sudo apt update

sudo apt install ros-humble-xacro ros-humble-joint-state-publisher-gui ros-humble-robot-state-publisher

Here's the purpose of each tool in the command you ran:

---

### \*\*1. `ros-humble-xacro`\*\*

\*\*Purpose:\*\*

- \*\*Xacro\*\* (XML Macros) is a tool to simplify writing URDF files.

- It allows you to use macros, variables, and parameters in your robot description, making the URDF more modular and easier to modify.

- For example, instead of writing repetitive XML code for multiple parts (like wheels), you can define a macro and reuse it.

\*\*Why it’s needed:\*\*

- Simplifies robot description files.

- Essential for creating complex robot URDFs, like defining multiple similar joints or links.

---

### \*\*2. `ros-humble-joint-state-publisher-gui`\*\*

\*\*Purpose:\*\*

- Publishes the states of the robot's joints (e.g., position, velocity).

- The \*\*GUI version\*\* allows you to manually adjust joint positions via a graphical interface, which is useful for testing and visualization.

\*\*Why it’s needed:\*\*

- Helps visualize how different parts of the robot (like wheels or arms) move in \*\*Rviz2\*\*.

- Used during testing to simulate joint movements without actual hardware.

---

### \*\*3. `ros-humble-robot-state-publisher`\*\*

\*\*Purpose:\*\*

- Takes the URDF file of your robot and calculates the positions of all links based on joint states (e.g., from `joint\_state\_publisher`).

- Publishes the robot's TF (transformations) to ROS topics, allowing tools like \*\*Rviz2\*\* and \*\*Nav2\*\* to understand the robot's structure and movement.

\*\*Why it’s needed:\*\*

- Crucial for visualizing the robot in \*\*Rviz2\*\* and ensuring that the robot's transformations (e.g., base link to wheel or sensor positions) are correct.

---

### \*\*Summary of Use:\*\*

These tools together allow you to:

1. Write efficient and reusable URDFs using \*\*Xacro\*\*.

2. Publish joint states with \*\*joint\_state\_publisher\_gui\*\* for testing.

3. Visualize and compute the robot’s TF tree using \*\*robot\_state\_publisher\*\*.

They are the foundation for working with robot descriptions in ROS2 and will be used in both simulation and real robot deployment. Let me know if you'd like more details or examples!

Now created ws

cd ~/autonomous\_vehicle\_ws

colcon build

create pkg for robot -description

"\\wsl.localhost\Ubuntu-22.04\home\user\Software\_implementaion\ws\_1\full\_car.urdf"