**Multi-Container Flask Application with PostgreSQL Using Docker Compose**

**Overview**

This project sets up a Flask application with a PostgreSQL database using Docker Compose. The application connects to PostgreSQL and provides a simple API to check the database connection.

**Prerequisites**

Before running this project, ensure you have the following installed:

- Docker

- Docker Compose

**Project Structure**

Flask-Docker/

│── app.py                   Flask application

│── requirements.txt         Python dependencies

│── Dockerfile               Dockerfile for Flask app

│── docker-compose.yml       Docker Compose configuration

└── README.md               Project documentation

**Setup and Running the Application**

**Step 1: Clone the Repository**

git clone https://github.com/KPkm25/Flask-Docker

cd Flask-Docker

**Step 2: Build and Start the Containers**

docker-compose up -d --build

This will:

- Build the Flask application image

- Start the PostgreSQL database container

**Step 3: Verify the Running Containers**

docker ps

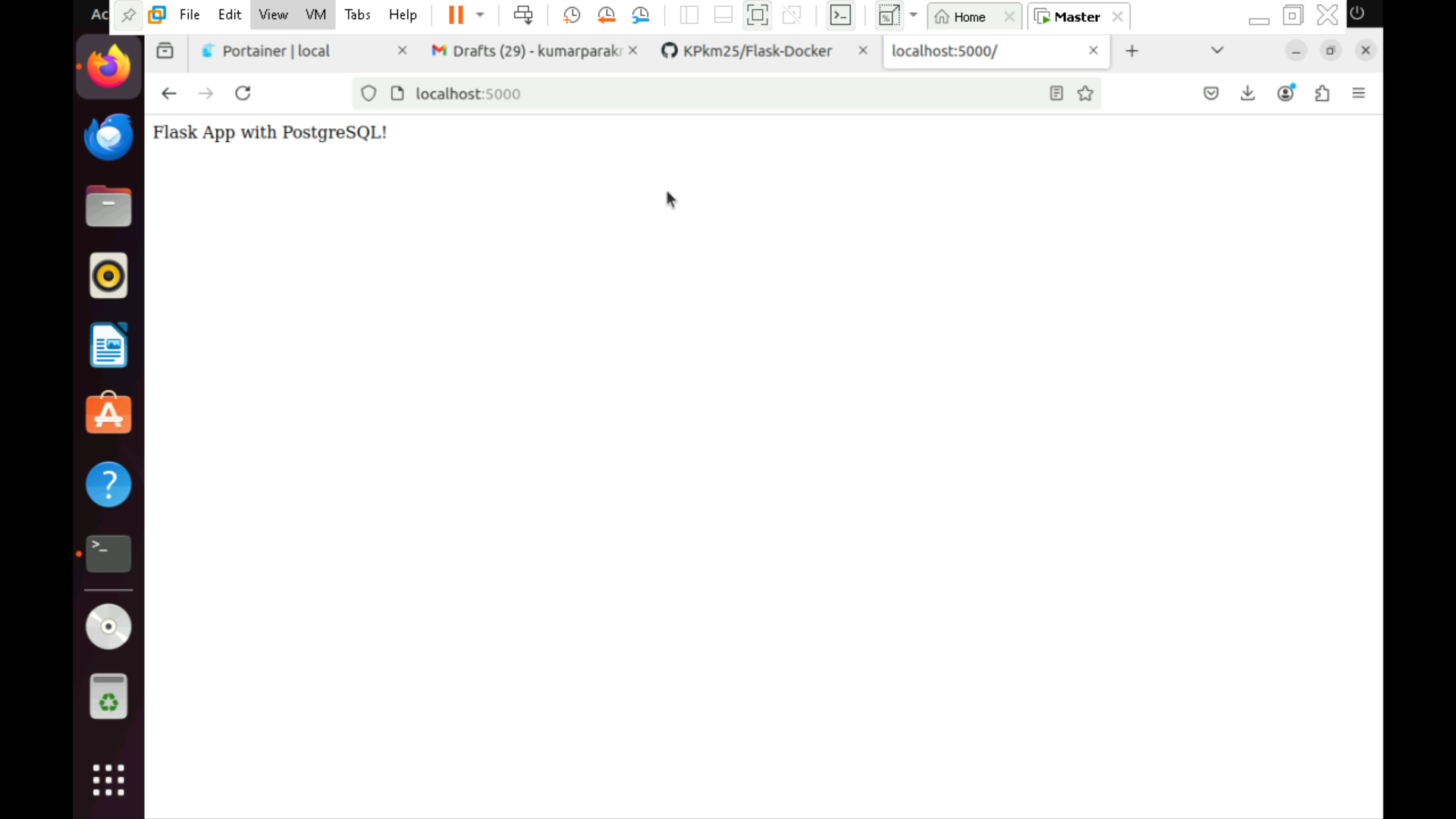
You should see `web` (Flask app) and `db` (PostgreSQL) services running.

