Complete Docker CI/CD Project Documentation (With Commands)

This document explains every single step of the Docker CI/CD project in a very simple manner. Each step includes the exact commands used so that even a beginner can follow it easily.

1. Project Objective

To automatically build and push a Docker image to Docker Hub whenever code is pushed to GitHub.

2. Tools Used

Python (Flask), Docker, Git, GitHub, GitHub Actions, Docker Hub.

3. Create Project Folder

```
mkdir cicd-docker-project cd cicd-docker-project
```

4. Create Flask Application

```
nano app.py
```

Paste Flask code and save the file.

5. Create requirements.txt

```
nano requirements.txt flask
```

6. Create Dockerfile

```
nano Dockerfile FROM python:3.10-slim WORKDIR /app COPY requirements.txt . RUN pip install -r requirements.txt COPY app.py . CMD ["python", "app.py"]
```

7. Build and Run Docker Locally

```
docker build -t test-cicd-app . docker run -p 5000:5000 test-cicd-app
```

8. Initialize Git Repository

```
git init git branch -m main git add . git commit -m "Initial commit"
```

9. Push Code to GitHub

```
git remote add origin https://github.com/USERNAME/REPO.git git push -u origin main
```

10. Create GitHub Actions Workflow

```
mkdir -p .github/workflows nano .github/workflows/docker.yml
```

11. GitHub Actions Workflow Purpose

This workflow builds the Docker image and pushes it to Docker Hub automatically.

12. Add GitHub Secrets

Add DOCKER_USERNAME and DOCKER_PASSWORD in GitHub repository settings.

13. Trigger CI/CD Pipeline

```
git add . git commit -m "Add CI/CD pipeline" git push origin main
```

14. Verification

Check GitHub Actions tab for successful run and Docker Hub for pushed image.

15. Conclusion

This project demonstrates real-world CI/CD automation using Docker and GitHub Actions.