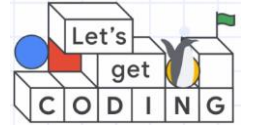


LYS-2019



# Arduino Uygulama Çalışması-3

## RGB Led:Renk Uzayı

**Sabitler :** Bir program içerisinde bir defa değer atandıktan sonra bir daha değeri değiştirilmeyecek olan ifadeler sabit olarak adlandırılırlar.

Sabitler, değişkenler gibi tanımlanırlar sadece başa İngilizce sabit anlamına gelen constant kelimesinin kısaltılmış şekli olan “**const**” kelimesi getirilir.

Örn:

```
const char RAFADAN = 'R';    //char tipinde bir sabit  
const int sure =1250;        // int tipinde bir sabit
```

Program içerisinde sabitler tanımlandıkları sabit ismiyle kullanılırlar ve bu isim kullanıldığı her yerde atandığı ilk değere karşılık gelir.

Sabitler değişkenlerin aksine RAM bellekte değil Arduino’nun program belleğinde saklanırlar. RAM belleğin etkin kullanımı açısından program boyunca değişmeyecek değerlerin sabit olarak tanımlanması daha uygundur.

Sabitler direktif kullanılarak da tanımlanabilirler. Bunun için “**define**” direktifi kullanılır. Direktifler Arduino işlemcisine değil de derleyiciye komut vere ifadelerdir ve “**#**” karakteri ile başlarlar.

Örn:

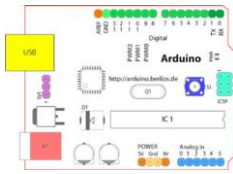
```
#define RAFADAN    'R'  
#define SURE      1205
```

Ayrıca “define” direktifi makro yazmak için de kullanılır. Derleyici bu direktifi kod içerisinde herhangi bir yerde gördüğünde “define” den sonra belirtilen isimle sonrasında yazılan ifadeleri yer değiştirir. Kullanımda bu hususa dikkat edilmelidir.

```
Örn:  delay(SURE);          //yukarıdaki tanımlamaya göre 1205 mS gecikme sağlanır.  
  
#define led_yak(led)    digitalWrite(led,HIGH) //makro tanımlama  
  
led_yak(8);              //8 nolu pine bağlı led yakılır.
```

### Fonksiyonlar

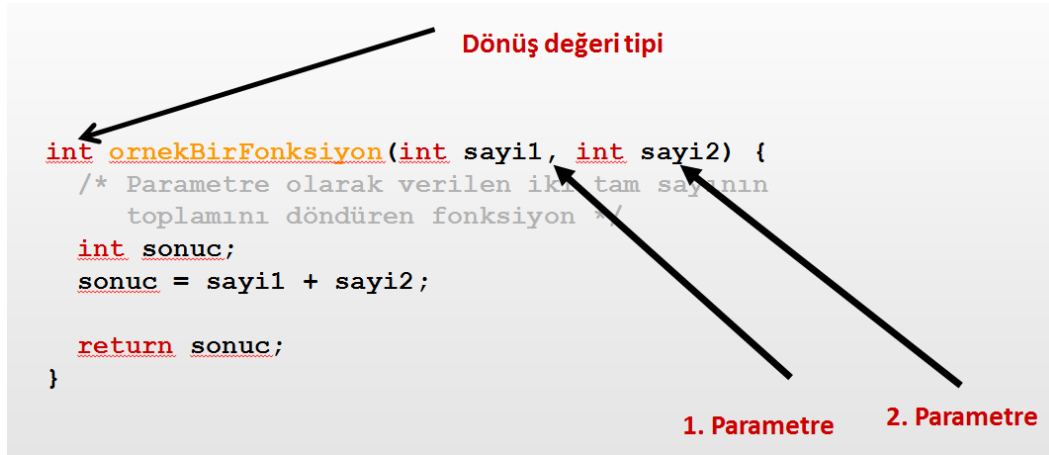
Bir programın bütünlüğü içerisinde, programa ait belirli bir amaç için yazılan kod parçalarıdır. Kendi başlarına bir anlam ifade etmezlerken bir program içerisinde belirlenen amaçlara uygun işlemleri yerine getirebilirler.



LYS-2019



Fonksiyonlar parametre adı verilen giriş değişkenlerine sahip olabilirler. Böylece verilen giriş değerine göre işlemler yaparlar. İşlem sonucu bir çıkış ifadesi döndürülecekse fonksiyonlar bir dönüş tipine ihtiyaç duyarlar. Bu durum fonksiyonlar tanımlanırken belirlenir.

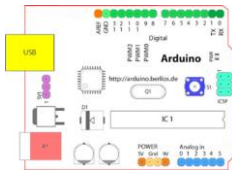


Fonksiyonlar programlarımızın işçileridir. Bir değişken gibi tanımlanırlar. Eğer dönüş değeri olmayacaksa, dönüş değeri tipine "void" (boş) yazılır.

```
void setColor(int red, int green, int blue)  
{  
    analogWrite(redPin, red);  
    analogWrite(greenPin, green);  
    analogWrite(bluePin, blue);  
}
```

## Uygulama:

```
1  
2 const int redPin = 5;      //sabit değeri tanımlama  
3 const int greenPin = 3;  
4  
5 #define bluePin 6         // sabit olarak tanımlama  
6 #define rgb_led(r,g,b) setColor(r,g,b) //makro tanımlama  
7  
8 byte r_deger=0,g_deger=0,b_deger=0;  
9  
10 void setup()  
11 {  
12     pinMode(redPin, OUTPUT);  
13     pinMode(greenPin, OUTPUT);  
14     pinMode(bluePin, OUTPUT);  
15  
16     randomSeed(analogRead(0));  
17 }
```



LYS-2019



```
19 void loop()
20 {
21     setColor(0, 255, 0); // green
22     delay(1000);
23     setColor(0, 0, 255); // blue
24     delay(1000);
25     setColor(255, 255, 0); // yellow
26     delay(1000);
27     setColor(80, 0, 80); // purple
28     delay(1000);
29     setColor(0, 255, 255); // aqua
30     delay(1000);
31     rgb_led(255, 255, 255); // beyaz
32     delay(1000);
33 }

41 // Dönüş değeri olmayan bir fonksiyon
42 void setColor(int red, int green, int blue)
43 {
44     analogWrite(redPin, red);
45     analogWrite(greenPin, green);
46     analogWrite(bluePin, blue);
47 }
```

2. İşlem yukarıdaki loop içersine mevcut kodları yorumyaparak aşağıdaki kodları yazarak tekrar deneyin.

```
19 void loop()
20 {
21     // setColor(0, 255, 0); // green
22     // delay(1000);
23     // setColor(0, 0, 255); // blue
24     // delay(1000);
25     // setColor(255, 255, 0); // yellow
26     // delay(1000);
27     // setColor(80, 0, 80); // purple
28     // delay(1000);
29     // setColor(0, 255, 255); // aqua
30     // delay(1000);
31     // rgb_led(255, 255, 255); // beyaz
32     // delay(1000);
33
34     //2.işlem
35     r_deger=random(0,255);
36     g_deger=random(0,255);
37     b_deger=random(0,255);
38     rgb_led(r_deger, g_deger, b_deger);
39     delay(250);
40 }
```

NOT: RGB şerit ledlerle aynı uygulama nasıl yapılabilir ? Araştırınız.