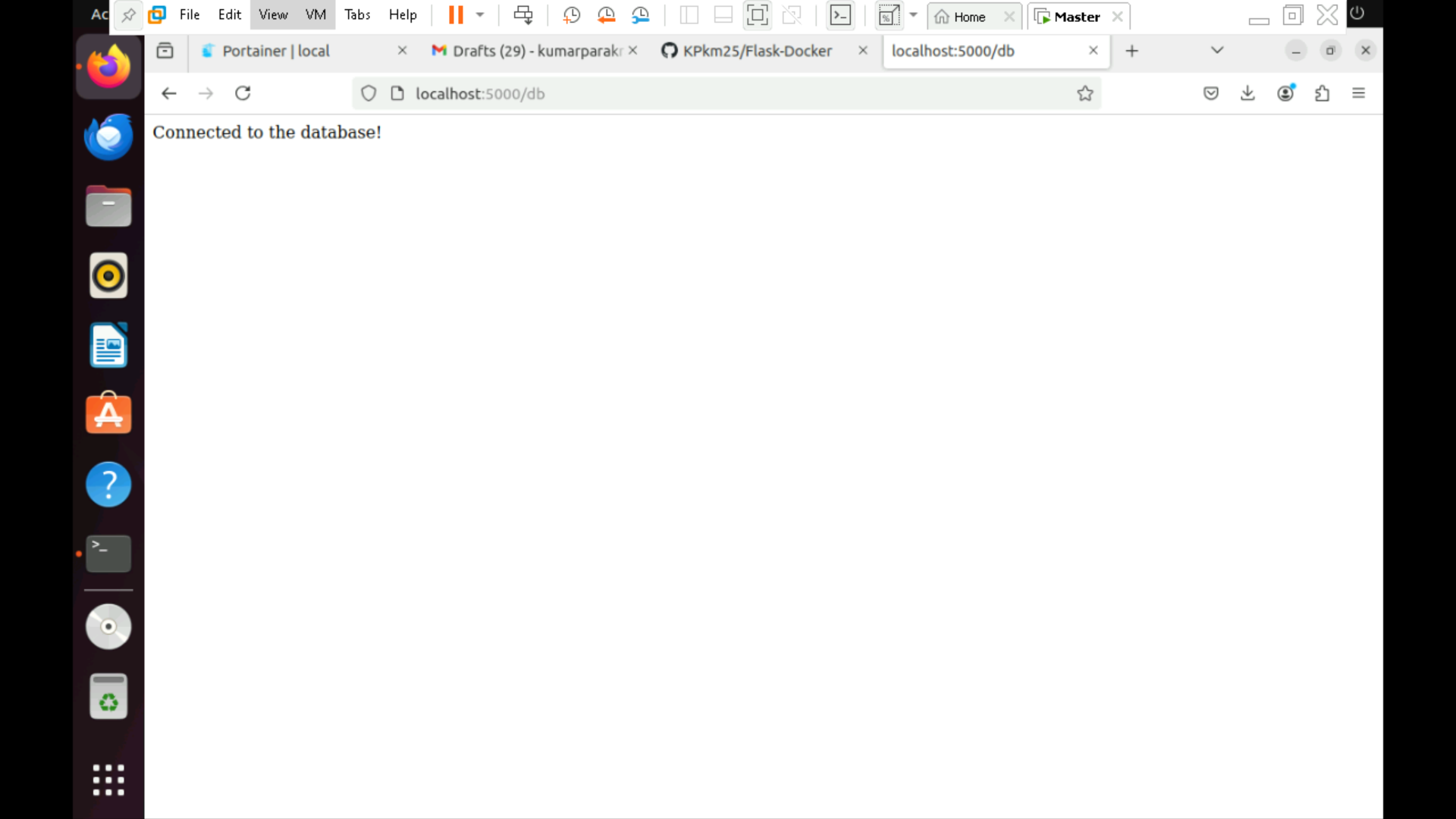
**Step 4: Test the Application**

Open your browser or use `curl` to access the endpoints:

- `http://localhost:5000/` → Should return `"Flask App with PostgreSQL!"`

- `http://localhost:5000/db` → Should confirm database connection





**Stopping and Cleaning Up**

To stop the containers:

docker-compose down

To remove unused images and volumes:

docker system prune -a

docker volume prune

**Troubleshooting**

**ERROR: Port 5432 Already in Use**

If you encounter an error like `port is already allocated`, another PostgreSQL instance may be running. Either:

- Stop the local PostgreSQL service:

  sudo systemctl stop postgresql

- Change the PostgreSQL port in `docker-compose.yml` (e.g., `5433:5432` instead of `5432:5432`).

**Restarting the Containers**

If the application does not work as expected, try rebuilding the containers:

docker-compose up -d --build