Title: DOCKER error: Pull access denied for coffeeteareadb, repository does not exist or may require 'docker login'

Post Body:

I have a `Spring MVC` application that I want to add to `Docker`. I created the `image`, configured `Docker`, but the application in `Docker` does not want to start.
In the application I use `Spring Boot` and `PostgresSQL` database.

Dockerfile:

```
FROM openjdk:11 ADD build/libs/Coffeetearea-0.0.1-SNAPSHOT.jar Coffeetearea-0.0.1-SNAPSHOT.jar #EXPOSE 8080:8080 ENTRYPOINT ["
```

docker-compose.yml:

```
version: '3.1' services:   app:     container_name: coffeetearea     image: coffeeteareaimage     build: ./     ports:       -
```

application.propeerties:

```
#Databse spring.jpa.database-platform=org.hibernate.dialect.PostgreSQLDialect spring.datasource.driver-class-name=org.postgres
```

MY STEPS in TERMINAL:

```
C:\Users\vartanyan\IdeaProjects\Coffeetearea>docker-compose up Creating network "coffeetearea_default" with the default driver
```

Accepted Answer: None

Highest Rated Answer:

The error means that Docker cannot find an image named `coffeeteareadb` locally or on https://hub.docker.com/ . If your image is in private repository (meaning that someone in your party have already created it) you have to login Docker into that repository first. Although for private repository you image name should look like an URL: `registry.example.com/image-name:tag`.

If you want `coffeeteareadb` to be a regular PostgreSQL database, you probably want to change the `image` here:

```
  coffeeteareadb:     image: postgres:13  # https://hub.docker.com/_/postgres
```

If you are new to Docker, an `image` is like an executable or binary file, while a `container` is something like a running process of that executable. An `image` consists of several incremental layers of data: an application, dependencies, some basic config, etc. When you run an image you create a `container`. A container is an instance of an image. It differs from the image in the way that apart from just application, it has some information on how to run it (which ports to map, which volumes to mount, etc). There can be many containers using the same image. So when you asked to select an image you basically need to tell what application you want to use. Docker will look for it locally and on the hub, download it, and create a container from it. If you want to create your own image, you need a Dockerfile (see this reference https://docs.docker.com/engine/reference/builder/).