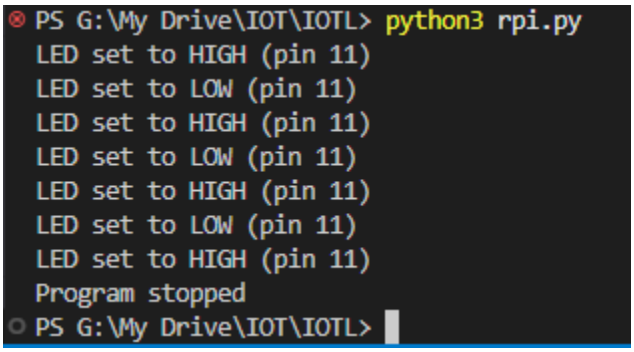
Experiment 4

Program:

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
led_pin = 11
GPIO.setup(led_pin, GPIO.OUT)

try:
    while True:
        GPIO.output(led_pin, GPIO.HIGH)
        print(f"LED set to HIGH (pin {led_pin})")
        time.sleep(0.5)
        GPIO.output(led_pin, GPIO.LOW)
        print(f"LED set to LOW (pin {led_pin})")
        time.sleep(0.5)

except KeyboardInterrupt:
    GPIO.cleanup()
    print("Program stopped")
```

Output:



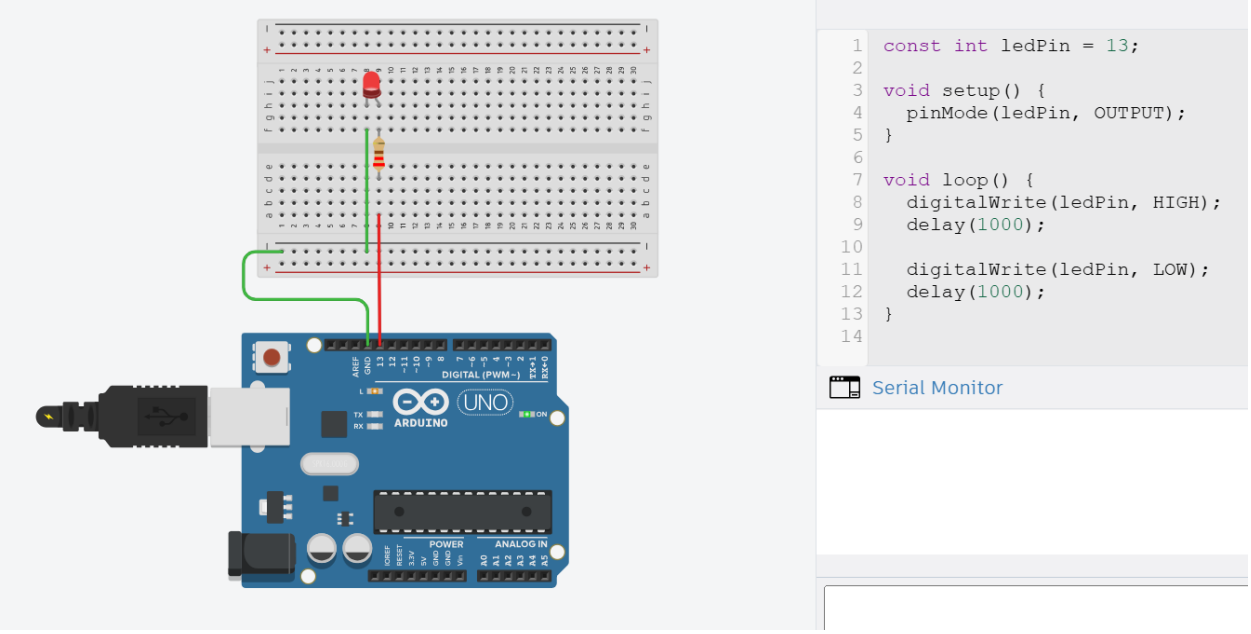
Experiment 5

Write a program using Arduino to control LED (One or more ON/OFF). Or Blinking

Program:

```
const int ledPin = 13;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);

  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

Output:



