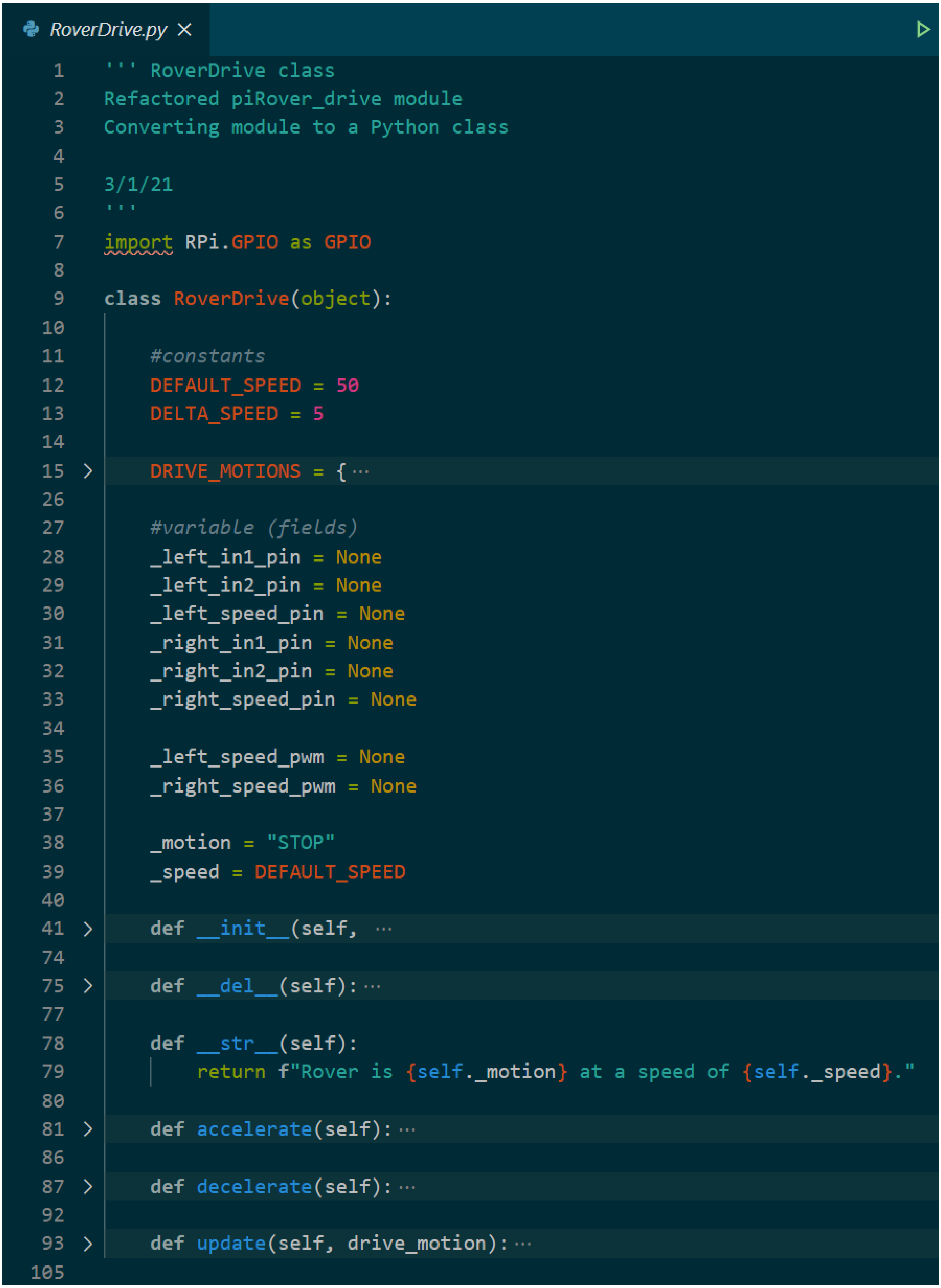
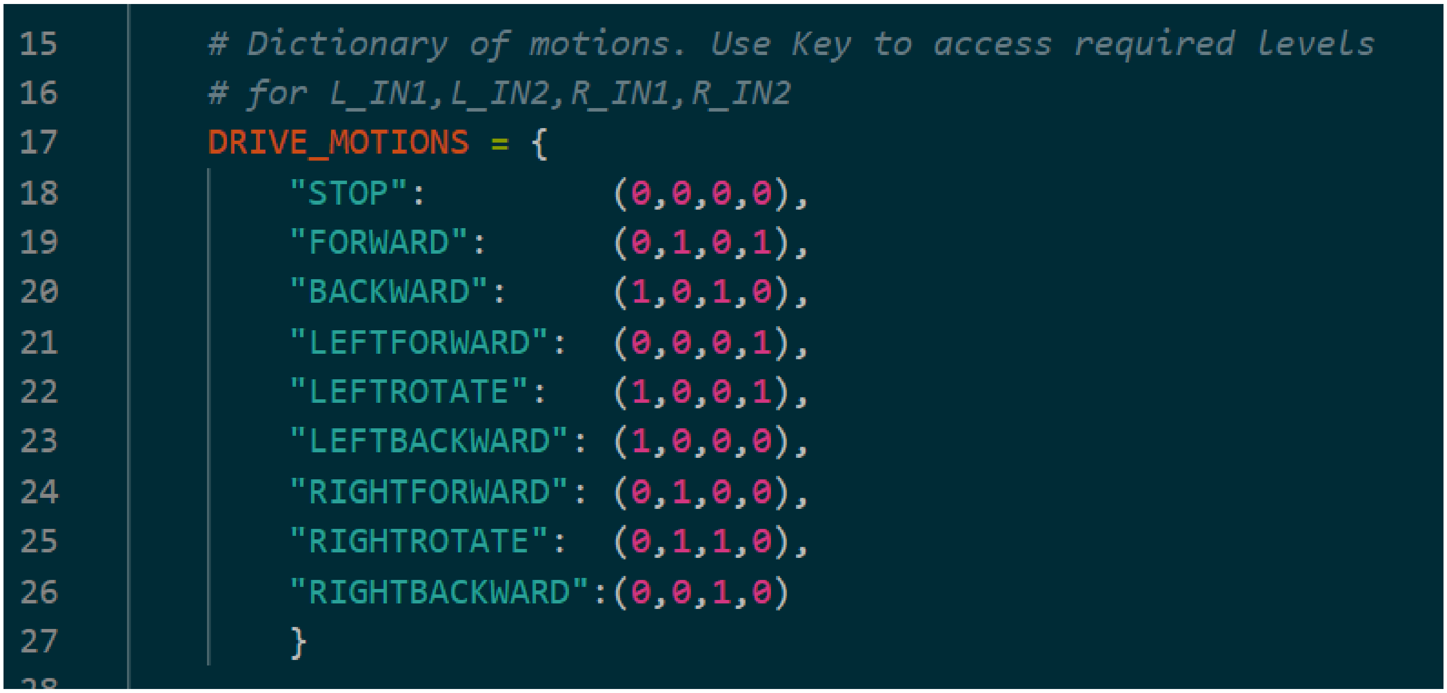
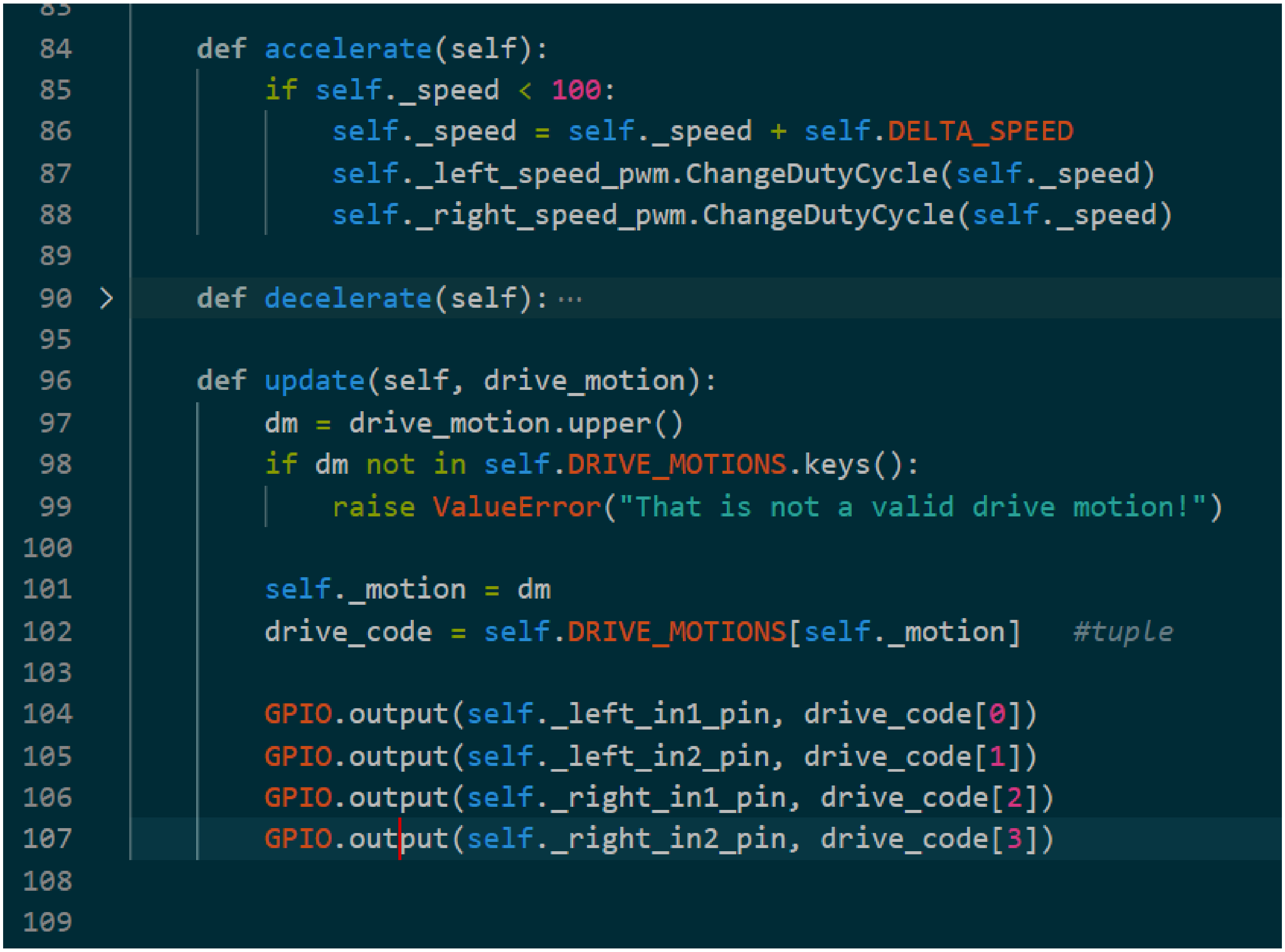


```python
''' RoverDrive class
Refactored piRover_drive module
Converting module to a Python class

3/1/21
'''
import RPi.GPIO as GPIO

class RoverDrive(object):

    #constants
    DEFAULT_SPEED = 50
    DELTA_SPEED = 5

    DRIVE_MOTIONS = {···

    #variable (fields)
    _left_in1_pin = None
    _left_in2_pin = None
    _left_speed_pin = None
    _right_in1_pin = None
    _right_in2_pin = None
    _right_speed_pin = None

    _left_speed_pwm = None
    _right_speed_pwm = None

    _motion = "STOP"
    _speed = DEFAULT_SPEED

    def __init__(self, ···

    def __del__(self):···

