

# ROS 2 Humble + Virtualenv Setup su Mint

Questa guida riassume tutti i passaggi necessari per configurare un virtualenv funzionante con ROS 2 Humble su Linux Mint, pronto per buildare pacchetti custom come turtlesim senza errori.

## 1 Creare il virtualenv con accesso ai pacchetti di sistema

```
cd ~/Documenti/AMR
rm -rf ros2pythonenv
python3 -m venv ros2pythonenv --system-site-packages
source ros2pythonenv/bin/activate
```

## 2 Installare le librerie Python aggiuntive

```
pip install --upgrade pip setuptools
pip install empy==3.3.4 lark-parser==0.11.3 catkin_pkg
```

## 3 Installare Qt sul sistema

```
sudo apt update
sudo apt install qtbase5-dev qtbase5-dev-tools libqt5core5a libqt5gui5
libqt5widgets5
```

## 4 Preparare il workspace

```
cd ~/Documenti/AMR/ros2pythonenv/src/lab1workspace
rosdep update
rosdep install -i --from-path src --rosdistro humble --skip-
keys=libqt5core5t64 -y
```

## 5 Pulire eventuali build precedenti

```
rm -rf build install log
```

## 6 Build dei pacchetti

```
# Build solo i pacchetti custom
colcon build --packages-select my-test-package
```

```
# Se vuoi modificare turtlesim e sovrascriverlo (opzionale)
# colcon build --allow-overriding turtlesim
```

## 7 Sorgere l'ambiente del workspace

```
source install/setup.bash
```

### Note Importanti

- Il virtualenv deve avere accesso a tutte le librerie Python richieste da ROS 2 Humble.
- Usare `--system-site-packages` evita la necessità di installare manualmente `ament_package` e altri pacchetti di sistema.
- Evitare di ricompilare pacchetti di sistema come `turtlesim` a meno che non li modifichi.
- Versioni specifiche dei pacchetti Python compatibili con Humble:
  - `empy==3.3.4`
  - `lark-parser==0.11.3`
  - `catkin_pkg`