

Bu kitaba sığmayan daha neler var!

eba
www.eba.gov.tr



Karekodu okut, bu kitapla ilgili EBA içeriklerine ulaş!

Teknoloji Yolculuğumuz

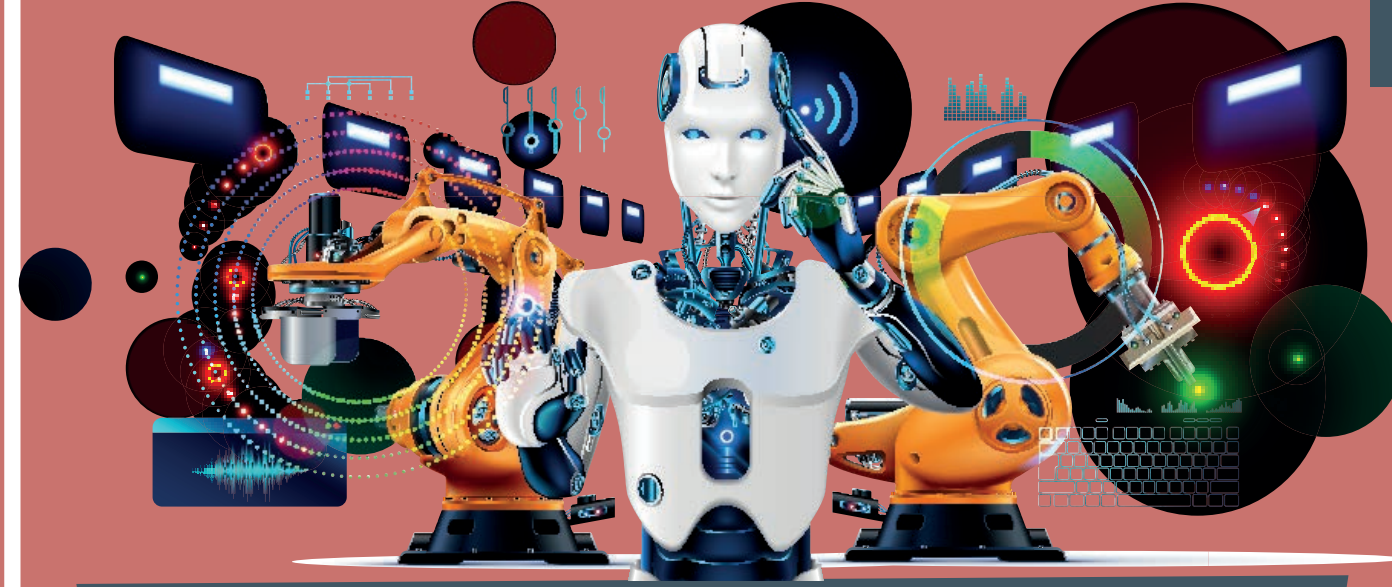


**BU DERS KİTABI MİLLÎ EĞİTİM BAKANLIĞINCA
ÜCRETSİZ OLARAK VERİLMİŞTİR.
PARA İLE SATILAMAZ.**

Bandrol Uygulamasına İlişkin Usul ve Esaslar Hakkında Yönetmeliğin Beşinci Maddesinin İkinci Fıkrası Çerçevesinde Bandrol Taşınması Zorunlu Değildir.

MESLEKİ VE TEKNİK ANADOLU LİSESİ

BİYOMEDİKAL CİHAZ TEKNOLOJİLERİ ALANI



ROBOTİK VE KODLAMA

11-12

DERS MATERYALİ



BİYOMEDİKAL CİHAZ TEKNOLOJİLERİ ALANI
ROBOTİK VE KODLAMA 11-12
DERS MATERYALİ

MESLEKİ VE TEKNİK ANADOLU LİSESİ

BİYOMEDİKAL CİHAZ TEKNOLOJİLERİ ALANI

ROBOTİK VE KODLAMA

11-12

DERS MATERYALİ

YAZARLAR

Engin KILIÇ
Erol OKUMUŞ
Murat KARATAŞ
Onur ÇAKAR
Tuğrul YILDIZHAN



Her hakkı saklıdır ve Millî Eğitim Bakanlığına aittir. Ders materyalinin metin, soru ve şekilleri kısmen de olsa hiçbir surette alınıp yayımlanamaz.

Dil Uzmanı
Melek DEMİR

Program Geliştirme Uzmanı
Dr. Eda ÖZ

Rehberlik Uzmanı
Cahide GÜRNAS KAFALI

Ölçme ve Değerlendirme Uzmanı
Özlem ÇABUKEL

Görsel Tasarım Uzmanı
Özden ALTUN

HAZIRLAYANLAR



İSTİKLÂL MARŞI

Korkma, sönmez bu şafaklarda yüzen al sancak;
Sönmeden yurdumun üstünde tüten en son ocak.
O benim milletimin yıldızıdır, parlayacak;
O benimdir, o benim milletimindir ancak.

Çatma, kurban olayım, çehreni ey nazlı hilâl!
Kahraman ırkıma bir gül! Ne bu şiddet, bu celâl?
Sana olmaz dökülen kanlarımız sonra helâl.
Hakkıdır Hakk'a tapan milletimin istiklâl.

Ben ezelden beridir hür yaşadım, hür yaşarım.
Hangi çılgın bana zincir vuracakmış? Şaşarım!
Kükremiş sel gibiyim, bendimi çiğner, aşarım.
Yırtarım dağları, enginlere sığmam, taşarım.

Garbın âfâkını sarmışsa çelik zırhlı duvar,
Benim iman dolu göğsüm gibi serhaddim var.
Ulusun, korkma! Nasıl böyle bir imanı boğar,
Medeniyet dediğin tek dişi kalmış canavar?

Arkadaş, yurduma alçakları uğratma sakın;
Siper et gövdeni, dursun bu hayâsızca akın.
Doğacaktır sana va' dediği günler Hakk'ın;
Kim bilir, belki yarın, belki yarından da yakın.

Bastığın yerleri toprak diyerek geçme, tanı:
Düşün altındaki binlerce kefensiz yatanı.
Sen şehit oğlusun, incitme, yazıktır, atanı:
Verme, dünyaları alsan da bu cennet vatanı.

Kim bu cennet vatanın uğruna olmaz ki feda?
Şüheda fışkıracak toprağı sıksan, şüheda!
Cânı, cânânı, bütün varımı alsın da Huda,
Etmesin tek vatanımdan beni dünyada cüda.

Ruhumun senden İlahî, şudur ancak emeli:
Değmesin mabedimin göğsüne nâmahrem eli.
Bu ezanlar -ki şehadetleri dinin temeli-
Ebedî yurdumun üstünde benim inlemeli.

O zaman vecd ile bin secde eder -varsa- taşım,
Her cerâhamdan İlahî, boşanıp kanlı yaşım,
Fışkırır ruh-ı mücerret gibi yerden na'sım;
O zaman yükselerek arşa değer belki başım.

Dalgalan sen de şafaklar gibi ey şanlı hilâl!
Olsun artık dökülen kanlarımın hepsi helâl.
Ebediyyen sana yok, ırkıma yok izmihlâl;
Hakkıdır hür yaşamış bayrağımın hürriyyet;
Hakkıdır Hakk'a tapan milletimin istiklâl!

Mehmet Âkif Ersoy

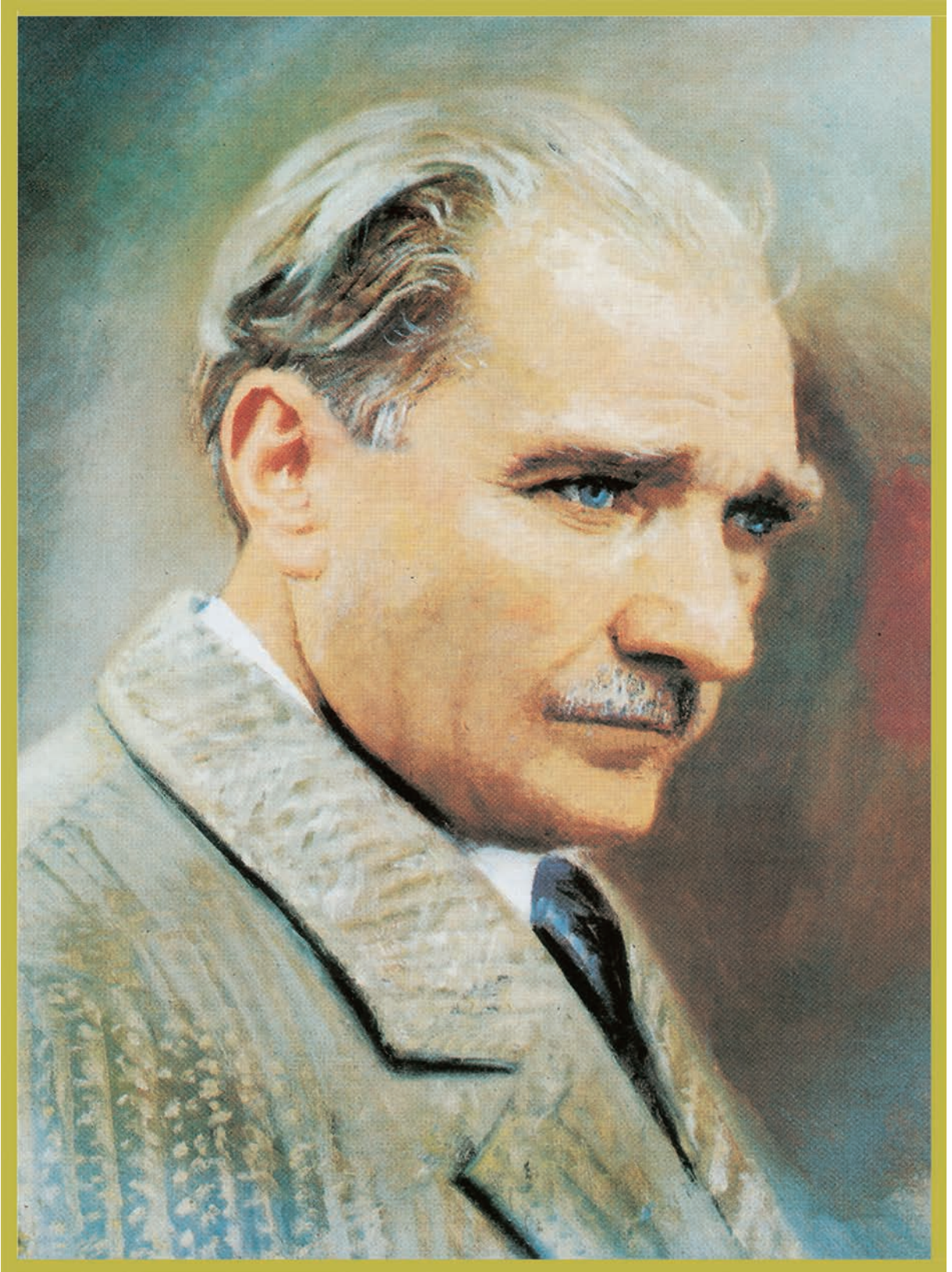
GENÇLİĞE HİTABE

Ey Türk gençliği! Birinci vazifen, Türk istiklâlini, Türk Cumhuriyetini, ilelebet muhafaza ve müdafaa etmektir.

Mevcudiyetinin ve istikbalinin yegâne temeli budur. Bu temel, senin en kıymetli hazinendir. İstikbalde dahi, seni bu hazineden mahrum etmek isteyecek dâhilî ve hâricî bedhahların olacaktır. Bir gün, istiklâl ve cumhuriyeti müdafaa mecburiyetine düşersen, vazifeye atılmak için, içinde bulunacağın vaziyetin imkân ve şeraitini düşünmeyeceksin! Bu imkân ve şerait, çok namüsaît bir mahiyette tezahür edebilir. İstiklâl ve cumhuriyetine kastedecek düşmanlar, bütün dünyada emsali görülmemiş bir galibiyetin mümessili olabilirler. Cebren ve hile ile aziz vatanın bütün kaleleri zapt edilmiş, bütün tersanelerine girilmiş, bütün orduları dağıtılmış ve memleketin her köşesi bilfiil işgal edilmiş olabilir. Bütün bu şeraitten daha elîm ve daha vahim olmak üzere, memleketin dâhilinde iktidara sahip olanlar gaflet ve dalâlet ve hattâ hıyanet içinde bulunabilirler. Hattâ bu iktidar sahipleri şahsî menfaatlerini, müstevlîlerin siyâsî emelleriyle tevhit edebilirler. Millet, fakr u zaruret içinde harap ve bîtap düşmüş olabilir.

Ey Türk istikbalinin evlâdı! İşte, bu ahval ve şerait içinde dahi vazifen, Türk istiklâl ve cumhuriyetini kurtarmaktır. Muhtaç olduğun kudret, damarlarındaki asil kanda mevcuttur.

Mustafa Kemal Atatürk



MUSTAFA KEMAL ATATÜRK

3.
ÖĞRENME BİRİMİ

PROGRAMLAMA
PLATFORMU
UYGULAMALARI



- ÖZET**
- 3.1. LED İLE YAPILAN UYGULAMALAR
 - 3.2. BİYOMEDİKAL SENSÖR UYGULAMALARI
 - 3.3. BİYOMEDİKAL CİHAZ MODÜLLERİNDEKİ UYGULAMALAR
- ANALİZİ YAPILAN KONULAR**
- Besler
 - Devre tasarımı ve simülasyon programı
 - Ölçme
 - LED
 - Nabız sensörü
 - Sensör
 - Ultrasonik mesafe sensörü
- YARARLANILAN KAYNAKLAR**
- LED yapısı ve elektrikli devrelerdeki işlevleri
 - Ohm Kanunu
 - Dirençlerin renk kodları
 - Devre tasarımı programı (elektrotronik devreleri kurulumu)
 - Devre tasarımı ve simülasyon programıyla devre kurup çalıştırma
 - Mikrodenetleyici ile LED uygulamaları
 - Mesafe sensörü yapısı ve uygulamaları
 - Besler yapısı ve uygulamaları
 - Nabız sensörü yapısı ve kullanımı
 - İşık etkili dirençli devrelerin kullanımı
 - Sıcaklık sensörü kullanımı

3. PROGRAMLAMA PLATFORMU UYGULAMALARI	71
3.1. LED İLE YAPILAN UYGULAMALAR	72
3.1.1. LED Yapısı	72
3.1.2. LED Çeşitleri	72
3.1.3. Elektrik Akımı	73
3.1.4. Elektrik Akım Birimleri ve Dönüşümleri	73
3.1.5. Gerilim	74
3.1.6. Direnç	74
3.1.7. Direnç Renk Kodları	74
3.1.8. Ohm Kanunu	76
3.1.9. Breadboard ve Yapısı	77
3.1.10. Devre Tasarım ve Simülasyon Programını Kullanmak	78
3.2. BİYOMEDİKAL SENSÖR UYGULAMALARI	89
3.2.1. Mesafe Sensörü Uygulaması	89
3.2.2. Buzzer Uygulaması	91
3.2.3. Nabız Sensörü Uygulaması	95
3.3. BİYOMEDİKAL CİHAZ MODÜLLERİNDEKİ UYGULAMALAR	98
3.3.1. Işık Etkili Direnç Sensörü Uygulaması	98
3.3.2. Sıcaklık Sensörü Uygulaması	99
ÖLÇME VE DEĞERLENDİRME	101

CEVAP ANAHTARI	104
KAYNAKÇA	105

DERS MATERYALİNİN TANITIMI

1. ÖĞRENME BİRİMİ

GÖRSEL BLOK PROGRAMLAMA

Öğrenme birimi adını gösterir.
Öğrenme birimi numarasını gösterir.
Öğrenme birimi konularını gösterir.