    def __str__(self):
        return f"Rover is {self._motion} at a speed of {self._speed}."

    def accelerate(self):···

    def decelerate(self):···

    def update(self, drive_motion):···
```



```
    # Dictionary of motions. Use Key to access required levels
    # for L_IN1,L_IN2,R_IN1,R_IN2
    DRIVE_MOTIONS = {
        "STOP":         (0,0,0,0),
        "FORWARD":      (0,1,0,1),
        "BACKWARD":     (1,0,1,0),
        "LEFTFORWARD":  (0,0,0,1),
        "LEFTROTATE":   (1,0,0,1),
        "LEFTBACKWARD": (1,0,0,0),
        "RIGHTFORWARD": (0,1,0,0),
        "RIGHTROTATE":  (0,1,1,0),
        "RIGHTBACKWARD":(0,0,1,0)
        }
```



```
    def accelerate(self):
        if self._speed < 100:
            self._speed = self._speed + self.DELTA_SPEED
            self._left_speed_pwm.ChangeDutyCycle(self._speed)
            self._right_speed_pwm.ChangeDutyCycle(self._speed)

    def decelerate(self):···

    def update(self, drive_motion):
        dm = drive_motion.upper()
        if dm not in self.DRIVE_MOTIONS.keys():
            raise ValueError("That is not a valid drive motion!")

        self._motion = dm
        drive_code = self.DRIVE_MOTIONS[self._motion]   #tuple

        GPIO.output(self._left_in1_pin, drive_code[0])
        GPIO.output(self._left_in2_pin, drive_code[1])
        GPIO.output(self._right_in1_pin, drive_code[2])
        GPIO.output(self._right_in2_pin, drive_code[3])
```