

ROS (Robot Operating System) provee librerías y herramientas para ayudar a los desarrolladores de software a crear aplicaciones para robots. ROS provee abstracción de hardware, controladores de dispositivos, librerías, herramientas de visualización, comunicación por mensajes, administración de paquetes y más. ROS está bajo la licencia open source, BSD.

Comando	Descripción
<code>printenv grep -i ROS</code>	Imprime las variables de entorno necesarias para operar ROS 2: ROS_VERSION=2 ROS_PYTHON=3 ROS_DISTRO=foxy
<code>export ROS_DOMAIN_ID=<your_domain_id></code> <code>echo "export ROS_DOMAIN_ID=<your_domain_id>" >> ~/.bashrc</code>	El ID de dominio se utiliza para segmentar la red a fin de evitar interferencias entre diferentes grupos de computadoras que ejecutan ROS 2 en la misma red de área local. Las máquinas con diferentes ID de dominio no hablarán ni interferirán entre sí. Este número entero ID debe estar entre 0-232 para que el demonio ROS 2 se inicie correctamente
<code>source /opt/ros/<distro>/setup.bash</code>	Actualiza variables de entorno para que se encuentren todos los paquetes de ROS2 correctamente. Se debe escribir como una línea más en el archivo ".bashrc".
<code>git clone https://github.com/ros/ros_tutorials.git -b <distro>-devel</code>	El comando "git clone" clona un repositorio para una distribución de ROS en particular.
<code>mkdir -p ~/<dev_ws>/src</code>	Crea un espacio de trabajo y la carpeta "src" donde se instalarán todos los paquetes.
<code>rosdep install -i --from-path src --rosdistro <distro> -y</code>	Verificar las dependencias cada vez que clona un repositorio.
<code>colcon build ó colcon build --packages-select <paquete></code>	Compila todos los paquetes del espacio de trabajo ó solamente el paquete seleccionado
<code>. install/local_setup.bash</code>	Actualiza las variables de ambiente para ubicar las rutas el nuevo paquete compilado y sus nodos.
<code>ros2 run <paquete> nodo arg1 arg2 ... agrN</code>	Ejecuta un nodo de un paquete específico, el nodo puede tener argumentos o no.
<code>ros2 ros2 <comando> -h</code>	Imprime todos los comandos de ros2 y se puede solicitar la ayuda por cada comando.
<code>ros2 pkg create --build-type ament_<python/cmake> nom_pkg -- dependencies <paquete1> <paquete2> ... <paqueteN></code>	En el directorio ~/<dev_ws>/src, se crea un nuevo paquete para compilarlo en python o cpp, y se anexan dependencias con otros paquetes.
<code>ros2 node -h ros2 topic -h ros2 service -h ros2 action -h ros2 interface -h</code>	Comandos comunes de ros2.
<code>ros2 launch <paquete> <archivo.launch.py></code>	El comando "launch" toma de la carpeta "launch", que se encuentra en el paquete correspondiente, el archivo.launch.py para ejecutar un conjunto de nodos con sus parámetros respectivos.