**Lesson 07 Lesson-End Project**

**Containerizing Legacy Docker Application**



****

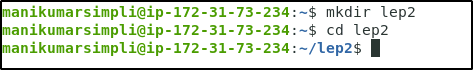
**Steps to be followed:**

1. Create a new directory
2. Create a Docker file
3. Install the Python requirements defined in the *req.txt* file to modify the parent image.
4. Create a *.yml* file in the project directory *lep2* that defines the services that make the app
5. Create the Django project
6. List all the contents of the Django Project
7. Change the ownership of new files as the current owner is root
8. Set-up a Database connection

**Step 1: Create a new directory**

***mkdir lep2***

***cd lep2***



**Step 2: Create a Dockerfile**

***nano Dockerfile***



2.1 Add the following code to the Dockerfile so that it can start Python3 parent image:

***FROM python:3***

***ENV PYTHONUNBUFFERED 1***

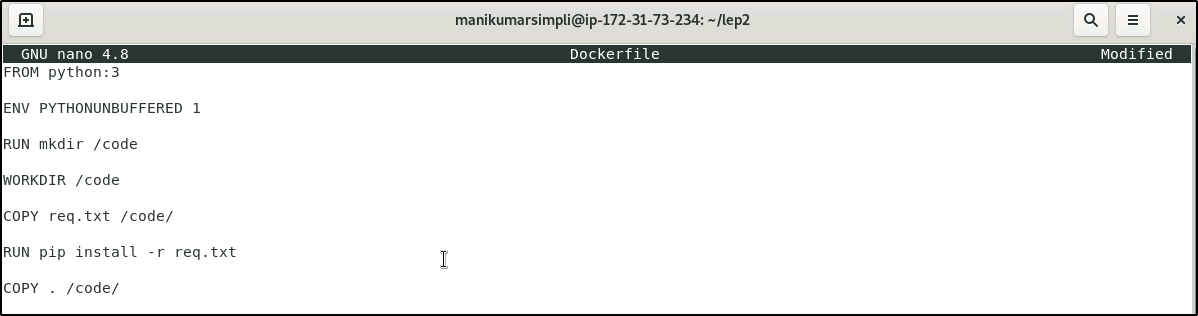
***RUN mkdir /code***

***WORKDIR /code***

***COPY req.txt /code/***

***RUN pip install -r req.txt***

***COPY . /code/***



****

**Step 3: Install the Python requirements defined in the *req.txt* file to modify the parent image**

3.1 Create a *req.txt* in your project directory *lep2*

***nano req.txt***



3.2 Add the following software in *req.txt* file:

*Django>=2.0,<3.0*

*psycopg2>=2.7, <3.0*



****

**Step 4: Create a *.yml* file in the project directory *lep2* that defines the services that make the app**

4.1 Create a docker-compose*.yml* file

***nano docker-compose.yml***



4.2 Add the following code to define the services:

