Title: Hello World in Docker: Your First Container

This recipe uses the What – How – Why format to help you run your first Docker container using the official hello-world image — a foundational step for all DevOps learners using Docker for the first time.

🔸 **What (Introduction & Theory)**

This recipe walks you through running your first Docker container using the hello-world image — a basic, official container used to test whether Docker is correctly installed and working.

It covers three essential tasks:

1. **Docker Version Check** – Confirms Docker Engine is available and running.
2. **Running Hello World Container** – Launches the image to test functionality.
3. **Cleaning Up** – Prunes the stopped container to keep the system clean.

These steps help ensure Docker is functioning correctly on your system, before you move on to building and running complex containers for tools like Jenkins, GitLab, or Nginx.

🔸 **How (Step-by-Step Process)**

**Part 1: Confirm Docker Installation**

**Step 1: Open Ubuntu Terminal in WSL2**

* Press Windows + S, type **WSL** .
* Right-click → select **Run as Administrator**.
* This opens your Ubuntu shell inside WSL2.

Step 2: Check Docker Version

**docker --version**

Expected output:

**Docker version 24.x.x, build xxxxxxxx**

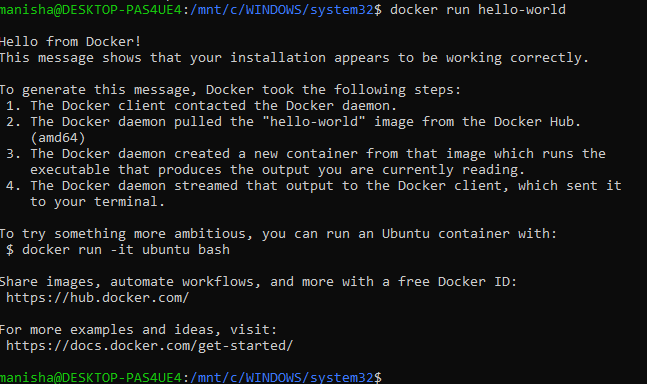
If this fails, install Docker using the official Linux instructions:  
<https://docs.docker.com/engine/install/ubuntu/>



Part 2: Run Hello World Container  
Step 3: Launch Hello World

**docker run hello-world**

This command:  
• Downloads the hello-world image (if not present),  
• Creates a container from it,  
• Executes the code, then stops.



Step 4: Confirm Output  
You should see this in the terminal:

**Hello from Docker!**

**This message shows that your installation appears to be working correctly.**

...

This proves Docker is working end-to-end.

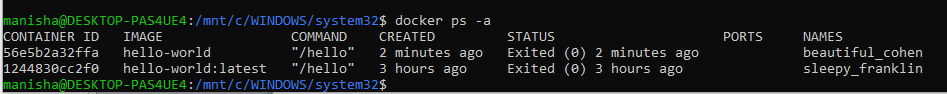
Part 3: Check and Clean Containers  
Step 5: View Containers

**docker ps -a**

Expected output includes:

**IMAGE hello-world**

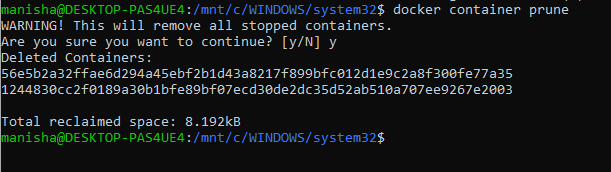
**STATUS Exited (0)**

****

Step 6: Remove Stopped Containers (Optional)

**docker container prune**

💡 This command removes all stopped containers. Confirm with y when prompted.



Output:  
• Your Docker installation is confirmed working.  
• You have successfully pulled and run your first container.  
• You're ready to start building and running real-world DevOps containers.

**🔸 Why (Practical Thinking & Reasoning)**

Why use the hello-world image?  
✔️ It's a safe, minimal image designed for testing.  
✔️ It proves Docker Engine, networking, and container lifecycle are all functional.

Why does the container stop immediately?  
✔️ Because hello-world simply prints a message and exits.  
✔️ It helps you understand how some containers are designed to do one job and quit.

Why check with docker ps -a?  
✔️ docker ps only shows **running** containers.  
✔️ docker ps -a shows both running and **stopped** ones like this one.

Why prune the container?  
✔️ To clean up unused containers.  
✔️ It helps prevent clutter and keeps your system efficient.

Why run Docker from Ubuntu in WSL2?  
✔️ Because it gives you full Linux container support without needing a VM.  
✔️ Most DevOps tooling and scripting are done on Linux-based systems.

What next after this setup?  
✔️ Try docker run -it ubuntu bash to launch an interactive Ubuntu container.  
✔️ Learn Dockerfile basics to build custom images.  
✔️ Explore Docker Compose for multi-container applications.