2^{nd} task

TASK:

Use Turtlebot3 with SLAM approach to create and save a map.

Name :atheer al-mutairi

Steps:

I installed Ubuntu 18.04 and ROS Melodic

Then installed Rviz and Gazebo

After that I used the following commands:

```
> sudo apt update
> sudo apt upgrade
> wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_melodic.sh
> chmod 755 ./install_ros_melodic.sh
> bash ./install_ros_melodic.sh
```

to install ROS on remote PC

then I had to install the dependences by the following command:

```
> sudo apt-get install ros-melodic-joy ros-melodic-teleop-twist-joy \ ros-melodic-teleop-twist-keyboard ros-melodic-laser-proc \ ros-melodic-rgbd-launch ros-melodic-depthimage-to-laserscan \ ros-melodic-rosserial-arduino ros-melodic-rosserial-python \ ros-melodic-rosserial-server ros-melodic-rosserial-client \ ros-melodic-rosserial-msgs ros-melodic-amcl ros-melodic-map-server \ ros-melodic-move-base ros-melodic-urdf ros-melodic-xacro \ ros-melodic-compressed-image-transport ros-melodic-rqt* \ ros-melodic-gmapping ros-melodic-navigation ros-melodic-interactive-markers
```

now I can start installing the robot packages, by using the commands listed:

```
> sudo apt-get install ros-melodic-dynamixel-sdk
> sudo apt-get install ros-melodic-turtlebot3-msgs
> sudo apt-get install ros-melodic-turtlebot3
```

Install Simulation Package:

```
> cd ~/catkin_ws/src/
> git clone -b melodic-devel https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git
> cd ~/catkin_ws && catkin_make
> Then write cd or open new terminal and write source ~/catkin_ws/devel/setup.bash or use echo "source ~/catkin_ws/devel/setup.bash" >> ~/.bashrc
```

Now I can launch a simulation world, so I decided to use a TurtleBot3 World with a robot called "waffle"

```
> export TURTLEBOT3_MODEL=waffle
> roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

I used a command to control the robot movement :

```
> roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

where W: Forward, A: Left, S:Stop, D: Right, X:Backward.

Now using **SLAM simulation** and create and save a map:

```
>   export TURTLEBOT3_MODEL=waffle
>   roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

new terminal

```
>   export TURTLEBOT3_MODEL=waffle
>   roslaunch turtlebot3_slam turtlebot3_slam.launch slam_methods:=gmapping
```

new terminal

```
>   export TURTLEBOT3_MODEL=waffle
>   roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

finely save the map:

```
>   rosrun map_server map_saver -f ~/map
```

Results: