**Basic Challenge Requirements (Level 1 to Level 2)**

1. Write an Arduino program that contains the required “Setup” and “Loop” procedures.
2. Define an integer variable to hold the pin position for an external “red” LED
3. Define an integer variable to hold the pin position for an external “green” LED
4. Add code to blink the “red” LED for one second and then blink the “green” LED for one second.
5. Cut and copy your program code below this line and submit to your GitHub repository.

**Modified code:**

/\*\*/int redPin = 12; // Red LED connected to digital pin 12int greenPin = 11; // Green LED connected to digital pin 11void setup(){ // run once, when the sketch starts pinMode(redPin, OUTPUT); // sets the digital pin as output pinMode(greenPin, OUTPUT); // sets the digital pin as output}void loop(){ // run over and over again digitalWrite(redPin, HIGH); // sets the Red LED on digitalWrite(greenPin, HIGH); // sets the Green LED on delay(1000); // waits for a second digitalWrite(redPin, LOW); // sets the Red LED off digitalWrite(greenPin, LOW); // sets the Green LED off delay(1000);}

**Standard Challenge Requirements (Level 3)**

1. Modify your program to read and write character strings from the serial monitor.
2. If the user types “red” then run the code to turn on the “red” LED.
   1. Also print “Red LED is On” to the serial monitor.
   2. Also make sure the “green” LED is off.
3. If the user types “green” then run the code to turn on the “green” LED.
   1. Also print “Green LED is On” to the serial monitor.
   2. Also make sure the “red” LED is off.
4. If the user types something other than “red” or “green” then run the code to turn on both LEDs off.
   1. Also print “Both LEDs are Off” to the serial monitor.
5. Cut and copy your program code below this line and submit to your GitHub repository.

**Modified code:**

/\*\*/int redPin = 12; // Red LED connected to digital pin 12int greenPin = 11; // Green LED connected to digital pin 11void setup(){ // run once, when the sketch startsSerial.begin(9600); pinMode(redPin, OUTPUT); // sets the digital pin as output pinMode(greenPin, OUTPUT); // sets the digital pin as output}void loop(){ // run over and over again int val = 0; //waiting for input while(Serial.available()==0); val=Serial.parseInt(); //read Int or parseFloat for...Float... Serial.println(val); for(int i=0;i>val;i++) digitalWrite(redPin, HIGH); // sets the Red LED on digitalWrite(greenPin, HIGH); // sets the Green LED on delay(1000); // waits for a second digitalWrite(redPin, LOW); // sets the Red LED off digitalWrite(greenPin, LOW); // sets the Green LED off delay(1000);}

**Enhanced Challenge Requirements (Level 4)**

1. Modify your program to read numbers from the serial monitor.
2. If the number is even then blink the “green” LED the number of times.
3. If the number is odd then blink the “red” LED the number of times.
4. Cut and copy your program code below this line and submit to your GitHub repository.

byte b;long i = 0;int redPin = 12; // Red LED connected to digital pin 12int greenPin = 11; // Green LED connected to digital pin 11void setup(){ // run once, when the sketch startsSerial.begin(9600);Serial.setTimeout(10); pinMode(redPin, OUTPUT); // sets the digital pin as output pinMode(greenPin, OUTPUT); // sets the digital pin as output}void loop(){ // run over and over again int val = 0; //waiting for input while(Serial.available()==0); i = Serial.parseInt(); val=Serial.parseInt(); //read Int or parseFloat for...Float... Serial.println(val); for(int i=0;i>val;i++) Serial.print("i =");Serial.println(i); digitalWrite(redPin, HIGH); // sets the Red LED on digitalWrite(greenPin, HIGH); // sets the Green LED on delay(1000); // waits for a second digitalWrite(redPin, LOW); // sets the Red LED off digitalWrite(greenPin, LOW); // sets the Green LED off delay(1000);}