KONULAR

1.1. BLOK TABANLI PROGRAMLAMA

NELER ÖĞRENECEKSİNİZ?

- Yazılım kavramını tanımlama
- Blok tabanlı programlama kavramını tanımlama
- Blok tabanlı programın kurulumunu yapma

TEMEL KAVRAMLAR

- Animasyon

Öğrenme biriminde öne çıkan kavramları gösterir.

Öğrenme biriminde neler öğrenileceği ile ilgili ön bilgileri gösterir.

HAZIRLIK ÇALIŞMALARI

1. Blok tabanlı programların hangileri olduğunu ve bu programlarla neler yapılabileceğini arkadaşlarınızla paylaşınız.

Öğrenme biriminin başında yapılacak ön çalışmaları gösterir.



Araştırma

Biyomedikal cihaz teknolojileri alanında kullanılacak yeni blok bileşenlerinin neler olabileceğini araştırınız.

Konu hakkında yapılacak araştırma görevlerini gösterir.



Sıra Sizde

Bulut kuklaları ve gökyüzü dekoru ekleyerek katmanlı yapıya göre gökyüzünde bulutların hareketinin animasyonunu yapınız.

Konuları pekiştirmek için yapılan faaliyetleri gösterir.



1. Uygulama

İşlem adımlarına göre yeşil bayrak tıklandığında bahçede yürüyen panda kuklası uygulamasını yapınız.

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını "bahçede yürüyen panda kuklası" olarak belirleyiniz.
- 3. Adım:** Görsel 1.33'te yer alan "camping" dekorunu sahneye ekleyiniz.
- 4. Adım:** Yeşil bayrağa tıklandığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıklandığında" blokunu kodlama alanına sürükleyiniz.
- 5. Adım:** Hareket blokunu kullanarak panda kuklasının konumunu x:-195, y:-123 olarak ayarlayınız.
- 6. Adım:** Program durduruluncaya kadar işlemin devam etmesi için Kontrol bloklarının altındaki "Sürekli tekrarla" komutunu ekleyip panda kuklasının hareketlerini belirleyecek komutları bu blok arasına yazınız.
- 7. Adım:** Görünüm blokundan "Sonraki kostüm" komutunu seçiniz.

Teorik bilgilerin pekiştirilmesi amacıyla öğrencilerin aktif olarak katıldığı ders içi etkinlikleri içerir.

DERS MATERYALİNİN TANITIMI



ÖLÇME VE DEĞERLENDİRME

Öğrenme birimi sonundaki ölçme ve değerlendirme çalışmalarını gösterir.

GENEL AĞ KAYNAKÇASI VE GÖRSEL KAYNAKÇASI



Öğrenme biriminde kullanılan Genel Ağ Kaynakça ve Görsel Kaynakça listesini gösterir.

Öğrenme biriminde kullanılan Genel Ağ Kaynakça ve Görsel Kaynakça listesini karekodla gösterir.

GENEL AĞ KAYNAKÇASI

<https://bidb.itu.edu.tr> Erişim Tarihi: 22.10.2022 Saat: 09.00

<https://www.tubitak.gov.tr> Erişim Tarihi: 22.10.2022 Saat: 09.10

<https://www.btkakademi.gov.tr> Erişim Tarihi: 22.10.2022 Saat: 09.15

<https://sozluk.gov.tr> Erişim Tarihi: 22.10.2022 Saat: 09.20

<https://www.tdk.gov.tr> Erişim Tarihi: 25.10.2022 Saat: 09.25

Öğrenme materyali yazımında kullanılan Genel Ağ Kaynakçasını gösterir.

GÖRSEL KAYNAKÇA

Görsel No	Görsel ID	Görsel No	Görsel ID	Görsel No	Görsel ID
Görsel 1.1	443228062	Görsel 1.31	Komisyon	Görsel 2.1	185170919
Görsel 1.2	86849608	Görsel 1.32	Komisyon	Görsel 2.2	Komisyon
Görsel 1.3	274890299	Görsel 1.33	Komisyon	Görsel 2.3	Komisyon
Görsel 1.4	Komisyon	Görsel 1.34	Komisyon	Görsel 2.4	Komisyon

Öğrenme materyali yazımında kullanılan Görsel Kaynakçayı gösterir.

CEVAP ANAHTARLARI

1. ÖĞRENME BİRİMİ

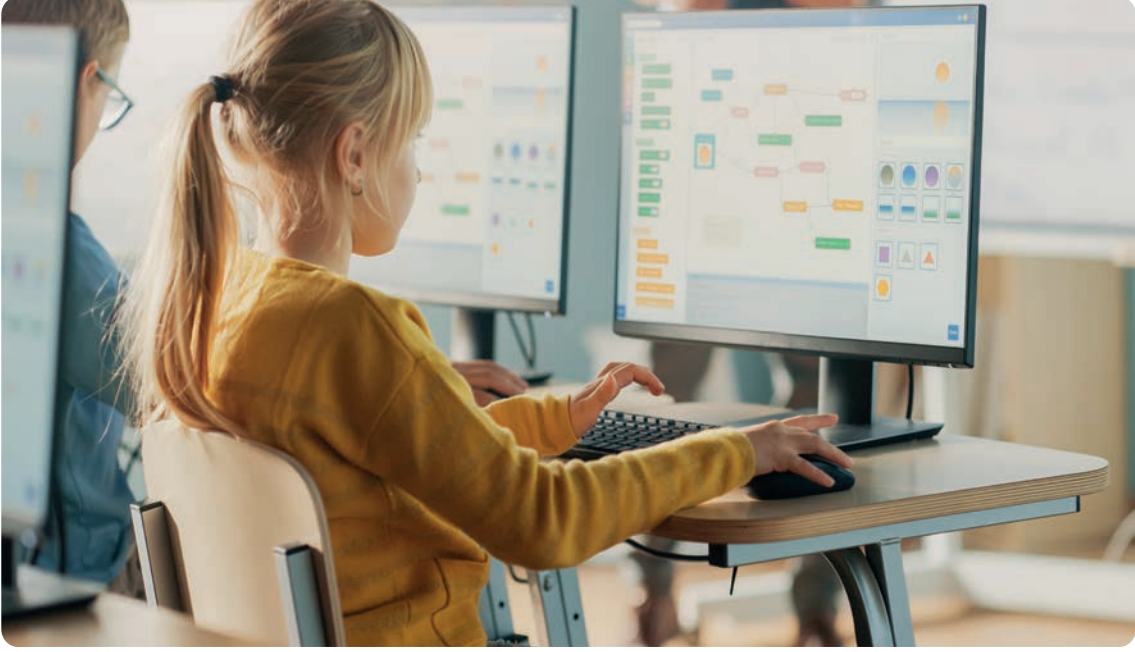
A	1. Doğru	2. Yanlış	3. Doğru	4. Yanlış	5. Yanlış
	6. Doğru	7. Doğru	8. Doğru		
B	9. A	10. E	11. C	12. D	13. A

Öğrenme birimi sonundaki ölçme ve değerlendirme çalışmalarının cevap anahtarlarını gösterir.

Bu ders materyalinde ölçü birimlerinin uluslararası kısaltmaları kullanılmıştır.

1. ÖĞRENME BİRİMİ

GÖRSEL BLOK PROGRAMLAMA



KONULAR

- 1.1. GÖRSEL BLOK PROGRAMLAMA PLATFORMU
- 1.2. GÖRSEL BLOK PROGRAMINI KULLANARAK KODLAMA

ANAHTAR KELİMELE

- Animasyon
- Blok programlar
- Blok tabanlı kodlama
- Blok tabanlı programlama
- Dekor
- Kodlama
- Kostüm
- Kukla
- Oyun

NELER ÖĞRENECEKSİNİZ?

- Yazılım kavramını tanımlama
- Blok tabanlı programlama kavramını tanımlama
- Blok tabanlı programın kurulumunu yapma
- Blok tabanlı programa çevrimiçi kayıt yapma
- Blok tabanlı programı kullanarak karakterler oluşturma
- Blok tabanlı programda yapılan çalışmalara sesler ekleme
- Blok tabanlı programı kullanarak koordinat düzlemi üzerinde şekiller çizme
- Blok tabanlı programı kullanarak matematiksel işlemler yapma
- Blok tabanlı programı kullanarak animasyonlar tasarlama
- Blok tabanlı programı kullanarak bilgisayar oyunları hazırlama
- Blok tabanlı programlamada yapılan işlemlerin çevrimiçi ortamda paylaşımı





HAZIRLIK ÇALIŞMALARI

1. Blok tabanlı programların hangileri olduğunu ve bu programlarla neler yapılabileceğini arkadaşlarınızla paylaşınız.
2. Blok tabanlı programların masaüstü ve internet ortamında kullanımı ile blok tabanlı kodlamanın avantajları neler olabilir? Düşüncelerinizi arkadaşlarınızla paylaşınız.

1.1. GÖRSEL BLOK PROGRAMLAMA PLATFORMU

Robotik kodlamalarda birçok programlama dili kullanılır. Bunlar içinde görsel blok programları bulunur.

1.1.1. Yazılım

Elektronik bileşenlerden oluşan donanım sistemine, belirli bir işi yapması için yazılmış kodlara **yazılım** denir (Görsel 1.1). Türk Dil Kurumu Güncel Türkçe Sözlük, yazılım kavramını “Bir bilgisayarda donanıma hayat veren ve bilgi işlemede kullanılan programlar, yordamlar, programlama dilleri ve belgelemelerin tümü.” olarak vermiştir.

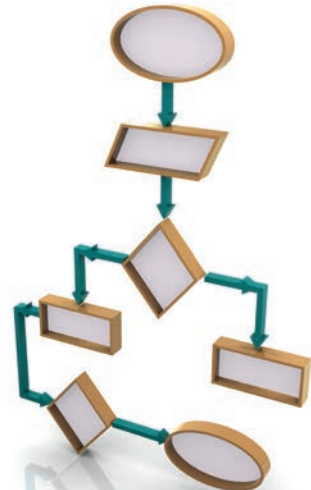


Görsel 1.1: Yazılım

1.1.2. Algoritma

Belirli bir problemi çözmek veya belirli bir amaca ulaşmak için çözüm yolunun adım adım planlanmasına **algoritma** denir (Görsel 1.2). Algoritmalar sadece kodlama işlemlerinde değil hayatın her alanında kullanılır. Örneğin çay demlerken uygulanan adımlar bile bir algoritmadır.

Yazılım geliştirerek bir amaca ulaşmak için öncelikle yapılacak işler adım adım anlamlı parçalara bölünerek planlanır, bir başka deyişle algoritma oluşturulur. Her bir parça, tercih edilen programlama dili ile bilgisayarda yazılır.



Görsel 1.2: Algoritma



1.1.3. Blok Tabanlı Programlama Ortamı

Kodlama yeterliliği, yazılım geliştirmenin yanında günümüz bireylerinin sahip olması gereken bir 21.yy. becerisidir. Bu durum göz önüne alındığında kodlama becerilerini geliştirebilmek için özellikle yeni başlayanlara yönelik basit arayüzü olan, kod yazımı gerektirmeyen ve oyun tasarımına olanak tanıyan araçların kullanılması önerilir (Görsel 1.3).



Görsel 1.3: Blok tabanlı programlama

Blok tabanlı kodlama ortamları, kod satırları yazmak yerine hazır bloklar hâlindeki programlama ifadelerinin sürükleyip bırak yöntemiyle bir araya getirilmesine ve kodlamanın geliştirilmesine olanak sağlar.

Farklı programlama dillerinde ekrana “Merhaba!” mesajı yazdıran kod örnekleri şunlardır:

```
print(Merhaba!) (Python programlama dilinde)
std::cout << "Merhaba!" << std::endl; (C++ dilinde)
System.out.print("Merhaba!"); (Java dilinde)
Console.write("Merhaba!");(C# dilinde)
```

Blok tabanlı kodlama oldukça kolaydır. mBlock'ta tek bir blok kullanılarak “Merhaba!” mesajının yazımı Görsel 1.4'te verilmiştir.



Görsel 1.4: Blok tabanlı programlamada mesaj yazımı



Blok tabanlı programların çevrimiçi (online) paylaşım özelliği sayesinde yapılan çalışmalar programı kullanan herkesle paylaşarak sosyalleşmeye de imkân sağlar. Bu program, internet üzerinden çevrimiçi kullanılabileceği gibi bilgisayara indirilip kurulum yapılarak çevrimdışı (offline) da kullanılabilir.

1.1.4. Blok Tabanlı Programlama Hesabı Oluşturmak

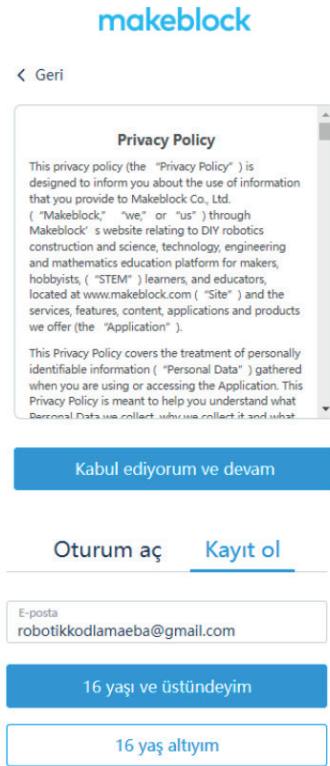
Blok tabanlı programlama hesabı oluşturmak için <https://ide.mblock.cc/mBlock> web sitesi adresiyle ulaşılan portalda sağ üst köşedeki avatar düğmesine tıklanır (Görsel 1.5).



