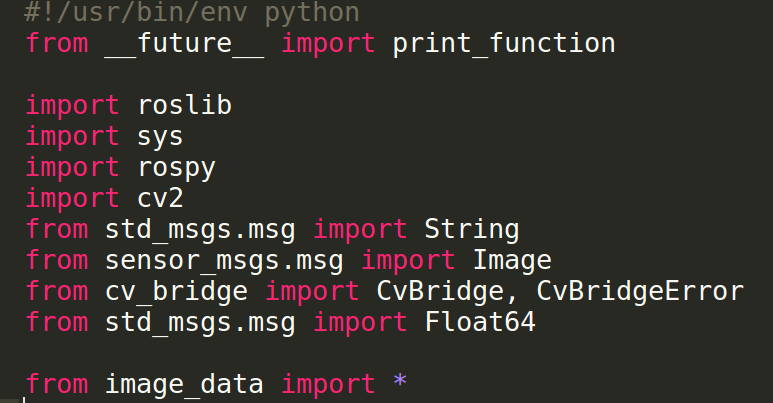
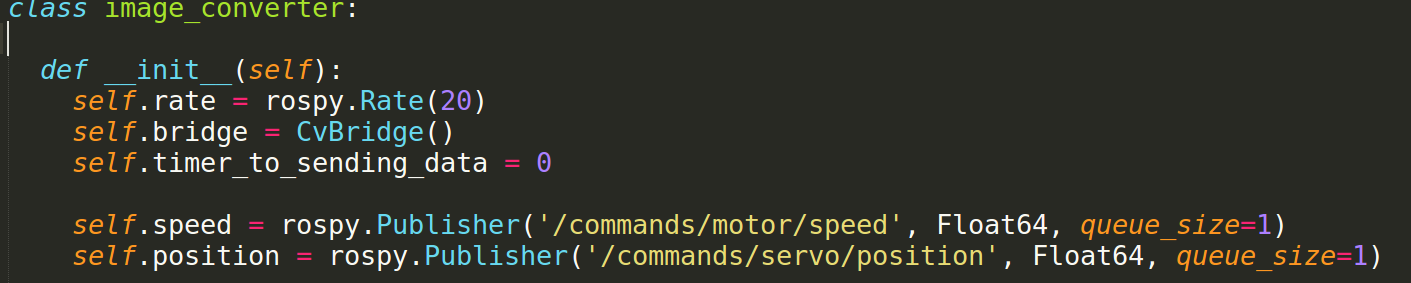
**Lane Detection ROS node Explanation**

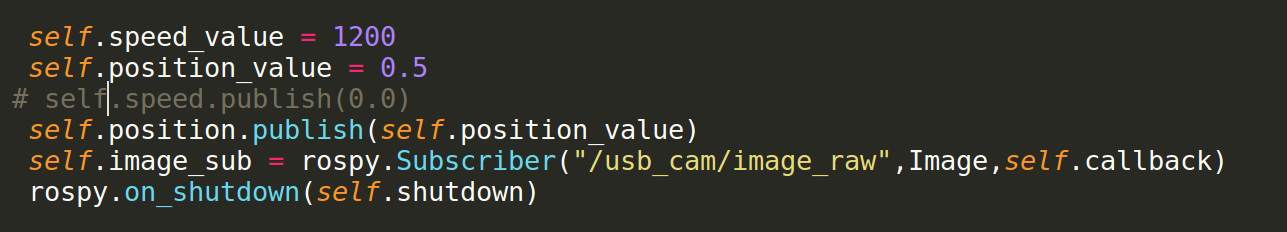


Import the required dependencies and line detection python code. rospy is a python library for ROS. It enables us to quickly interface with ROS topics, services and parameters.

In ROS, it passes the information in its own sensor\_msgs/Image message format. But for our line detecction python code we use the image with OpenCV. To convert the ROS image format to OpenCV format we use the cv\_bridge library provided by ROS. It acts as an interface between ROS and OpenCV. Here image\_data implies the line detection source file name.



In the wecar, we uses the VESC motor driver for controlling the movement of the WeCar. When we launch the VESC driver they are many topics are published. In this we use two topics namely /commands/motor/speed & /commands/servo/position for controlling the speed and steering respectively.



Here we have to subscribe the camera topics published by the webcamera. Image here represents the data type of that ROS images. The self.callback is a callback function defined as follows.



Callback function will be execute as soon as the images are received from the webcamera. In the callback function we convert the ROS image format into OpenCV format using the cv\_bridge library. We use the imgmsg\_to\_cv2 function from the v\_bridge is used for the conversion. After this we can send this image to lane detection source file which we explained earlier.

As mentioned before we use the 2 topics for controlling the speed and angle of the wecar. Depends on the available map and width we can tune the values of speed and angle. In the Lane Detection source code we use the current angle of the center line. If the angle is positive we move right if it is negative we move left.

To execute the lane detection node launch the following:

roslaunch lane\_detection lane\_detection.launch