Experiment 6

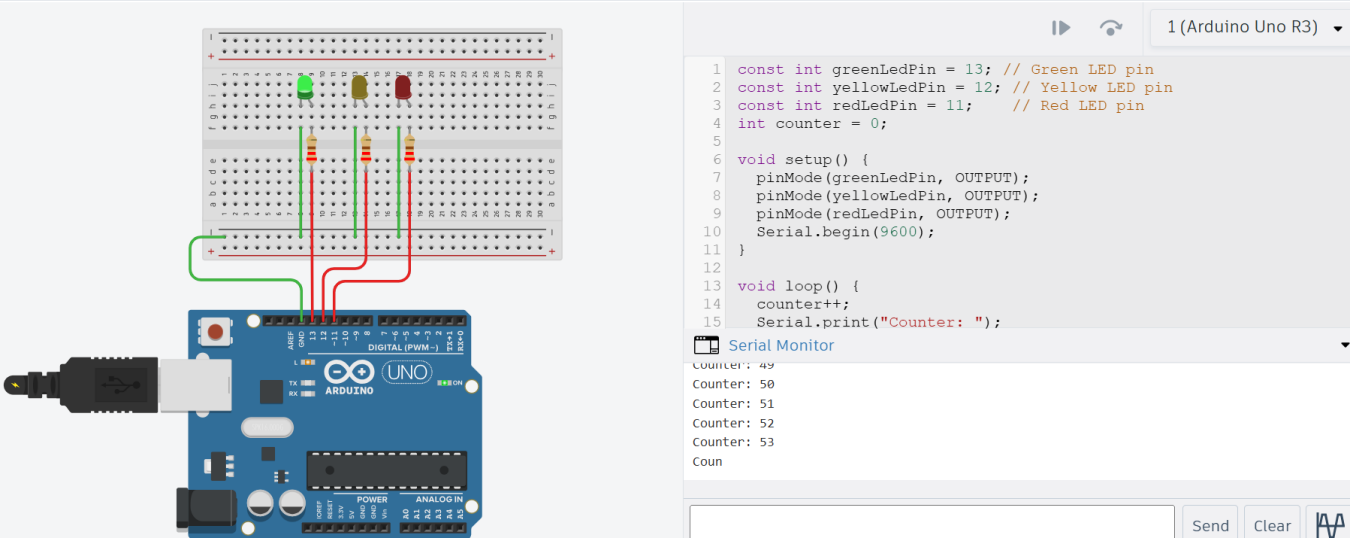
Create a program that illuminates the green LED if the counter is less than 100, illuminates the yellow LED if the counter is between 101 and 200 and illuminates the red LED if the counter is greater than 200

Program:

```
const int greenLedPin = 13; // Green LED pin
const int yellowLedPin = 12; // Yellow LED pin
const int redLedPin = 11;    // Red LED pin
int counter = 0;

void setup() {
  pinMode(greenLedPin, OUTPUT);
  pinMode(yellowLedPin, OUTPUT);
  pinMode(redLedPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  counter++;
  Serial.print("Counter: ");
  Serial.println(counter);

  if (counter < 100) {
    digitalWrite(greenLedPin, HIGH);
    digitalWrite(yellowLedPin, LOW);
    digitalWrite(redLedPin, LOW);
  } else if (counter >= 100 && counter <= 200) {
    digitalWrite(greenLedPin, LOW);
    digitalWrite(yellowLedPin, HIGH);
    digitalWrite(redLedPin, LOW);
  } else if (counter > 200) {
    digitalWrite(greenLedPin, LOW);
    digitalWrite(yellowLedPin, LOW);
    digitalWrite(redLedPin, HIGH);
  }
```

```
  if (counter >= 300) {
    counter = 0;
  }

  delay(100);
}
```

Output:



Experiment 7

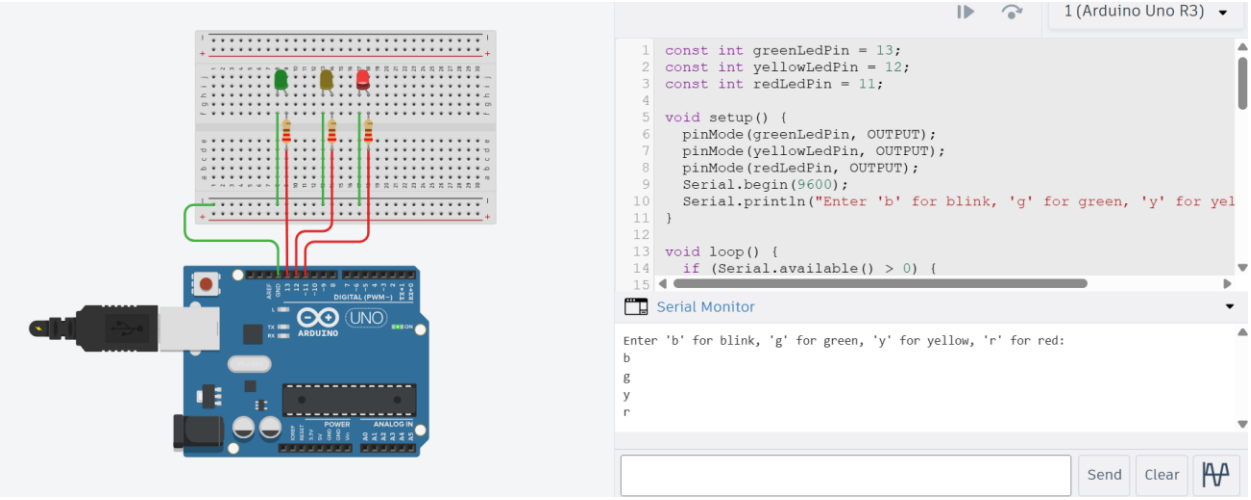
Create a program so that when the user enters ‘b’ the green light blinks, ‘g’ the green light is illuminated ‘y’ the yellow light is illuminated and ‘r’ the red light is illuminated