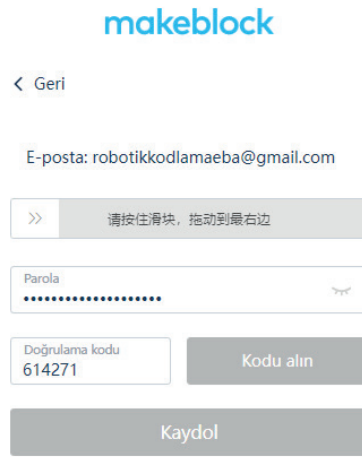
Görsel 1.5: Blok tabanlı programlama hesabı oluşturma

Çevrimiçi kayıt işlemi şu adımlar takip edilerek gerçekleştirilir:

- Kullanılacak e-posta adresi belirlenir.
- Yaş bilgisi girilir.
- Kullanım şartları okunarak kabul edilir.
- Oluşturulacak hesap için parola belirlenir.
- E-posta adresi doğrulaması yapılarak hesap oluşturulur (Görsel 1.6, Görsel 1.7).



Görsel 1.6: Blok tabanlı programlama sitesi kayıt adımları



Görsel 1.7: Blok tabanlı programlama sitesi kayıt sonu



Sıra Sizde

Blok tabanlı programlama sitesine kayıt işleminizi yapınız.

1.1.5. Blok Tabanlı Programlama Çevrimiçi Editörü

Blok tabanlı programlama portalı anlaşılabilir bir arayüze sahiptir. Üyelik bilgileriyle çevrimiçi editöre Görsel 1.8'deki gibi giriş yapılır.

Oturum açıldıktan sonra Görsel 1.9'daki arayüzle karşılaşılır. Bu şekilde mBlock programı internet üzerinden çevrimiçi kullanılabilir. Geliştirilen oyunlar ve uygulamalar, mBlock portalında depolanabilir ve başkalarıyla paylaşılabilir. Bu portalda daha önce paylaşılan uygulamaların geliştirilmesine de katkı sağlanabilir.

makeblock

Oturum aç

Kayıt ol

Email
robotikkodlamaeba@gmail.com

>> 请按住滑块，拖动到最右边

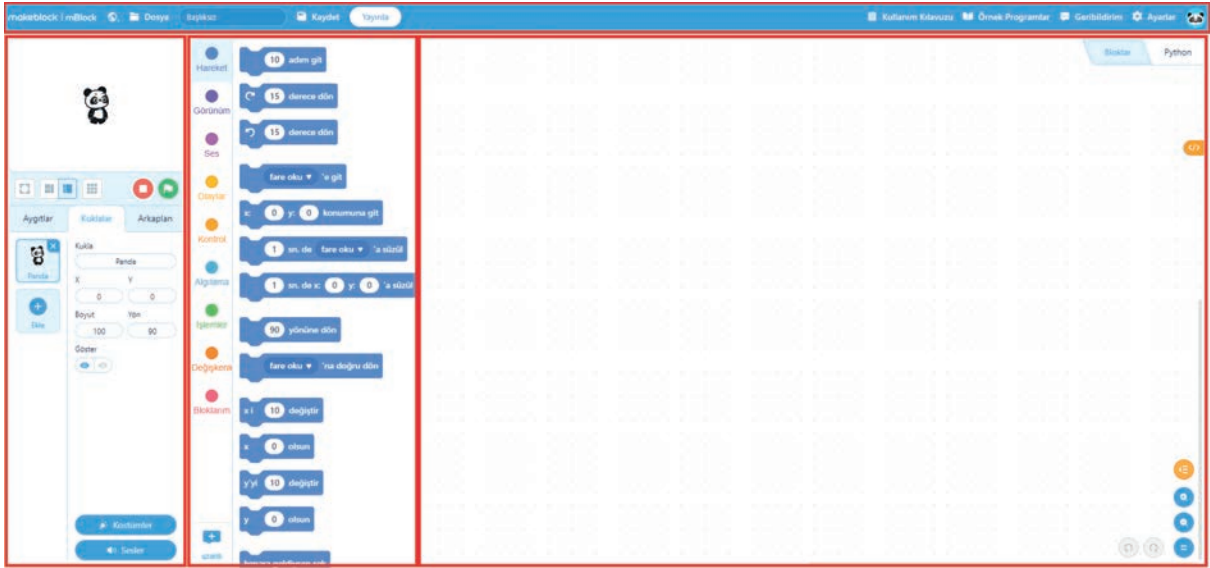
Parola
.....

[Şifreyi mi unuttun?](#)

Oturum aç

Görsel 1.8: Çevrimiçi kodlama sitesinde oturum açma

Ana Menü



Sahne

Kod Blokları, Kostüm ve Sesler

Kodlama Alanı

Görsel 1.9: Çevrimiçi arayüz

mBlock blok tabanlı programlama arayüzü; ana menü, bloklar, kodlama, sahne olmak üzere dört ana bölümden oluşur.



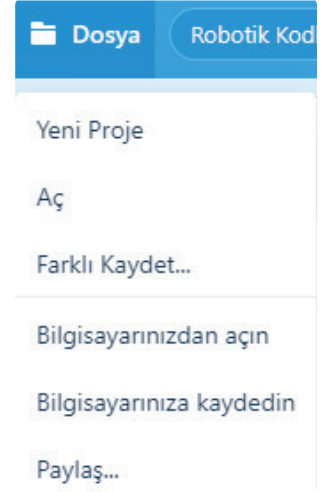
1.1.5.1. Ana Menü



simgesiyle mBlock program arayüzü istenen dilde kullanılabilir.

Görsel 1.10'da görülen dosya ana menüsü seçenekleri kullanılarak şu işlemler yapılır:

- “Yeni Proje” ile yeni bir proje oluşturulur.
- “Aç” ile çevrimiçi olarak daha önce oluşturulan proje açılır.
- “Farklı Kaydet” ile oluşturulan proje farklı isimde çevrimiçi ortama kaydedilir.
- “Bilgisayarınızdan açın” mBlock yazılımı kullanılarak oluşturulan ve bilgisayarda kayıtlı olan çalışma açılır.
- “Bilgisayarınıza kaydedin” ile uygulama bilgisayara kaydedilir.



Görsel 1.10: Dosya ana menüsü


Görsel 1.11'de görülen Düzenle ana menüsü seçeneklerinden şu işlemler yapılır:

- Görsel 1.11'deki gibi metin kutusu içine yeni oluşturulan projenin adı yazılır.
- “Kaydet” ile üzerinde çalışılan projenin çevrimiçi ortama kaydedilmesi sağlanır. Çevrimiçi proje, verilen isimde “Projelerim” sayfasına kaydolur.
- “Yayınla” ile projenin çevrimiçi ortamda diğer geliştiriciler tarafından görülmesi ve geliştirilmesi için paylaşım yapılır.



Görsel 1.11: Ana menü öğeleri


Ana menüde görülen seçeneklerle şu işlemler yapılır:

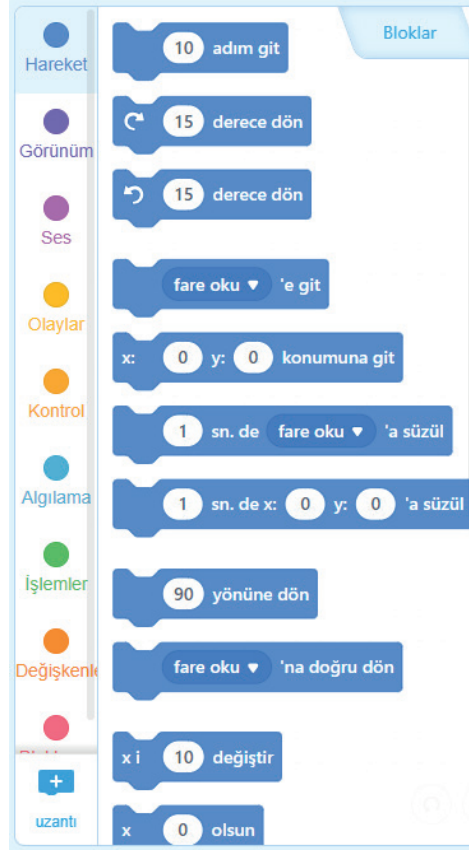
- “Kullanım Kılavuzu”nda mBlock yazılımının kullanım kılavuzu bulunur.
- “Örnek Programlar” seçeneğinde daha önce oluşturulmuş örnek programlar yer alır. Burada kategorilere ayrılan programlar arasından seçim yapılarak daha önce oluşturulmuş program açılabilir.
- “Geri Bildirim” seçeneği ile mBlock yazılım geliştiricilerine geri bildirim gönderilir.
- “Ayarlar” seçeneği ile mBlock yazılım ayarları yapılır.
-  simgesiyle profil ayarlarının yapıldığı sayfalar açılır.

1.1.5.2. Kod Blokları

Karakterleri kodlamak için kullanılan blokların bulunduğu alandır. Kod blokları, karaktere istenen hareketi yaptırabilmek için kolay tasarımıyla kullanıma sunulmuş araçlardır. Algoritmaya göre kodlama yapılabilmesi için gerekli olan tüm bloklar bulunur. Blokların renkleri farklıdır, renklere göre gruplandırma yapılmıştır ve bloklara kolayca erişilir. Ayrıca sağ taraftaki kaydırma çubuğuyla istenen kod blokuna ulaşılabilir (Görsel 1.12). Kod bloklarında bulunan kodlar alt alta sıralanıp, komut dizisi oluşturularak karakterin hareket etmesi, müzik çalması ve işlemler yapması sağlanır.

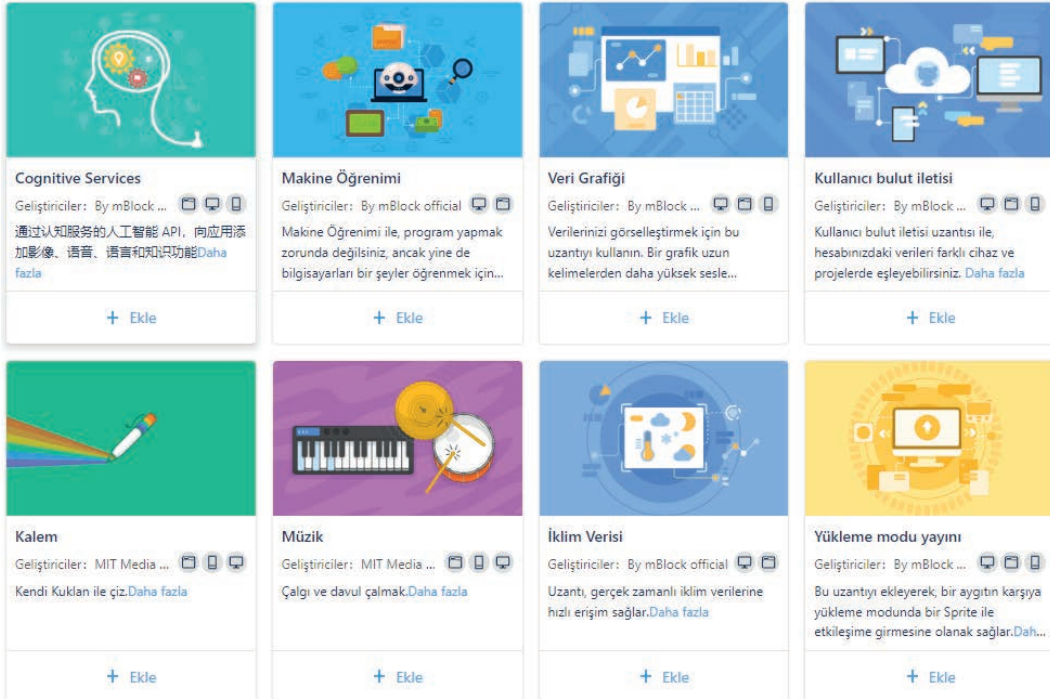


Görsel 1.12'de görülen  düğmesiyle kod bloklarına farklı uzantılar eklenebilir.



Görsel 1.12: Kodlama blokları

Çevrimiçi mBlock yazılımı her açıldığında eklenen uzantı yeniden eklenir (Görsel 1.13).



Görsel 1.13: Yeni blok bileşenleri

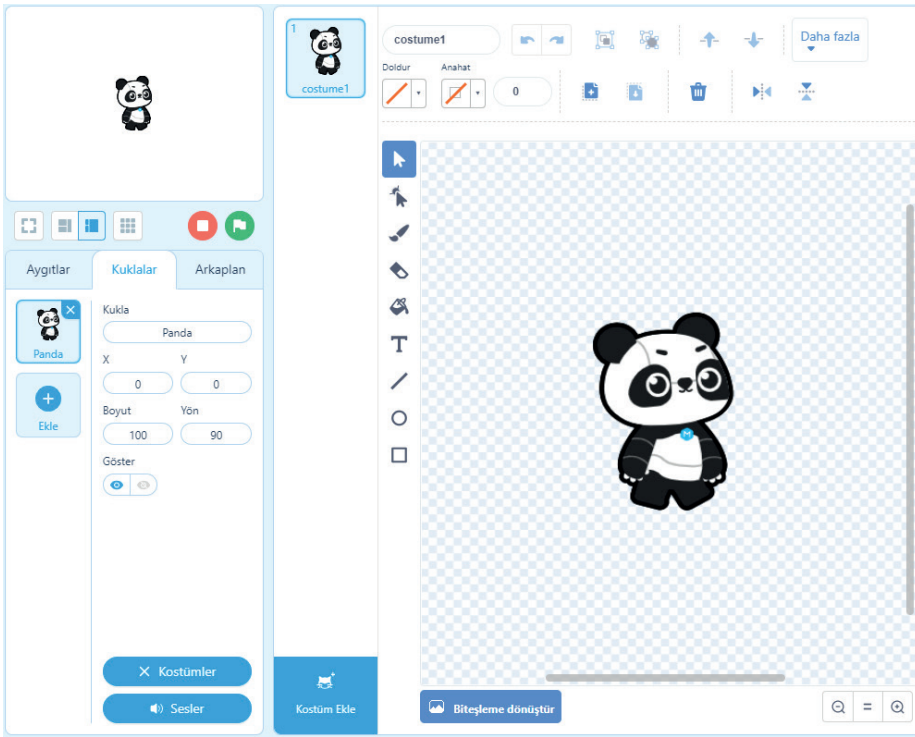


Araştırma

Biyomedikal cihaz teknolojileri alanında kullanılacak yeni blok bileşenlerinin neler olabileceğini araştırınız. Araştırma sonuçlarınızı sınıfta arkadaşlarınızla paylaşınız.