***version: '3.3'***

***services:***

***db:***

***image: postgres***

***web:***

***build: .***

***command: python manage.py runserver 0.0.0.0:8000***

***volumes:***

***- .:/code***

***ports:***

***- "8000:8000"***

***depends\_on:***

***- db***

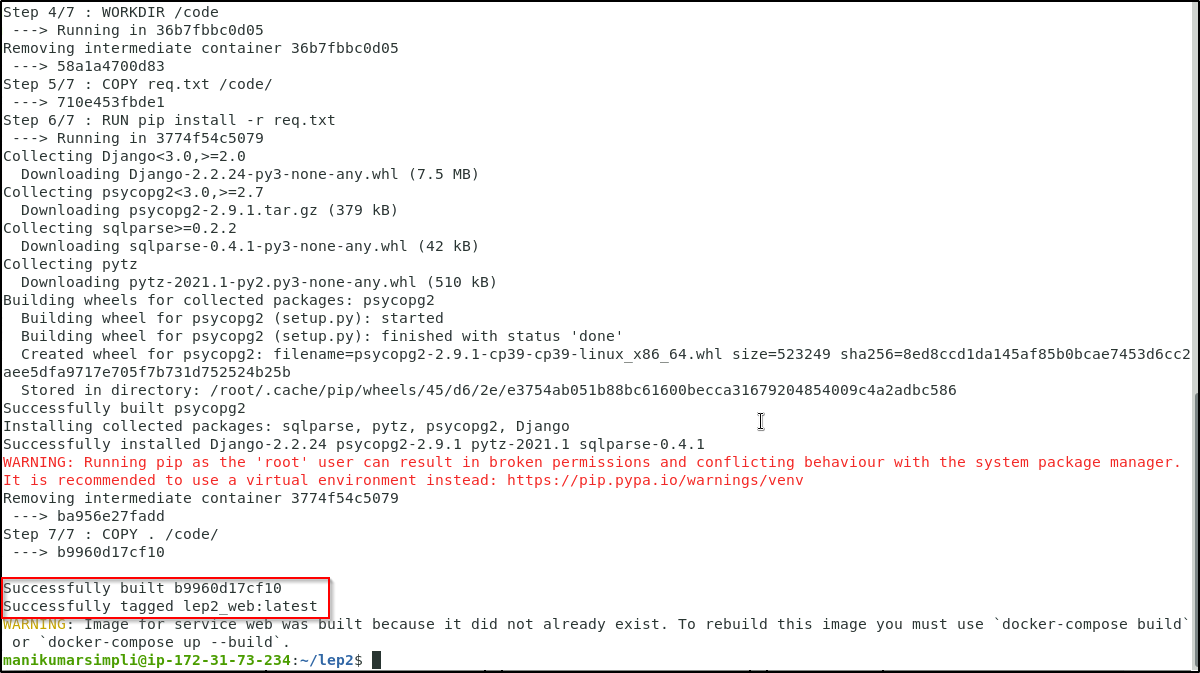


****

**Step 5: Use the following command to create the Django project:**

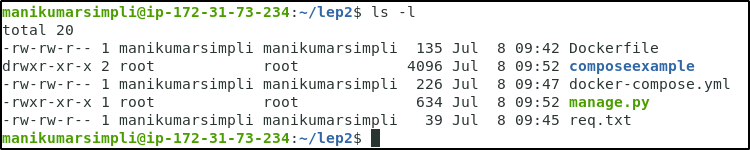
***sudo docker-compose run web django-admin startproject composeexample .***

**



**Step 6: List the contents of the Django project by using the following command:**

***ls -l***



**Step 7: Change the ownership of new files as the current owner is root:**

***sudo chown -R $USER:$USER .***



**Step 8: Set up a database connection for Django**

8.1 Edit *cmps\_eg/settings.py* file in the project directory:

*edit composeexample.py*



8.2 Press E to edit the *composeexample.py* file

8.3 Replace the *DATABASES=……* commands with the following:

***DATABASES = {***

***'default': {***

***'ENGINE': 'django.db.backends.postgresql',***

***'NAME': 'postgres',***

***'USER': 'postgres',***

***'HOST': 'db',***

***'PORT': 5432,***

***}***

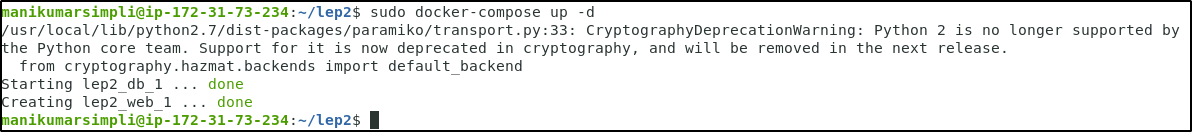
***}***



****

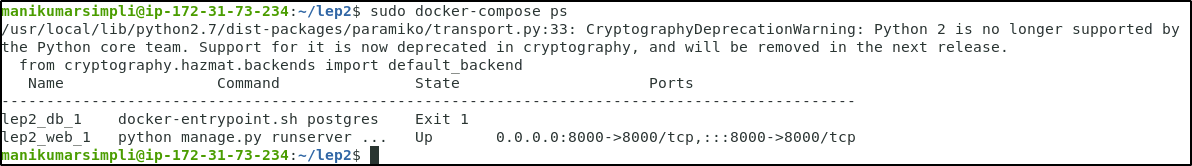
8.4 Run the following command to start the application:

***sudo docker-compose up -d***



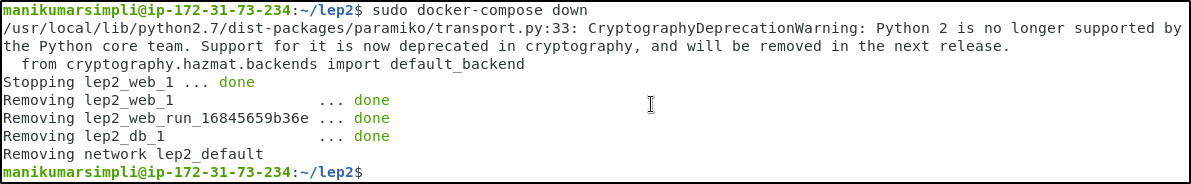
8.5 List the containers by using this command:

***sudo docker-compose ps***



8.6 Use the following command to bring the application down:

***sudo docker-compose down***



8.7 Remove the Django app by using the following command:

***rm -rf django***

