Detailed Steps to Create and Run a Python File in Docker

Step 1: Ensure Docker is Running in the Background

- Open Docker Desktop or your Docker application.
- Ensure it is running and ready for commands.
- Confirm by typing the following in the terminal:
- docker --version

Step 2: Create a Folder and Add Your Python File

1. **Create a Folder:**
   - Create a folder named Docker in your system.
2. mkdir Docker
3. cd Docker
4. **Add a Python File:**
   - Inside the Docker folder, create a Python file named first.py.
   - Add the following simple Python code as an example:
   - print("Hello, Docker!")

Step 3: Create a Dockerfile Definition

1. **Create a Text File:**
   - In the same Docker folder, create a file named Dockerfile.txt.
2. **Add the Following Text to the File:**
3. # Use the official Python image with version 3.12
4. FROM python:3.12-slim
5.
6. # Set the working directory inside the container
7. WORKDIR /app
8.
9. # Copy the Python file to the working directory
10. COPY first.py .
11.
12. # Command to execute the Python script

13. CMD ["python", "first.py"]

Step 4: Convert the Text File to a Dockerfile

1. **Open a Terminal in the Folder:**
   - Right-click in the Docker folder and select **Open Terminal** or **Open Command Prompt**.
2. **Rename Dockerfile.txt to Dockerfile:**
   - Type the following command to rename the file:
   - mv Dockerfile.txt Dockerfile

On Windows, you can manually rename the file using File Explorer.

Step 5: Build the Docker Image

1. **Run the Build Command:**
   - In the terminal, type the following command to build the Docker image:
   - docker build -t python312 .
   - **Explanation:**
     - docker build: Builds an image from the Dockerfile.
     - -t python312: Assigns the name python312 to the image.
     - .: Refers to the current directory as the build context.
2. **Verify the Image:**
   - Check if the image was created:
   - docker images

Step 6: Run the Docker Container

1. **Run the Image to Execute the Python Script:**
   - Use the following command to create and run a container:
   - docker run python312
   - The output of first.py should appear in the terminal:
   - Hello, Docker!

Step 7: Log In to Docker Hub

1. **Login Command:**
   - Use the following command to log in to your Docker Hub account:
   - docker login
   - Enter your username and password when prompted.

Step 8: Tag the Docker Image

1. **Create a Tag for Your Image:**
   - Use the following command to tag your image for Docker Hub:
   - docker tag python312 <your-dockerhub-username>/python312
   - Replace <your-dockerhub-username> with your actual Docker Hub username.

Step 9: Push the Image to Docker Hub

1. **Push the Tagged Image:**
   - Upload your image to Docker Hub:
   - docker push your-dockerhub-username/python312
   - This makes the image available online.

Step 10: Verify the New Container in Docker Hub

1. **Check Docker Hub:**
   - Log in to Docker Hub.
   - Navigate to your repositories and find the new image (python312).

You have successfully built, run, tagged, and pushed a Docker image containing your Python script!