Python code for Blinking Led using Raspberry pi

```
#!/usr/bin/python

import RPi.GPIO
import time

class Led(object):
    '''
    Represents a physical LED. It uses the RPi GPIO naming scheme and NOT
Broadcomm's
    '''
    def __init__(self, pin_number):
        '''
        Set up the hardware connection
        Params: pin_number of type int - Follow RPi GPIO naming scheme
        '''
        self.pin_number = pin_number
        self.__setup_gpio__()

    def __setup_gpio__(self):
        RPi.GPIO.setmode(RPi.GPIO.BOARD)
        RPi.GPIO.setup(self.pin_number, RPi.GPIO.OUT)

    def clean_up(self):
        '''
        Reset the GPIO header to its initial state.
        '''
        RPi.GPIO.cleanup(self.pin_number)

    def on_light(self):
        '''
        Switch on the LED
        '''
        RPi.GPIO.output(self.pin_number, True)

    def off_light(self):
        '''
        Switch off the LED
        '''
        RPi.GPIO.output(self.pin_number, False)

    def blink(self, drift_time=0.2):
        self.on_light()
        time.sleep(float(drift_time))
        self.off_light()
        time.sleep(float(drift_time))

    def blinkn(self, number_times):
        '''
        Blink for any number of times
        Param: number_times of type int
        '''
        for i in range(0, int(number_times)):
            self.blink()
```

```
    def blink_non_stop(self):
        while True:
            self.blink()
```

Python Code For Traffic LED using Raspberry pi

```
import RPi.GPIO as GPIO

import time

import signal

import sys

# Setup

GPIO.setmode(GPIO.BCM)

GPIO.setup(9, GPIO.OUT)

GPIO.setup(10, GPIO.OUT)

GPIO.setup(11, GPIO.OUT)

# Turn off all lights when user ends demo

def allLightsOff(signal, frame):

   GPIO.output(9, False)

   GPIO.output(10, False)

   GPIO.output(11, False)

   GPIO.cleanup()

   sys.exit(0)

signal.signal(signal.SIGINT, allLightsOff)

while True:

   # Red

   GPIO.output(9, True)

   time.sleep(3)

   # Red and amber

   GPIO.output(10, True)

   time.sleep(1)

   # Green

   GPIO.output(9, False)
```

```
GPIO.output(10, False)

GPIO.output(11, True)

time.sleep(5)

# Amber

GPIO.output(11, False)

GPIO.output(10, True)

time.sleep(2)

# Amber off (red comes on at top of loop)

GPIO.output(10, False)
```