1.1.5.3. Kostümler

Kuklalar sekmesinde bulunan Kostümler butonuna tıklanarak kostümler ekranı açılır. Kostümler bölümünde kütüphanedeki kuklaların kostümlerinden seçim yapılabilir, istenen kostüm çizilebilir, bilgisayardaki bir görsel eklenerek kostüm şeklinde değerlendirilebilir. Mevcut kostümler, X düğmesine basılarak silinebilir veya Kopyala düğmesi ile kostümün aynısından bir tane daha eklenebilir (Görsel 1.14).



Görsel 1.14: Kostümler ekranı

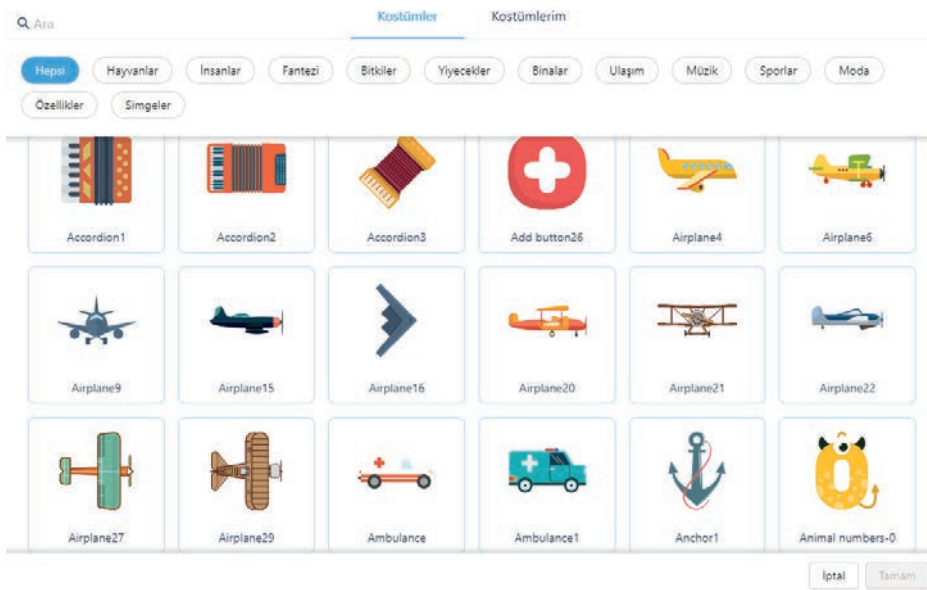
Kostümler sekmesinde bulunan seçenekler şunlardır:

- **Kostüm** Kukla kostümünün adını değiştirmek için kullanılır.
- Kukla üzerine yapılan değişiklikleri ileri veya geri almak için kullanılır.
- Sayfada çizilen kostüm bölümlerinin bir bütün olmasını ve bir bütün olan kostüm bölümlerinin ayrılmasını sağlar.
- Kostümde seçilen katmanın bir katman ileri veya geri gitmesini sağlar.
- Kostümde seçilen katmanın en ileri veya en geri gitmesini sağlar.
- **Doldur** Yazı veya şekillerin rengini seçmek için kullanılır.
- **Anahat** Kontur kalınlığını ve rengini ayarlamak için kullanılır.



- Seçilen bölümü kopyalamak ve yapıştırmak için kullanılır.
- Seçilen bölümü silmek için kullanılır.
- Seçilen bölümün yatay veya dikey çevrilmesi için kullanılır.
- Nesne seçmek için kullanılır.
- Şekli hassas değiştirmek için kullanılır.
- Seçilen renkte fırça ile boyamak için kullanılır.
- Çizim silmek için kullanılır.
- Çizim için otomatik boyamak için kullanılır.
- Yazı yazmak için kullanılır.
- Çizgi, eğri, çember ve kare çizmek için kullanılır.

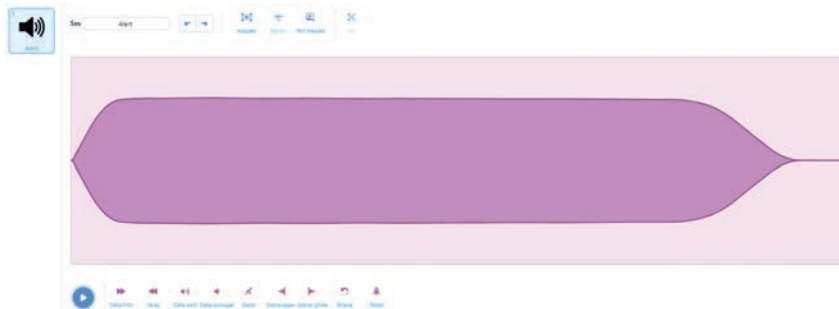
Bir Kostüm Seç simgesine tıklanarak kuklaya farklı kostümler eklenebilir (Görsel 1.15).



Görsel 1.15: Kostüm ekle penceresi

1.1.5.4. Sesler

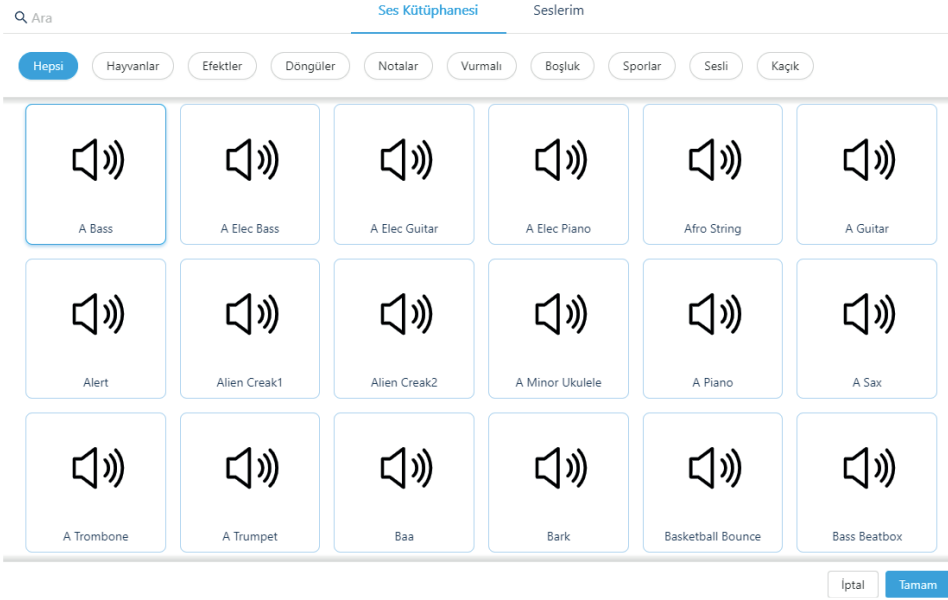
mBlock programında hazırlanan projelerdeki sahne veya karakterlere ses eklemek mümkündür. Bunun için Kuklalar sekmesinde bulunan Sesler butonuna tıklanarak sesler ekranı açılır (Görsel 1.16).



Görsel 1.16: Sesler ekranı



Ses Ekle butonuna tıkladığında açılan ekrandan kütüphanedeki seslerden biri seçilebilir, bilgisayarda yer alan bir ses eklenebilir veya ses kaydı yapılabilir. Bu seslerden biri seçilip kullanılabilir (Görsel 1.17).



Görsel 1.17: Ses ekle penceresi

1.1.5.5. Kodlama

Kodlama işleminde algoritmanın amacına göre ihtiyaç duyulan bloklar sürüklenerek kodlama alanına bırakılır. Kodlama alanına sürüklenen bloklar alt alta takılarak birleştirilir ve bu şekilde blok tabanlı kodlama yapılır. Kodlama alanının sağ alt köşesinde bulunan (+) düğmesine tıklanarak kodlama bloklarının büyük görünmesi, (-) düğmesine tıklanarak kodlama bloklarının küçük görünmesi ve (=) düğmesine tıklanarak kodlama bloklarının normal boyutta görünmesi sağlanır (Görsel 1.18).



Görsel 1.18: Kodlama alanı

1.1.5.6. Sahne

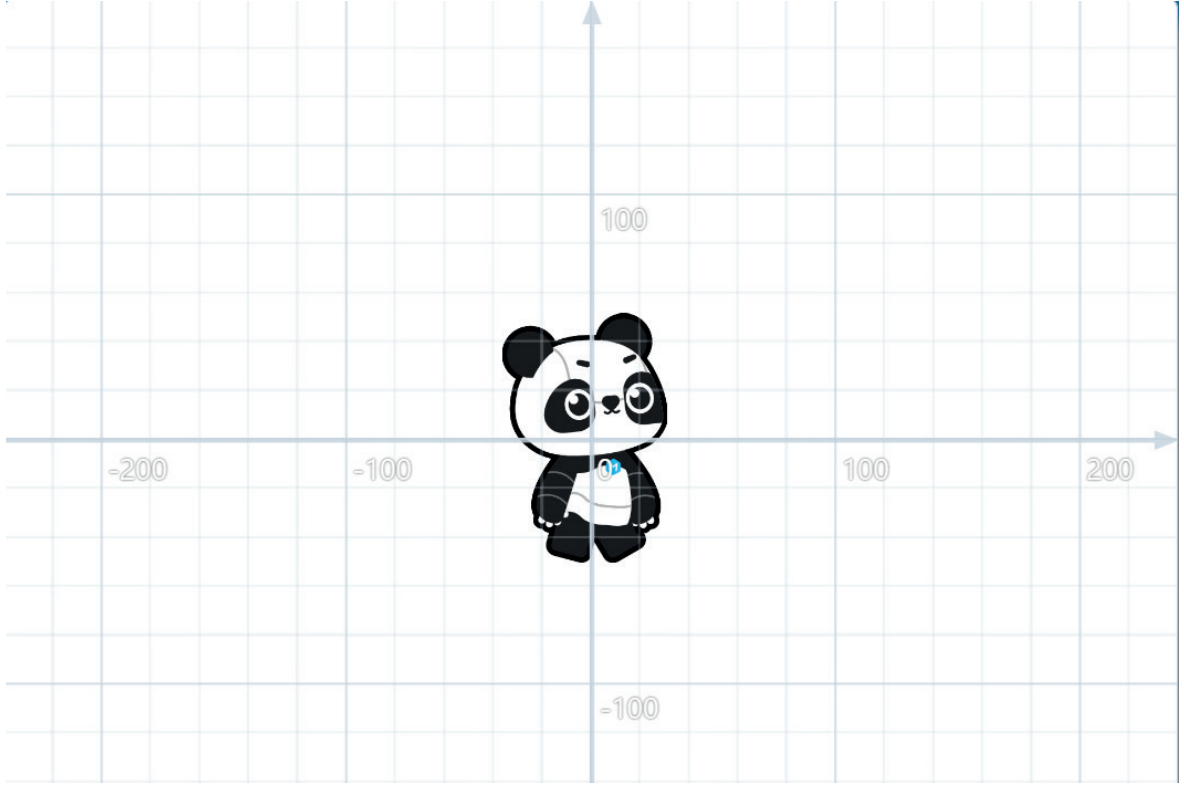
Sahne, uygulamaların ön izlemesinin yapıldığı alandır (Görsel 1.19). mBlock programında hazırlanan her uygulama bu bölümde çalışır. Sahne, geliştirilen uygulamanın canlandırıldığı yerdir.



Görsel 1.19: Sahne alanı













mBlock sahnesinin anlaşılması için koordinat düzlemi kavramı bilinmelidir (Görsel 1.20). mBlock ekranı aslında bir koordinat düzlemidir. Kuklanın sahne üzerinde bulunduğu konum, koordinat sistemi gibi düşünülebilir. mBlock açıldığında kuklalar (0,0) noktasında, bir başka deyişle X ve Y doğrularının kesişim noktasında yer alır. Koordinat sistemi yatay ve dikey iki doğrunun kesişiminden oluşur. X eksenini -240 ile 240, Y eksenini ise -180 ile 180 arasındadır. Kuklalar ileriye doğru gittiğinde X konumu artar, geriye doğru gittiğinde ise X konumu azalır. Yukarıya doğru gerçekleşen bir harekette Y konumu artar, aşağıya iniş durumunda Y konumu azalır.



Görsel 1.20: Koordinat düzlemi

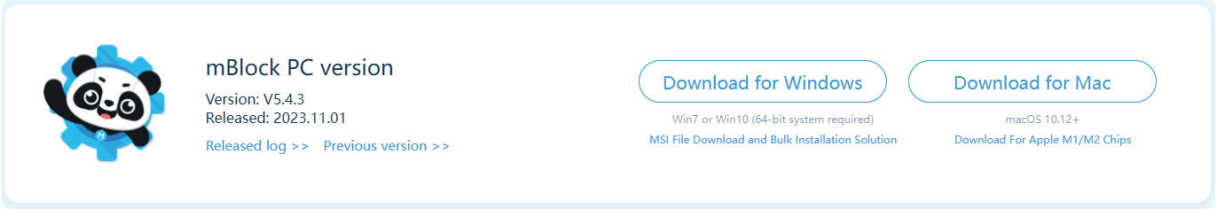
Uygulamanın sahne bölümündeki seçenekler şunlardır:

-  Hazırlanan uygulamanın çalıştırılması için kullanılır.
-  Çalışan uygulamanın durdurulması için kullanılır.
-  Sahne alanının küçülmesini ve kod yazılacak alanın genişlemesini sağlar.
-  Küçülen sahne alanının büyümesini sağlar.
-  Sahne alanının tam ekranda görünmesini sağlar.
-  Tam ekranda görünen sahne alanının normal ekran ölçülerinde görünmesini sağlar.
-  Kukla ismini değiştirmek için kullanılır.
-  Kukla koordinatlarını görmek ve değiştirmek için kullanılır.
-  Kukla boyutunu ve yönünü ayarlamak için kullanılır.
-  Kuklayı sahnede göstermek veya gizlemek için kullanılır.



