV=production

**🧩 Logparser Example with Blockchain Concept**

This example demonstrates a Python-based log parser that integrates blockchain concepts for secure and immutable log storage.

**Python Script**

import hashlib

import json

from datetime import datetime

class Block:

def \_\_init\_\_(self, index, timestamp, data, previous\_hash):

self.index = index

self.timestamp = timestamp

self.data = data

self.previous\_hash = previous\_hash

self.hash = self.calculate\_hash()

def calculate\_hash(self):

block\_string = f"{self.index}{self.timestamp}{self.data}{self.previous\_hash}"

return hashlib.sha256(block\_string.encode()).hexdigest()

class Blockchain:

def \_\_init\_\_(self):

self.chain = [self.create\_genesis\_block()]

def create\_genesis\_block(self):

return Block(0, str(datetime.now()), "Genesis Block", "0")

def add\_block(self, data):

previous\_block = self.chain[-1]

new\_block = Block(len(self.chain), str(datetime.now()), data, previous\_block.hash)

self.chain.append(new\_block)

def is\_chain\_valid(self):

for i in range(1, len(self.chain)):

current\_block = self.chain[i]

previous\_block = self.chain[i - 1]

if current\_block.hash != current\_block.calculate\_hash():

return False

if current\_block.previous\_hash != previous\_block.hash:

return False

return True

# Example usage

blockchain = Blockchain()

blockchain.add\_block("Log entry 1: User login")

blockchain.add\_block("Log entry 2: File uploaded")

for block in blockchain.chain:

print(f"Index: {block.index}, Timestamp: {block.timestamp}, Data: {block.data}, Hash: {block.hash}")

**Dockerfile**

FROM python:3.9-slim

WORKDIR /app

COPY . .

RUN pip install -r requirements.txt

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

**Steps to Run**

1. Place the Python script in a file named [logparser.py](https://logparser.py).
2. Create a Dockerfile with the content above.
3. Build the Docker image:

docker build -t logparser-blockchain .

1. Run the container:

docker run -d --name logparser logparser-blockchain

1. View the logs to see the blockchain in action:

docker logs -f logparser

**Project Structure**

docker-app-lab/

├── logparser-blockchain/

│ ├── Dockerfile

│ ├── logparser.py

│ ├── requirements.txt

│ ├── docker-compose.yml

│ └── README.md

└── other-projects/

[**requirements.txt**](https://requirements.txt)

# Core dependencies

flask>=2.0.0

requests==2.31.0

# Development and testing

black # Code formatting

pytest # Unit testing

# Optional

python-dotenv # Environment variable management

Let me know if you