



me ajude a fazer o exercício

Vamos fazer o exercício passo a passo, em modo “tutorial”, para você conseguir repetir no seu PC.^[1]

Visão geral do que você precisa fazer

Você precisa:

- Criar o pacote `mars_rover_tasks` do tipo `ament_python` com as dependências pedidas.^[1]
- Criar o nó Python `ObstacleDetectorNode` no arquivo `subscriber_obstacle_detector.py`.^[1]
- Configurar o `setup.py` com o entry point `subscriber_obstacle_detector_executable`.^[1]
- Compilar só esse pacote e rodar o nó com `ros2 run`.^[1]

Passo 1: Criar o pacote

No terminal:

```
cd ~/ros2_ws
source install/setup.bash    # se já existir install
cd src
ros2 pkg create --build-type ament_python mars_rover_tasks \
--dependencies rclpy std_msgs nav_msgs sensor_msgs geometry_msgs
```

Isso cria a estrutura básica do pacote com as dependências certas.^[1]

Passo 2: Criar o nó Python

Dentro da pasta do pacote:

```
cd ~/ros2_ws/src/mars_rover_tasks/mars_rover_tasks
```

Crie o arquivo `subscriber_obstacle_detector.py` com este conteúdo (código já fornecido na solução):^[1]

```
#!/usr/bin/env python
import rclpy
from rclpy.node import Node

class ObstacleDetectorNode(Node):
    def __init__(self, node_name="obstacle_detector_node"):
```

```

        self._node_name = node_name
        super().__init__(self._node_name)
        self.get_logger().info(self._node_name + " Ready...")

def main(args=None):
    rclpy.init(args=args)
    node = ObstacleDetectorNode()
    rclpy.shutdown()

if __name__ == '__main__':
    main()

```

Dê permissão de execução (opcional, mas bom hábito):

```
chmod +x subscriber_obstacle_detector.py
```

Passo 3: Configurar o setup.py

Abra `~/ros2_ws/src/mars_rover_tasks/setup.py` e ajuste para algo equivalente ao da solução:^[1]

```

from setuptools import find_packages, setup
import os
from glob import glob

package_name = 'mars_rover_tasks'

setup(
    name=package_name,
    version='0.0.0',
    packages=find_packages(exclude=['test']),
    data_files=[
        ('share/ament_index/resource_index/packages',
            ['resource/' + package_name]),
        ('share/' + package_name, ['package.xml']),
        (os.path.join('share', package_name), glob('launch/*.launch.py')),
    ],
    install_requires=['setuptools'],
    zip_safe=True,
    maintainer='user',
    maintainer_email='user@todo.todo',
    description='TODO: Package description',
    license='TODO: License declaration',
    tests_require=['pytest'],
    entry_points={
        'console_scripts': [
            'subscriber_obstacle_detector_executable = mars_rover_tasks.subscriber_obstacle_detector:main'
        ],
    },
)

```

O importante aqui é a linha do `console_scripts`, que liga o nome executável `subscriber_obstacle_detector_executable` à função `main` do seu script.^[1]

Passo 4: Compilar e rodar

No terminal:

```
cd ~/ros2_ws
source install/setup.bash    # se existir
colcon build --packages-select mars_rover_tasks
source install/setup.bash
ros2 run mars_rover_tasks subscriber_obstacle_detector_executable
```

Se tudo estiver certo, você verá algo como:^[1]

```
[INFO] [...] [obstacle_detector_node] Ready...
```

Se quiser, manda o erro/print do terminal que aparece para você, e dá para ir corrigindo linha por linha.

1.3.4-Define-a-Topic.pdf