1.1.6. Blok Tabanlı Programlama Editörünü Bilgisayara Kurmak

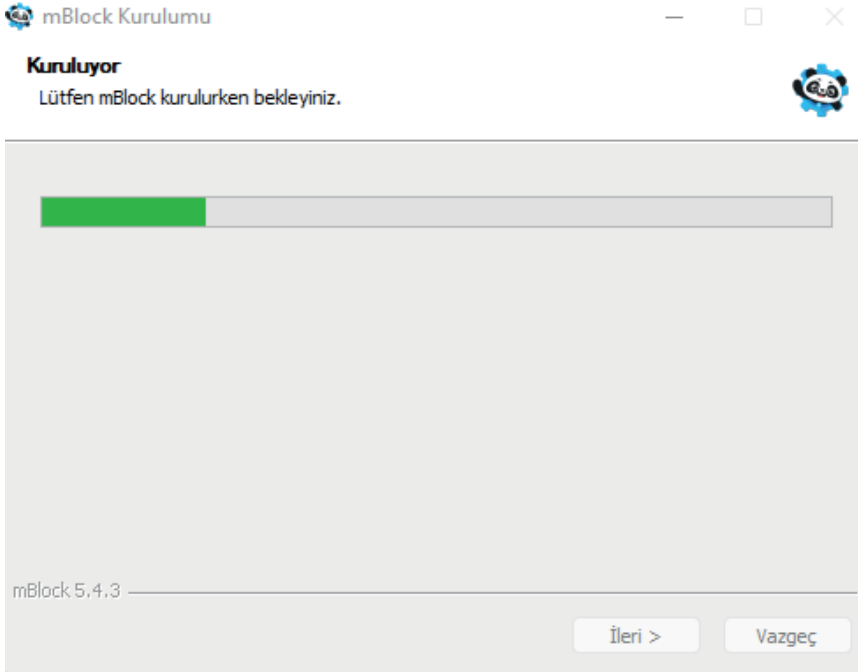
mBlock blok tabanlı kodlama programını çevrimdışı kullanabilmek için <https://www.mblock.cc/en/download/> adresi ziyaret edilip kurulum için gerekli dosya bilgisayara indirilir (Görsel 1.21).



Görsel 1.21: mBlock indirme ekranı

Kurulum yapmak için şu adımlar uygulanır:

- Bilgisayarda kurulu işletim sistemi seçilir.
- Program bilgisayara indirilir, “.exe” uzantılı dosya çalıştırılır ve karşılaşılan yönergeler doğrultusunda kişisel bilgisayara kurulum yapılır (Görsel 1.22).



Görsel 1.22: mBlock kurulumu



Sıra Sizde

mBlock kurulum dosyasını bilgisayarınıza indirerek kurulumu gerçekleştiriniz.



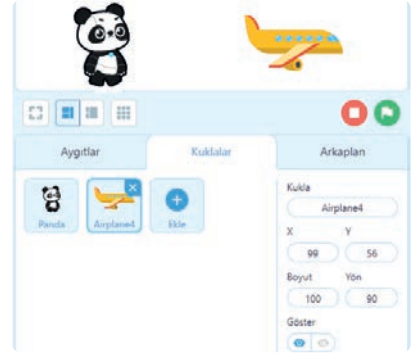
1.2. GÖRSEL BLOK PROGRAMINI KULLANARAK KODLAMA

Görsel blok programı kullanılarak kodlama yapılırken sahneye kukla, kostüm, ses, dekor ve arka plan eklenebilir.

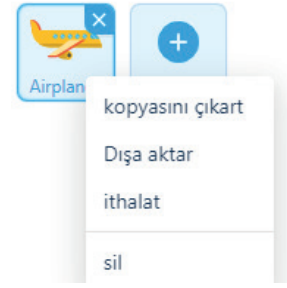
1.2.1. Kukla Ekleme

Sahnenin alt kısmındaki Kuklalar sekmesi içinde Ekle butonuna tıklanıp açılan kukla kütüphanesinden istenen kukla seçilerek sahneye eklenir (Görsel 1.23). Sahneye istenen sayıda kukla eklenebilir. Fare ile kolayca kuklaların ilk konumları belirlenebilir. Sahnenin altındaki kuklalardan hangisi seçiliyse ona kod yazılır. Kostümler penceresinde çizim elemanları kullanılarak kukla çizilir. İstenirse kuklalara farklı kostümler çizilerek bunlar uygulamalarda kullanılabilir veya hazır kostümler tercih edilebilir. Büyüteç işaretiyle arama yapılarak bilgisayardaki görsellerden biri, kukla olarak uygulamaya eklenebilir.

Eklenecek kuklanın üzerine farenin sağ tuşuyla tıklanarak açılan menüden hızlıca kuklanın kopyası elde edilir, kukla silinir, dışa veya içe aktarılır (Görsel 1.24).



Görsel 1.23: Kukla ekleme



Görsel 1.24: Kukla işlemleri

1.2.2. Kostüm Ekleme

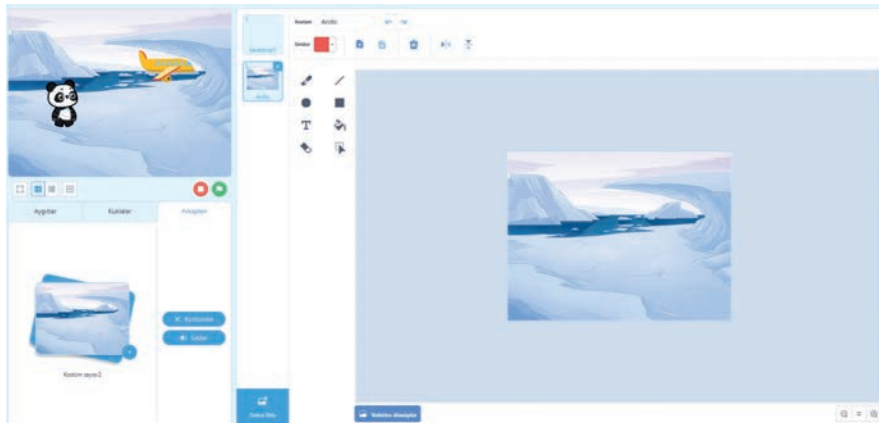
mBlock uygulamaları Kuklalar olarak adlandırılan nesnelere oluşur. Seçilen bir kuklaya farklı kostümler giydirilerek görünüm ve sonraki görünüm değiştirilebilir. Kukla işlemlerinde olduğu gibi Kostümler penceresi açılarak yeni kostümler eklenebilir ve çizilebilir (Görsel 1.25).



Görsel 1.25: Kostüm penceresi butonu

1.2.3. Dekor Ekleme

Sahnenin alt kısmındaki "Arka plan" sekmesi içinde "Dekor Ekle" butonuna tıklanıp, açılan dekor kütüphanesinden istenen dekor seçilerek sahneye eklenir (Görsel 1.26). Arka plan sekmesi içindeyken kodlama alanına gelindiğinde dekorla ilgili kodlama yapılır.

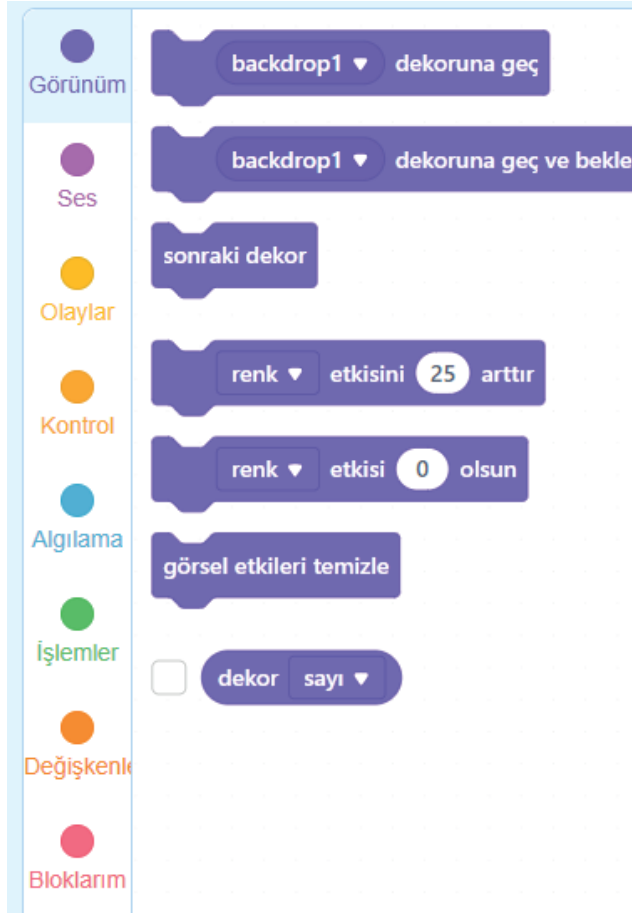


Görsel 1.26: Dekor seçme penceresi



Not

Arka plan kod bloklarında hareket bloklarının olmadığı görülür (Görsel 1.27).



Görsel 1.27: Arka plan kod blokları



Sıra Sizde

Bilgisayarınızdaki görselleri mBlock uygulamasına kukla, kostüm ve dekor olarak ekleyiniz.



1.2.4. Kodlama Blokları

Kuklaların her türlü eylemi yapabilmeleri için kodlama blokları kullanılır.

1.2.4.1. Hareket Blokları

Kuklaların konumu, yönü ve hareketinin belirlenmesi için kullanılan bloklardır. Tablo 1.1'de hareket blokları verilmiştir.

Tablo 1.1: Hareket Blokları

Sıra	Blok	Açıklama
1		Kuklayı belirlenmiş yöne doğru 10 adım hareket ettirir.
2		Kuklanın istenen derece kadar saat yönünde dönmesini sağlar.
3		Kuklanın istenen derece kadar saat yönü tersine dönmesini sağlar.
4		Kuklayı sahnede rastgele bir x, y konumuna götürür.
5		Kuklayı sahnede belirtilen x, y konumuna götürür.
6		Kuklayı belirli bir saniye sürede belirtilen konuma götürür.
7		Kuklayı belirli bir saniye sürede belirtilen konuma götürür.
8		Kuklanın hangi yöne döneceğini belirler (0=yukarı, 90=sağ, 180=aşağı, -90=sol).
9		Kuklayı fare imlecine doğru yönlendirir.
10		Kuklanın bulunduğu konumun x değerini istenen değer kadar artırır veya azaltır.
11		Kuklanın x konumunu istenen değere getirir.
12		Kuklanın bulunduğu konumun y değerini istenen değer kadar artırır veya azaltır.
13		Kuklanın y konumunu istenen değere getirir.
14		Kukla kenara geldiği zaman geldiği yönün tam tersine sekmesini sağlar.
15		Kuklanın sağa veya sola dönme, etrafında dönebilme ve hiç dönmeme ayarlarını yapar.
16	<input type="checkbox"/>	Kuklanın x pozisyonu bilgisini verir. Bu seçenek işaretlendiğinde x değeri ekranda görülür.
17	<input type="checkbox"/>	Kuklanın y pozisyonu bilgisini verir. Bu seçenek işaretlendiğinde y değeri ekranda görülür.
18	<input type="checkbox"/>	Kuklanın yön bilgisini verir. Bu seçenek işaretlendiğinde yön bilgisi ekranda görülür.



1. Uygulama

İşlem adımlarına göre yeşil bayrak tıkladığında bahçede yürüyen panda kuklası uygulamasını yapınız.

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “bahçede yürüyen panda kuklası” olarak belirleyiniz.
- 3. Adım:** Görsel 1.28’de yer alan “camping” dekorunu sahneye ekleyiniz.



Görsel 1.28: Bahçede yürüyen panda kuklası

- 4. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” bloğunu kodlama alanına sürükleyiniz.
- 5. Adım:** Hareket blokunu kullanarak panda kuklasının konumunu x:-195, y:-123 olarak ayarlayınız.
- 6. Adım:** Program durduruluncaya kadar işlemin devam etmesi için Kontrol bloklarının altındaki “Sürekli tekrarla” komutunu ekleyip panda kuklasının hareketlerini belirleyecek komutları bu blok arasına yazınız.
- 7. Adım:** Görünüm blokundan “Sonraki kostüm” komutunu seçiniz.
- 8. Adım:** Hareket blokundan “10 adım git” komutunu seçiniz.
- 9. Adım:** Kontrol blokundan “1 saniye bekle” komutunu seçip, süreyi 0.16 olarak değiştiriniz.
- 10. Adım:** Hareket blokundan “Kenara geldiyse sektir” bloğunu ekleyiniz.
- 11. Adım:** Hareket blokundan “dönüş stilini sol-sağ yap” bloğunu ekleyerek panda kuklası sahnenin sonuna geldiğinde panda kuklasının geriye dönüş yapmasını sağlayınız.

Birinci uygulamanın kod blokları Görsel 1.29’da verilmiştir.



Görsel 1.29: Bahçede yürüyen panda kuklası uygulamasının kod blokları



Sıra Sizde

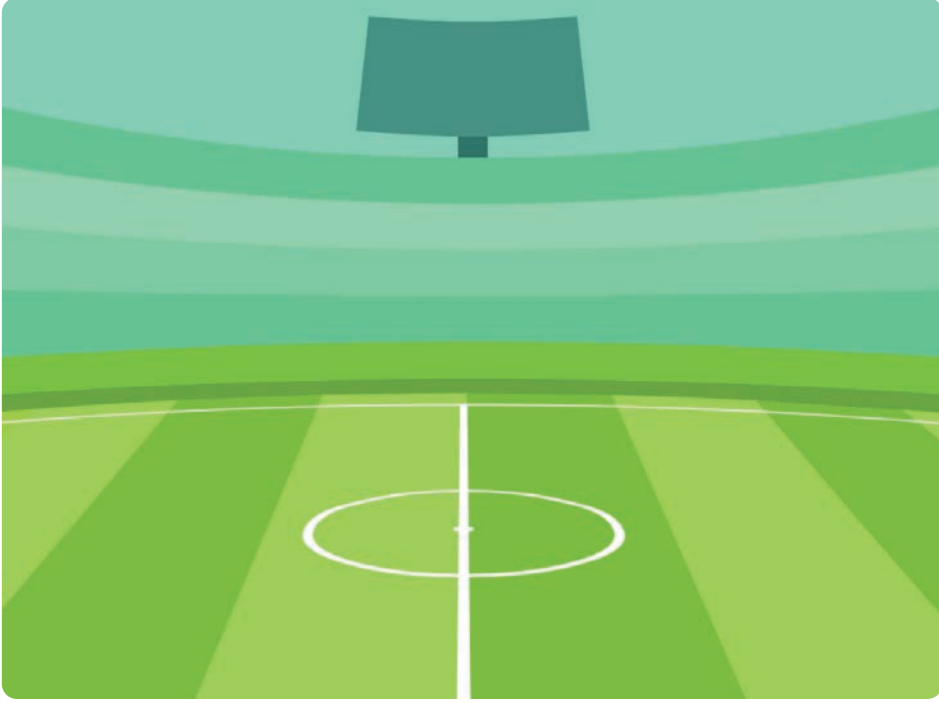
Aynı kukla ve dekoru kullanarak bahçenin etrafında yürüyen panda kuklası uygulamasını yapınız.