Program:

```
const int greenLedPin = 13;
const int yellowLedPin = 12;
const int redLedPin = 11;

void setup() {
  pinMode(greenLedPin, OUTPUT);
  pinMode(yellowLedPin, OUTPUT);
  pinMode(redLedPin, OUTPUT);
  Serial.begin(9600);
  Serial.println("Enter 'b' for blink, 'g' for green, 'y' for yellow, 'r' for red:");
}

void loop() {
  if (Serial.available() > 0) {
    char input = Serial.read();
    Serial.println(input);

    switch (input) {
      case 'b':
        blinkGreen();
        break;
      case 'g':
        digitalWrite(greenLedPin, HIGH);
        digitalWrite(yellowLedPin, LOW);
        digitalWrite(redLedPin, LOW);
        break;
      case 'y':
        digitalWrite(greenLedPin, LOW);
        digitalWrite(yellowLedPin, HIGH);
        digitalWrite(redLedPin, LOW);
```

```
        break;
      case 'r':
        digitalWrite(greenLedPin, LOW);
        digitalWrite(yellowLedPin, LOW);
        digitalWrite(redLedPin, HIGH);
        break;
      default:
        Serial.println("Invalid input.");
    }
  }
}

void blinkGreen() {
  for (int i = 0; i < 5; i++) { // Blink 5 times
    digitalWrite(greenLedPin, HIGH);
    delay(250);
    digitalWrite(greenLedPin, LOW);
    delay(250);
  }
}
```

Output:



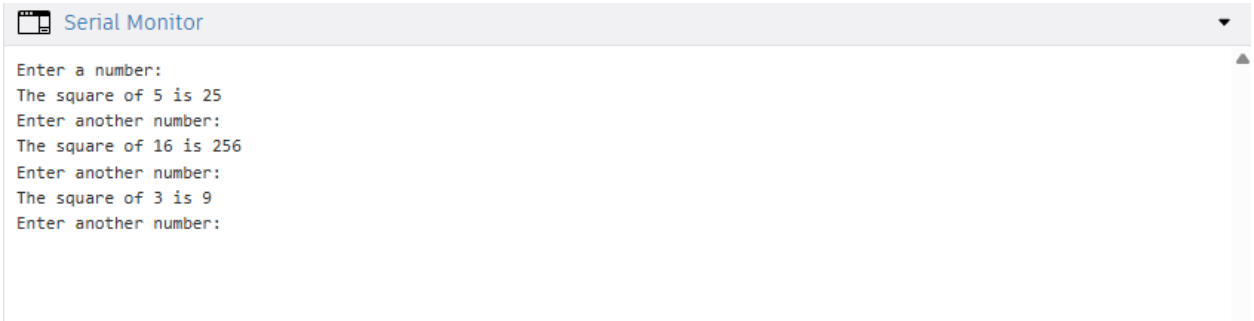
Experiment 8

Write a program that asks the user for a number and outputs the number squared that is entered

Program: (Arduino Uno)

```
void setup() {
    Serial.begin(9600);
    Serial.println("Enter a number: ");
  }
 void loop() {
    if (Serial.available() > 0) {
      int num = Serial.parseInt();
      int squared = num * num;
      Serial.print("The square of ");
      Serial.print(num);
      Serial.print(" is ");
      Serial.println(squared);
      Serial.println("Enter another number: ");
    }
  }
```

Output:



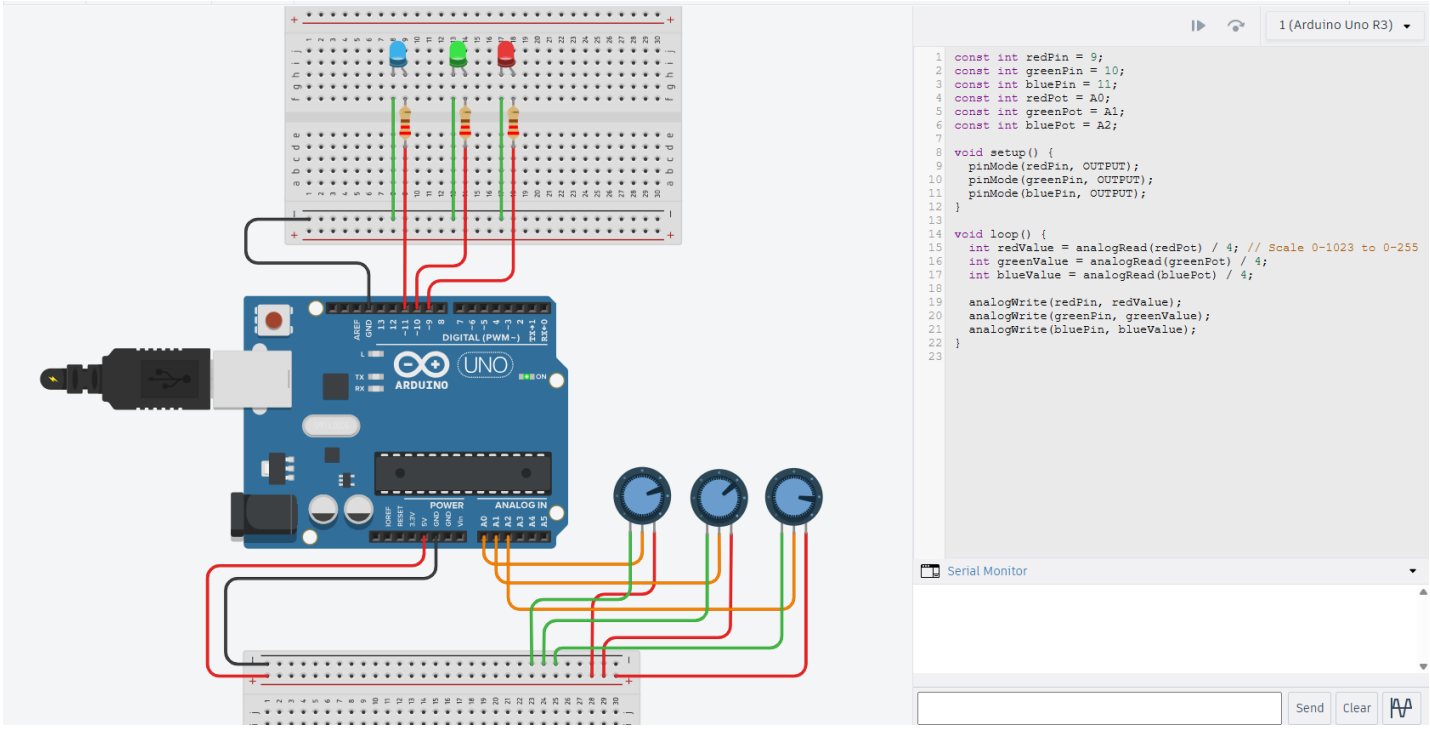
Experiment 9

Write a program to control the color of the LED by turning 3 different potentiometers. One will be read for the value of Red, one for the value of Green, and one for the value of Blue

Program:

```
const int redPin = 9;
const int greenPin = 10;
const int bluePin = 11;
const int redPot = A0;
const int greenPot = A1;
const int bluePot = A2;

void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
}

void loop() {
  int redValue = analogRead(redPot) / 4; // Scale 0-1023 to 0-255
  int greenValue = analogRead(greenPot) / 4;
  int blueValue = analogRead(bluePot) / 4;

  analogWrite(redPin, redValue);
  analogWrite(greenPin, greenValue);
  analogWrite(bluePin, blueValue);
}
```

Output:


1 (Arduino Uno R3)
const int redPin = 9;
const int greenPin = 10;
const int bluePin = 11;
const int redPot = A0;
const int greenPot = A1;
const int bluePot = A2;
void setup() {
pinMode(redPin, OUTPUT);
pinMode(greenPin, OUTPUT);
pinMode(bluePin, OUTPUT);
}
void loop() {
int redValue = analogRead(redPot) / 4; // Scale 0-1023 to 0-255
int greenValue = analogRead(greenPot) / 4;
int blueValue = analogRead(bluePot) / 4;
analogWrite(redPin, redValue);
analogWrite(greenPin, greenValue);
analogWrite(bluePin, blueValue);
}
Serial Monitor
Send
Clear
ARDUINO
UNO
DIGITAL (PWM~)
POWER
ANALOG IN