2. Uygulama

İşlem adımlarına göre yeşil bayrak tıkladığında futbol sahasında zıplayan top uygulamasını yapınız.

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “zıplayan top” olarak belirleyiniz.
- 3. Adım:** Görsel 1.30’da yer alan “Football Field-2” dekorunu sahneye ekleyiniz.



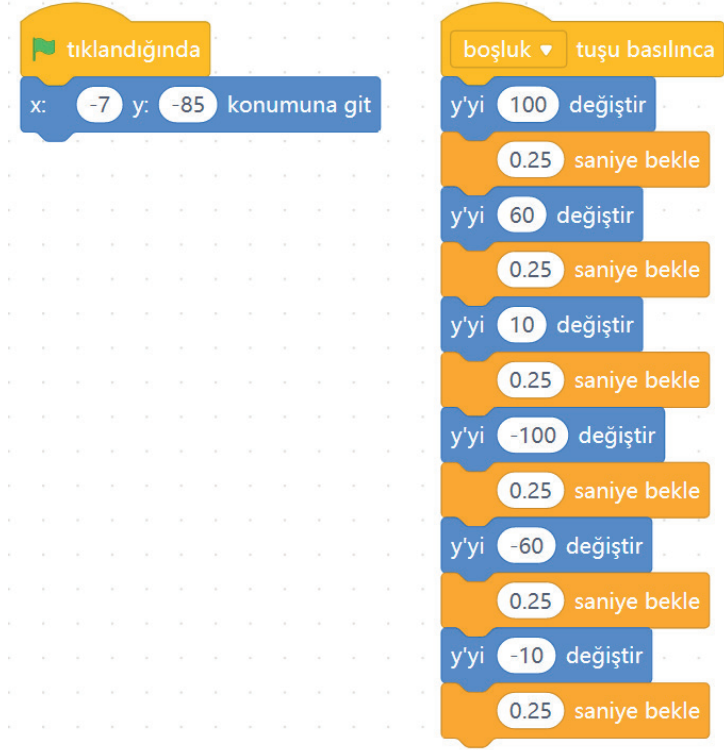
Görsel 1.30: Zıplayan top

- 4. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 5. Adım:** Git blokunu $x:-7$, $y:-85$ konumuna ekleyiniz.
- 6. Adım:** Topu zıplatmak için boşluk tuşuna basıp olay blokunu sürükleyiniz.
- 7. Adım:** Hareket bloklarından “y konumunu 100 değiştir” blokunu ekleyiniz.
- 8. Adım:** Kontrol bloklarından “0.25 saniye bekleyiniz” blokunu ekleyiniz.
- 9. Adım:** Hareket bloklarından “y konumunu 60 değiştir” blokunu ekleyiniz.
- 10. Adım:** Kontrol bloklarından “0.25 saniye bekleyiniz” blokunu ekleyiniz.
- 11. Adım:** Hareket bloklarından “y konumunu 10 değiştir” blokunu ekleyiniz.



- 12. Adım:** Kontrol bloklarından “0.25 saniye bekleyiniz” blokunu ekleyiniz.
- 13. Adım:** Hareket bloklarından “y konumunu -100 değiştir” blokunu ekleyiniz.
- 14. Adım:** Kontrol bloklarından “0.25 saniye bekleyiniz” blokunu ekleyiniz.
- 15. Adım:** Hareket bloklarından “y konumunu -60 değiştir” blokunu ekleyiniz.
- 16. Adım:** Kontrol bloklarından “0.25 saniye bekleyiniz” blokunu ekleyiniz.
- 17. Adım:** Hareket bloklarından “y konumunu -10 değiştir” blokunu ekleyiniz.

İkinci uygulamanın kod blokları Görsel 1.31’de verilmiştir.



Görsel 1.31: Zıplayan top uygulamasının kod blokları

Not

Uygulamada yerçekimi göz önüne alınarak yukarı zıplarken topun yavaşladığı, aşağı düşerken topun hızlandığı düşünülür.



Sıra Sizde

Araç kuklası ve uygun bir dekor kullanarak aracı yatay ekseninde hareket ettiren uygulamayı hazırlayınız.



1.2.4.2. Görünüm Blokları

Görünüm blokları, sahnenin ve sahnedeki kuklanın görünümüyle ilgili değişikliklerin yapılmasını sağlayan kod bloklarıdır. Tablo 1.2’de görünüm bloklarının tümü verilmiştir.

Tablo 1.2: Görünüm Blokları

Sıra	Blok	Açıklama
1		Kukla istenen süre boyunca “Merhaba!” yazan kutudaki metni konuşma balonu içinde gösterir.
2		Kukla “Merhaba!” yazan kutudaki metni konuşma balonu içinde gösterir.
3		Kukla istenen süre boyunca “Hmm...” yazan kutudaki metni konuşma balonu içinde gösterir.
4		Kukla “Hmm...” yazan kutudaki metni konuşma balonu içinde gösterir.
5		Kuklanın istenen kostüme geçmesini sağlar.
6		Kuklanın o anki kostümünden bir sonraki kostümüne geçmesini sağlar.
7		Dekoru, seçili dekor ile değiştirir.
8		Dekoru, bir sonraki dekor ile değiştirir.
9		Kuklanın boyutunu istenen değer kadar değiştirir.
10		Kuklanın boyutunu istenen % değerinde değiştirir.
11		Kuklanın rengi, balıkgözü, hızlı dön, piksellere böl, mozaik, parlaklık, hayalet efekti özelliklerini sayı yazan yerdeki değer kadar değiştirir.
12		Kuklanın rengi, balıkgözü, hızlı dön, piksellere böl, mozaik, parlaklık, hayalet efekti özelliklerini sayı yazan yerdeki değer yapar.
13		Kuklanın üzerinde uygulanmış bütün efektleri temizler.
14		Kuklayı gösterir ve gizler.
15		Kuklanın üst katmana çıkmasını sağlar. Kukla, bu kod bloku kullanıldığında diğer karakterlerin önünde görünür.
16		Kuklanın girilen sayı kadar katman öne veya alta gitmesini sağlar.
17		Kuklanın kostüm sayısını veya ismini ekranda gösterir.
18		Dekorun sayısını veya ismini ekranda gösterir.
19	<input type="checkbox"/>	Kuklanın büyüklüğünü ekranda gösterir.
20	<input type="checkbox"/>	Dekorun sayısını veya ismini ekranda gösterir.
21	<input type="checkbox"/>	Kuklanın büyüklüğünü ekranda gösterir.



1.2.4.3. Ses Blokları

Ses blokları, uygulamanın daha ilgi çekici hâle getirilmesini sağlayan bloklardır. Tablo 1.3'te ses blokları verilmiştir.

Tablo 1.3: Ses Blokları

Sıra	Blok	Açıklama
1		Seçili müzik çalmaya başlar ve ses çalma bittiğinde diğer bloka geçer.
2		Seçili müzik çalmaya başlar. Uygulama, müziğin bitmesini beklemeden diğer blokları çalıştırmaya devam eder.
3		Çalmaya devam eden tüm müzikler durur.
4		Çalan müziğe, girilen değer kadar değişiklik uygular.
5		Çalan müziği, girilen değer seviyesine getirir.
6		Devam eden tüm ses etkilerini temizler.
7		Ses yüksekliğini girilen değer kadar değiştirir.
8		Ses düzeyini girilen değere getirir.
9		Ses düzeyini ekranda gösterir.



3. Uygulama

İşlem adımlarına göre yeşil bayrak tıkladığında sahnede hareket eden harfler uygulamasını yapınız (Görsel 1.32).



Görsel 1.32: Hareketli harfler

1. Adım: Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.

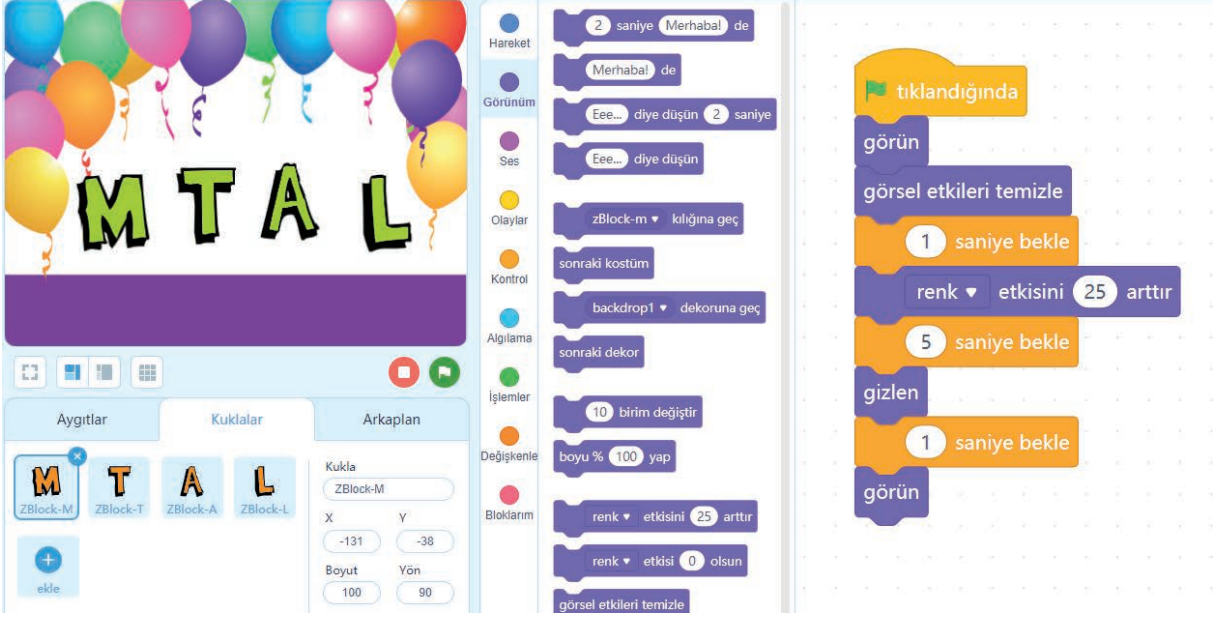
2. Adım: Projenizin adını "hareketli harfler" olarak belirleyiniz.



- 3. Adım:** Görsel 1.32'de yer alan "Party" dekorunu sahneye ekleyiniz.
- 4. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıkladığında" blokunu kodlama alanına sürükleyiniz.
- 5. Adım:** Mesleki ve Teknik Anadolu Lisesi kelimelerinin baş harfleri olan M, T, A ve L harflerinin kuklalarını ekleyiniz.
- 6. Adım:** Party dekorunu ekleyiniz.
- 7. Adım:** Party dekoru üzerindeki harfleri fare ile yan yana konumlandırınız.
- 8. Adım:** M harfi kuklası için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıkladığında" blokunu kodlama alanına sürükleyiniz.
- 9. Adım:** Görünüm blokundan "görün" blokunu ekleyiniz.
- 10. Adım:** Görünüm blokundan "görsel etkileri temizle" blokunu ekleyiniz.
- 11. Adım:** Kontrol bloklarından "1 saniye bekle" blokunu ekleyiniz.
- 12. Adım:** Görünüm bloklarından "renk etkisini %25 arttır" blokunu ekleyiniz.
- 13. Adım:** Kontrol bloklarından "5 saniye bekle" blokunu ekleyiniz.
- 14. Adım:** Görünüm blokundan "gizlen" blokunu ekleyiniz.
- 15. Adım:** Kontrol bloklarından "1 saniye bekle" blokunu ekleyiniz.
- 16. Adım:** Görünüm blokundan "görün" blokunu ekleyiniz.
- 17. Adım:** Ekranda oluşan kod blokunu diğer harf kuklalarının üzerine sürükleyip bırakınız.
- 18. Adım:** T, A ve L harfi kuklaları için ilk bekleme kontrolünü 2, 3 ve 4 saniye yapınız.
- 19. Adım:** T, A ve L harfi kuklaları için ikinci bekleme kontrolünü 6, 7 ve 8 saniye yapınız.
- 20. Adım:** Dekora tıklayarak Dekorlar sekmesine geliniz.
- 21. Adım:** Party dekoruna sağ tıklayarak "kopyasını çıkart" işlemi gerçekleştiriniz.
- 22. Adım:** Yeni oluşan "Party2" isimli dekorun balon renklerinden bazılarını "renk seçip doldur" düğmesi ile değiştiriniz.
- 23. Adım:** Dekor blokunu ve yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıkladığında" blokunu kodlama alanına sürükleyiniz.
- 24. Adım:** Kontrol bloklarından "sürekli tekrarla" blokunu ekleyiniz.
- 25. Adım:** Kontrol bloklarından "1 saniye bekle" blokunu ekleyiniz.
- 26. Adım:** Görünüm bloklarından "Party2" dekoruna geç blokunu ekleyiniz.
- 27. Adım:** Kontrol bloklarından "1 saniye bekle" blokunu ekleyiniz.
- 28. Adım:** Görünüm bloklarından "Party" dekoruna geç blokunu ekleyiniz.



Üçüncü uygulamanın kod blokları Görsel 1.33'te verilmiştir.



Görsel 1.33: Hareketli harfler kod blokları

Not

Sırt çantası kullanılarak benzer kod blokları bir kere yazılır. Bu kodlar gerektiğinde sürük-
le bırak yöntemiyle sırt çantasından kodlama alanına bırakılarak uygulamalar daha hızlı
geliştirilir.



Sıra Sizde

Zaman, renk ve ses parametrelerini değiştirerek aşağıda boş bırakılan yere üçüncü uygulamayı
kendi isminiz için hazırlayınız.



1.2.4.4. Olaylar Blokları

Bir uygulamanın çalışmaya başlayabilmesi için bir tetikleyiciye ihtiyaç vardır. Uygulamanın başlangıcı için genellikle sahenin sağ üst köşesindeki yeşil bayrak kullanılır. Benzer şekilde uygulamayı sonlandırmak için sabit görevi bulunan kırmızı düğme tıklanır ancak geliştirilen uygulama, yeşil bayrağın dışında farklı düğmeler veya eylemlerle kontrol edilebilir. Tablo 1.4’te olaylar blokları verilmiştir.

Tablo 1.4: Olaylar Blokları

Sıra	Blok	Açıklama
1		Yeşil bayrağa tıkladığında bu bloka eklenen diğer bloklar çalışmaya başlar.
2		Klavyeden basılacak herhangi bir tuşu tetikleyici olarak kullanan kod bloğudur.
3		Tetikleyici olarak kullanılan kod bloğudur.
4		Sahne dekorları istendiği zaman program içinde değiştirilebilir. Bu blok ile belirtilen bir dekor olduğunda çalışması istenen kodlar bu kod bloğunun altına eklenir.
5		Ses şiddeti, süreölçer ve video hareketi değerlerinin belirli bir seviyenin üzerinde olduğunda çalışması istenen kodlar bu kod bloğunun altına eklenir.
6		“haber1” haberi alındığında çalışması istenen bloklar bu kod bloğunun altına eklenir.
7		“haber1” haberi tüm kuklalar ve dekorlar için çalışma ortamında yayımlanır. Herhangi bir kukla, haber1 haberi geldiğinde yapacağı görev varsa görevini yerine getirir.
8		“haber1” haberini tüm kuklalara gönderir ve kuklanın kodu bitirmesini bekler.



4. Uygulama

İşlem adımlarına göre yeşil bayrak tıkladığında katmanlı yapıya göre yoldaki bir kişinin, aracın arkasından geçerek yürümesi uygulamasını yapınız (Görsel 1.34).



Görsel 1.34: Katmanlı yapı

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “katmanlı yapı” olarak belirleyiniz.



- 3. Adım:** Görsel 1.34'te yer alan "Street2" dekorunu sahneye ekleyiniz.
- 4. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıkladığında" blokunu kodlama alanına sürükleyiniz.
- 5. Adım:** dog7 kuklasını ekleyiniz ve boyutunu %50 yapınız.
- 6. Adım:** car1 kuklasını ekleyiniz ve boyutunu %200 yapınız.
- 7. Adım:** dog7 kuklası için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki "yeşil bayrağa tıkladığında" blokunu kodlama alanına sürükleyiniz.
- 8. Adım:** Hareket blokundan "x:-224, y:-80 konumuna git" blokunu ekleyiniz.
- 9. Adım:** Yürüyüşün sürekli olması için kontrol bloklarından "sürekli tekrarla" blokunu ekleyiniz.
- 10. Adım:** Hareket blokundan "10 adım git" blokunu ekleyiniz.
- 11. Adım:** Görünüm blokundan "sonraki kostüm" blokunu ekleyiniz.
- 12. Adım:** Sonraki kostüm blokundan "0.25 saniye bekle" blokunu ekleyiniz.
- 13. Adım:** Yürüyen kuklanın arabanın arkasından geçmesi için görünüm blokundan "geri katmana git" blokunu ekleyiniz.
- 14. Adım:** car1 kuklasını ekleyiniz.
- 15. Adım:** dog7 kuklası için yazdığınız kodları car1 kuklasına sürükleyip bırakınız.
- 16. Adım:** Konumuna git blokunu "x:-400, y:-90 konumuna git" şeklinde değiştiriniz.
- 17. Adım:** "10 adım git" blokunu "25 adım git" şeklinde değiştiriniz.
- 18. Adım:** Arabanın önden geçmesi için görünüm blokunu "ön katmana git" şeklinde değiştiriniz.

Dördüncü uygulamanın kod blokları Görsel 1.35'te verilmiştir.



Görsel 1.35: Katmanlı yapı dog7 kod blokları



Sıra Sizde

Bulut kuklaları ve gökyüzü dekoru ekleyerek katmanlı yapıya göre gökyüzünde bulutların hareketinin animasyonunu yapınız.



1.2.4.5. Kontrol Blokları

Kontrol kategorisindeki kod blokları ile programın akışı belirlenen şartlara göre yönlendirilebilir, tekrar eden görevler yerine getirilebilir, koşul sağlanıncaya kadar kod akışı bekletilebilir. Tablo 1.5'te kontrol blokları verilmiştir.

Tablo 1.5: Kontrol Blokları

Sıra	Blok	Açıklama
1		Bir kuklaya yazılan kod blokunun belirtilen süre kadar beklemesini sağlar. Bu sırada diğer kuklalara yazılan kodlar çalışmasını devam ettirir.
2		Yazılan kodun veya kod blokunun belirtilen sayı kadar tekrar etmesini sağlar. Bu blok, yazılımda çok önemli bir döngü bloktur.
3		Kod blokunun içine yazılan blokların sürekli olarak tekrar etmesini sağlar. Sonsuz döngü olarak bilinir. Uygulama durdurulmadan döngüden çıkamaz.
4		“eğer” kod bloku, bir şart ifadesinin yerine getirilip getirilmediğini sorgular. “eğer” şart ifadesi sağlanırsa bu kod blokunun içine yazılan kodları çalıştırır. Şart sağlanmazsa alttaki kod blokuna geçer.
5		“eğer” şartı sağlanırsa bu kod blokunun içine yazılan komutları çalıştırır. Şart sağlanmazsa “değilse” bloku içindeki kodları çalıştırır.
6		Bir şart sağlanıncaya kadar program akışını o kukla için bekletir.
7		Koşul sağlanıncaya kadar tekrar edilmesi gereken komutları çalıştırır. Bu döngü, while döngüsü olarak da bilinir.
8		Tüm kod bloklarının çalışmasını, sadece eklendiği komut dizisini veya eklendiği kuklanın diğer komut dizilerini durdurur.
9		Bir kuklanın ikizi oluşturulduğunda ikizin yapacağı görevleri belirtmek için kullanılan başlangıç bloktur.
10		Sahne de bulunan diğer kuklaların veya mevcut kuklanın ikizini yaratmak için kullanılan kod bloktur.
11		Oluşturulmuş bir ikiz kuklanın silinmesini sağlar.



5. Uygulama

İşlem adımlarına göre yeşil bayrağa tıkladığında koordinat düzleminin (90, -90) noktasından başlayarak sahneye 150x150 piksel ölçülerinde bir üçgen çizdiren uygulamayı hazırlayınız (Görsel 1.36).

1. Adım: Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.

2. Adım: Projenizin adını “üçgen çizimi” olarak belirleyiniz.

3. Adım: Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.

4. Adım: Hareket blokundan “x:90, y:-90 konumuna git” blokunu ekleyiniz.

5. Adım: Eklenti Ekle düğmesinden “kalem” eklentisini ekleyiniz.

6. Adım: Kalem blokundan “tümünü sil” blokunu ekleyiniz.

7. Adım: Kalem blokundan “kalem rengini turuncu yap” blokunu ekleyiniz.

8. Adım: Kalem blokundan “kalem kalınlığını 10 yap” blokunu ekleyiniz.

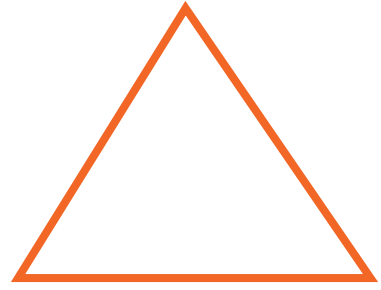
9. Adım: Kalem blokundan “kalemi bastır” blokunu ekleyiniz.

10. Adım: Kontrol blokundan “3 defa tekrarla” blokunu ekleyiniz.

11. Adım: Hareket blokundan “150 adım git” blokunu ekleyiniz.

12. Adım: Kontrol blokundan “1 saniye bekle” blokunu ekleyiniz.

13. Adım: Hareket blokundan “120 derece dön” blokunu ekleyiniz.



Görsel 1.36: Üçgen çizimi

Beşinci uygulamanın kod blokları Görsel 1.37’de verilmiştir.



Sıra Sizde

Yeşil bayrağa tıkladığı zaman kenar renkleri birbirinden farklı bir kare çiziniz.

Görsel 1.37: Üçgen çizimi kod blokları



6. Uygulama

İşlem adımlarına göre (0,0) noktasından başlayarak sahneye bir çember çizdiren uygulamayı hazırlayınız (Görsel 1.38).



Görsel 1.38: Çember çizimi

1. Adım: Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.

2. Adım: Projenizin adını “çember çizimi” olarak belirleyiniz.

3. Adım: Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.

4. Adım: Görünüm blokundan “gizlen” blokunu ekleyerek kuklanın görünmesini engelleyiniz.

5. Adım: Hareket blokundan “x:0, y:0 konumuna git” blokunu ekleyiniz.

6. Adım: Eklenti Ekle düğmesinden “kalem” eklentisini ekleyiniz.

7. Adım: Kalem blokundan “tümünü sil” blokunu ekleyiniz.

8. Adım: Kalem blokundan “kalem kalınlığını 9 yap” blokunu ekleyiniz.

9. Adım: Kalem blokundan “kalem rengini kırmızı yap” blokunu ekleyiniz.

10. Adım: Kalem blokundan “kalemi bastır” blokunu ekleyiniz.

11. Adım: Kontrol blokundan “360 defa tekrarla” blokunu ekleyiniz.

12. Adım: Hareket blokundan “1 adım git” blokunu ekleyiniz.

13. Adım: Hareket blokundan “1 derece dön” blokunu ekleyiniz.

Altıncı uygulamanın kod blokları Görsel 1.39’da verilmiştir.



Görsel 1.39: Çember çizimi kod blokları



Sıra Sizde

Yeşil bayrağa tıkladığı zaman kenar renkleri birbirinden farklı bir beşgen çiziniz.



1.2.4.6. Algılama Blokları

Algılama blokları ile fare imleci, klavye tuşları, ses şiddeti, video hareketi gibi birçok durumu kuklaların tespit etmesi sağlanır. Bu bloklar, kullanıcı ile etkileşimli uygulamalar geliştirmek için çok kullanışlıdır. Algılama blokları genellikle kontrol bloklarıyla birlikte kullanılır. Tablo 1.6'da algılama blokları verilmiştir.

Tablo 1.6: Algılama Blokları

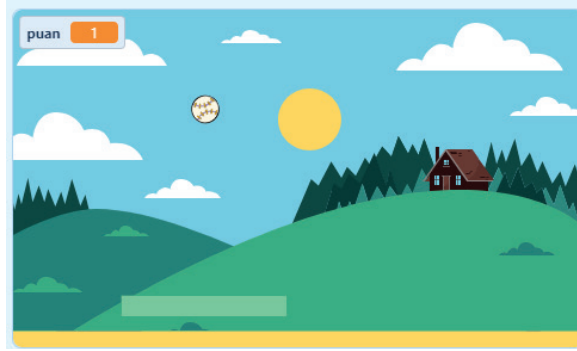
Sıra	Blok	Açıklama
1		Sahnedeki kuklaların fare imlecine, kenara veya kuklaya değip değmediğini sorgular. Sorgulanan kukla, seçilen nesnelere birine değiyorsa şart sağlanır.
2		Sahnedeki kuklanın seçilen bir renge değip değmediğini sorgular. Kolayca renk seçmek için renk seçme aracı kullanılır.
3		Seçilen renkteki bir nesnenin başka renkte bir nesneye değip değmediğini sorgular.
4		Kuklanın fareye veya diğer kuklalara mesafesini sorgular.
5		Sahne soru sorar ve kullanıcının verdiği cevabı yanıt değişkenine aktarır.
6		Üstteki komutta sorulan sorunun cevabını yanıt değişkenine atar.
7		Klavyeden hangi tuşa basıldığını sorgular.
8		Fareye basılı olma durumunu sorgular.
9		Farenin sahne üzerindeki x konumunu verir.
10		Farenin sahne üzerindeki y konumunu verir.
11		Sürüklenbilir modu değiştirir.
12		Mikrofondan alınan ses seviyesini verir.
13		Zamanlayıcı değerini verir.
14		Zamanlayıcıyı sıfırlar.
15		Kuklaların x konumu, y konumu, yönü, kılık numarası, kostüm ismi gibi bilgileri verir.
16		Bilgisayardaki sistem saatinden yıl, ay, gün, saat, dakika ve saniye bilgisini verir.
17		2000 yılından itibaren geçen gün sayısını verir.



7. Uygulama

İşlem adımlarına göre sahneye tek kişilik pinpon oyunu hazırlayınız.

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “pinpon oyunu” olarak belirleyiniz.
- 3. Adım:** “forest” dekorunu ekleyiniz.
- 4. Adım:** “Base Ball” kuklasını ekleyiniz.
- 5. Adım:** Kukla çiz bölümünden Görsel 1.40’taki gibi raket kuklası çiziniz ve kuklanın adını “raket” olarak veriniz.



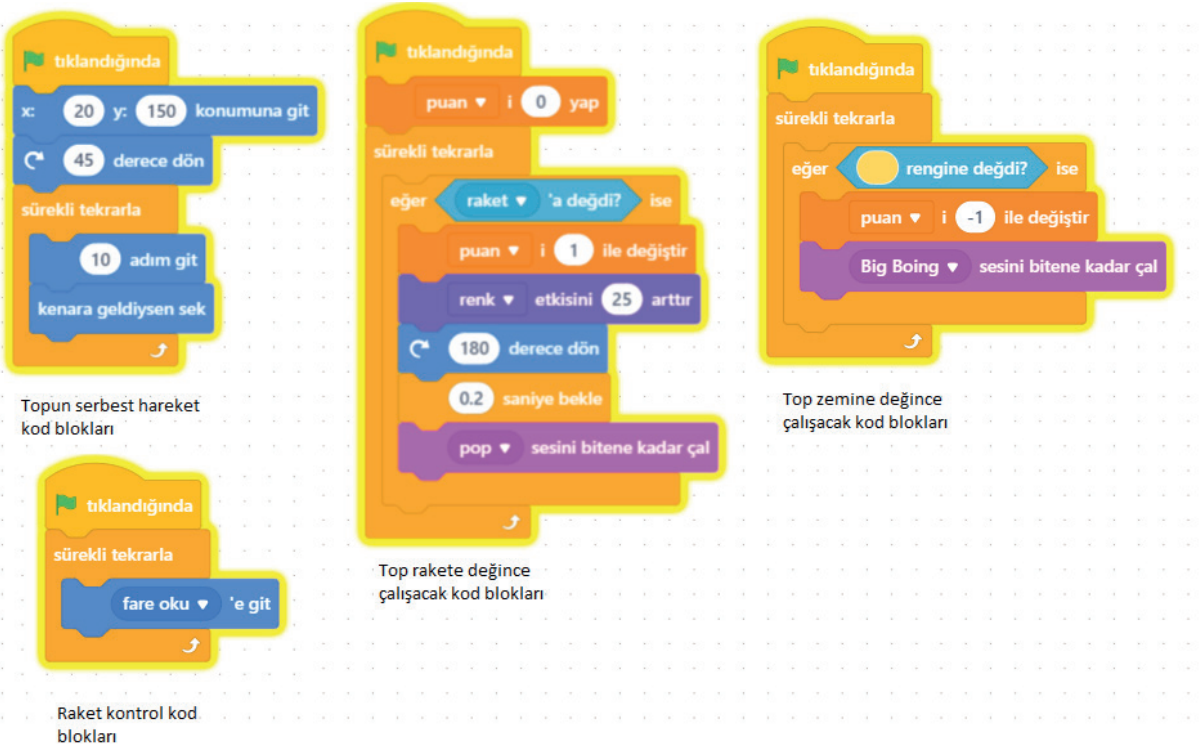
Görsel 1.40: Pinpon oyunu

- 6. Adım:** Eklediğiniz “forest” dekorunun altına Görsel 1.40’taki gibi sarı renkli zemin çiziniz.
- 7. Adım:** Raket isimli kukla için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 8. Adım:** Kontrol blokundan “sürekli tekrarla” blokunu kodlama alanına sürükleyiniz.
- 9. Adım:** Hareket blokundan “fare imlecine git” blokunu sürekli tekrarla bloku içine sürükleyiniz.
- 10. Adım:** BaseBall kuklasının hareketi için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 11. Adım:** Hareket blokundan “x:20, y:150 konumuna git” blokunu ekleyiniz.
- 12. Adım:** Hareket blokundan “45 derece dön” blokunu ekleyiniz.
- 13. Adım:** Kontrol blokundan “sürekli tekrarla” blokunu kodlama alanına sürükleyiniz.
- 14. Adım:** Sürekli tekrarla bloku içine hareket blokundan “10 adım git” blokunu ekleyiniz.
- 15. Adım:** Hareket blokundan “kenara geldiyse sek” blokunu ekleyiniz.
- 16. Adım:** Base Ball kuklasının rakete çarpma durumu için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 17. Adım:** Değişkenler blokundan “puan” isimli değişken oluşturunuz.
- 18. Adım:** Değişkenler blokundan “puan değişkenini 0 yap” blokunu ekleyiniz.
- 19. Adım:** Kontrol blokundan “sürekli tekrarla” blokunu kodlama alanına sürükleyiniz.
- 20. Adım:** Sürekli tekrarla bloku içine kontrol blokundan “eğer” blokunu kodlama alanına sürükleyiniz.



- 21. Adım:** “eğer” bloku şartı için “fare-imlecine geliyor mu” blokunu ekleyiniz ve fare imlecini “Raket” olarak değiştiriniz.
- 22. Adım:** Değişkenler blokundan “puanı 1 kadar değiştir” blokunu ekleyiniz.
- 23. Adım:** Görünüm blokundan “renk etkisini 25 değiştir” blokunu ekleyiniz.
- 24. Adım:** Hareket blokundan “180 derece dön” blokunu ekleyiniz.
- 25. Adım:** Kontrol blokundan “0.2 saniye bekle” blokunu ekleyiniz.
- 26. Adım:** Sesler sekmesinden “Pop” sesini ekleyiniz.
- 27. Adım:** Sesler blokundan “Ping Pong Hit bitene kadar çal” blokunu ekleyiniz.
- 28. Adım:** Base Ball kuklası zemine çarpma durumu için yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 29. Adım:** Kontrol blokundan “sürekli tekrarlar” blokunu kodlama alanına sürükleyiniz.
- 30. Adım:** Sürekli tekrarlar bloku içine kontrol blokundan “eğer” blokunu kodlama alanına sürükleyiniz.
- 31. Adım:** Algılama blokundan “rengine geliyor mu” blokunu ekleyiniz. Rengi “zemindeki sarı” olarak değiştiriniz.
- 32. Adım:** Değişkenler blokundan “puanı -1 kadar değiştir” blokunu ekleyiniz.
- 33. Adım:** Sesler sekmesinden “Big Boing” sesini ekleyiniz.
- 34. Adım:** Sesler blokundan “Big Boing bitene kadar çal” blokunu ekleyiniz.

Yedinci uygulamanın kod blokları Görsel 1.41’de verilmiştir.



Görsel 1.41: Pinpong oyunu kod blokları



Sıra Sizde

Puan 0'dan küçük olduğu anda oyunu durduran pinpon oyununu hazırlayınız.

1.2.4.7. Operatör Blokları

Operatörler kod blokları bölümünde matematiksel işlemler, mantıksal işlemler ve metin işlemleri için kullanılan kod blokları bulunur. Tablo 1.7'de operatör blokları verilmiştir.

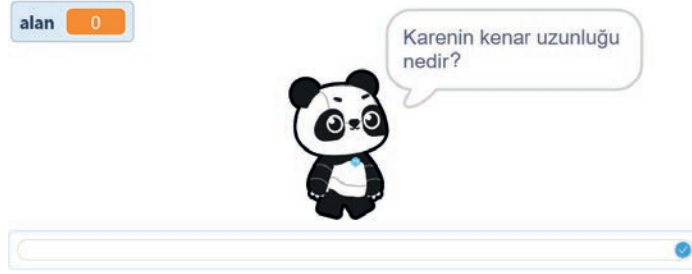
Tablo 1.7: Operatör Blokları

Sıra	Blok	Açıklama
1		Aritmetik işlem blokları sırasıyla iki sayıyı toplamak, çıkarmak, çarpmak ve bölmek için kullanılır.
2		Belirtilen iki sayı arasında rastgele bir sayı seçer.
3		İki sayıdan büyük olanı sorar. $60 > 50$ ifadesinde sonuç doğru döner. Harfleri de karşılaştırabilir. $c > b$ ifadesi doğru sonuç üretecektir.
4		İki sayıdan küçük olanı sorar. $50 < 60$ ifadesinde sonuç doğru döner. Harfleri de karşılaştırabilir. $b < c$ ifadesi doğru sonuç üretecektir.
5		İki sayının eşitliğini sorar. $50 = 50$ ifadesinde sonuç doğru döner. Harfleri de karşılaştırabilir. $c = c$ ifadesi doğru sonuç üretecektir.
6		"ve" operatörü, şart ifadelerinden ancak ikisi de doğru olduğunda doğru sonucu verir.
7		"veya" operatörü, şart ifadelerinden sadece biri doğru olduğunda doğru sonucu verir.
8		"değil" operatörü, doğru olan sonucu yanlış, yanlış olan sonucu doğruya çevirir.
9		İki tane metinsel ifadeyi birleştirir.
10		Girilen metin ifadesinin belirtilen sıradaki harfini verir.
11		Girilen metin ifadesinin toplam karakter uzunluğunu verir.
12		Girilen metin ifadesinin içinde istenen harfi arar.
13		Soldaki sayının sağdaki sayıya göre modunu alır. Örneğin $5 \text{ mod } 3$ işleminin sonucu 2'dir.
14		Girilen sayıyı en yakın tam sayıya yuvarlar.
15		Girilen sayının seçilen matematiksel fonksiyon sonucunu verir.



8. Uygulama

İşlem adımlarına göre bir kenar uzunluğu girilen karenin alanını hesaplayan uygulamayı hazırlayınız (Görsel 1.42).



Görsel 1.42: Karenin alan hesabı

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “karenin alan hesabı” olarak belirleyiniz.
- 3. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 4. Adım:** Değişkenler blokundan “alan” isimli değişken oluşturunuz.
- 5. Adım:** Değişkenler blokundan “alan değişkenini 0 yap” blokunu ekleyiniz.
- 6. Adım:** Algılama blokundan “karenin kenar uzunluğu diye sor ve bekle” blokunu ekleyiniz.
- 7. Adım:** “Alan değişkenini 0 yap” bölümüne operatörler blokundan “çarpım” blokunu ekleyiniz.
- 8. Adım:** Çarpım parametreleri için “yanıt ve yanıt” bloklarını algılama bloklarından ekleyiniz.
- 9. Adım:** Görünüm bloklarından “2 saniye boyunca Eee... de” blokunu ekleyiniz.
- 10. Adım:** Eee... yazan yere operatörler blokundan “muz ile elma birleştir” bloku ekleyiniz.
- 11. Adım:** Muz yazan yere “karenin alanı” yazınız.
- 12. Adım:** Elma yazan yere alan değişkenini sürükleyip bırakınız.



Görsel 1.43: Karenin alan hesabı kod blokları

Sekizinci uygulamanın kod blokları Görsel 1.43'te verilmiştir.



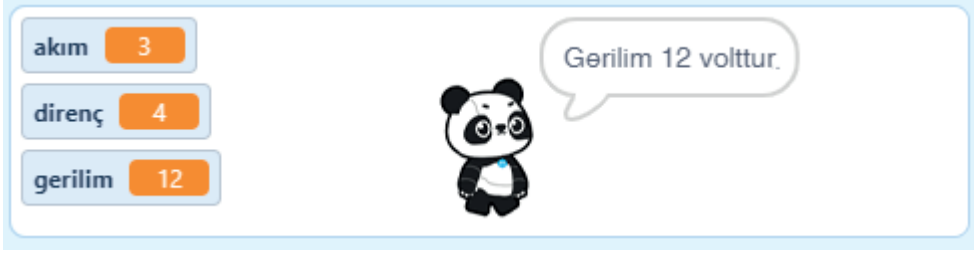
Sıra Sizde

Üçgenin alanını hesaplayan uygulamayı yapınız.



9. Uygulama

İşlem adımlarına göre bir iletkenin uçlarındaki gerilimi (Ohm Kanunu) hesaplayan uygulamayı hazırlayınız (Görsel 1.44).



Görsel 1.44: Ohm Kanunu ile ilgili uygulama

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “Ohm Kanunu” olarak belirleyiniz.
- 3. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 4. Adım:** Değişkenler blokundan “akım” isimli değişken oluşturunuz.
- 5. Adım:** Değişkenler blokundan “direnc” isimli değişken oluşturunuz.
- 6. Adım:** Değişkenler blokundan “gerilim” isimli değişken oluşturunuz.
- 7. Adım:** Algılama blokundan “Akım değerini giriniz? diye sor ve bekle” blokunu ekleyiniz.
- 8. Adım:** Akım değişkenine yanıt değerini algılama blokundan sürükleyip bırakınız.
- 9. Adım:** Algılama blokundan “Direnc değerini giriniz? diye sor ve bekle” blokunu ekleyiniz.
- 10. Adım:** Direnc değişkenine yanıt değerini algılama blokundan sürükleyip bırakınız.
- 11. Adım:** Çarpım operatörü parametrelerine akım ve direnc değişkenlerini sürükleyip bırakınız.
- 12. Adım:** Görünüm blokundan “Merhaba! de” blokunu ekleyiniz.
- 13. Adım:** “Merhaba!” yazısı yerine operatörler blokundan “muz ile elma birleştir” blokunu ekleyiniz.
- 14. Adım:** Elma yazan yere operatörler blokundan “muz ile elma birleştir” blokunu ekleyiniz.
- 15. Adım:** Muz yazısı yerine “Gerilim” yazınız.
- 16. Adım:** İçteki muz yazısı yerine gerilim değişkenini sürükleyip bırakınız.
- 17. Adım:** Elma yazısı yerine volt yazınız.



Dokuzuncu uygulamanın kod blokları Görsel 1.45'te verilmiştir.



Görsel 1.45: Ohm Kanunu ile ilgili uygulamanın kod blokları



Sıra Sizde

Güç hesabı yapan bir uygulama hazırlayınız.

Not

Güç hesabı, $watt = volt \times amper$ ile yapılır.

1.2.4.8. Değişken Blokları

Kodlamada kullanılan editörün izin verdiği veri tipleri çerçevesinde istenen verileri, bilgileri saklamaya yarayan yapılara **değişken** denir. Değişken tanımlanarak değişkenlere erişilebilir, değişkenler birbirinden ayrılabilir ve değişkenlerle ilgili diğer işlemler yapılabilir. Tablo 1.8'de değişken blokları verilmiştir.

Tablo 1.8: Değişken Blokları

Sıra	Blok	Açıklama
1		Bir Değişken Oluştur seçeneği, yeni değişken oluşturur. Değişkenin seçili kukla için veya tüm kuklalar için kullanılacağı değişken oluşturulurken seçilir.
2		Değişkeni ve değerini gösterir.
3		Değişkene girilen değeri atar.
4		Değişkenin değerini girilen değer kadar artırır. Değişkenin değerini azaltmak için – değer kullanılır.
5		Değişkeni sahnede gösterir.
6		Sahnede gösterilen değişkeni gizler.



1.2.4.9. Liste Blokları

Listeler, aynı veri çeşidinden değişkenlerin aynı isim altında tanımlanmasıdır. Listeler kullanılarak aynı isimle birden fazla değişkene erişilebilir ve değişkenlerle ilgili diğer işlemler yapılabilir. Tablo 1.9'da liste blokları verilmiştir.

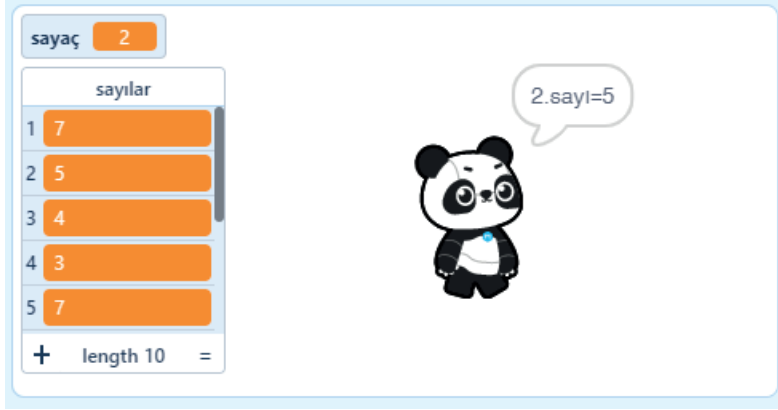
Tablo 1.9: Liste Blokları

Sıra	Blok	Açıklama
1		Bir Liste Oluştur seçeneği, yeni bir liste oluşturur.
2		Liste değişkeni bilgisini tutar. Liste Değişkeni kutusu işaretlendiğinde ekranda liste bilgisi görülür.
3		İstenen ifadeyi seçilen listeye ekler.
4		Seçilen listede istenen sıradaki nesneyi siler.
5		Seçilen listedeki tüm elemanları siler.
6		Girilen veriyi seçilen listenin istenen sırasına ekler.
7		Seçilen listenin sırası girilen elemanlarının yerini değiştirir.
8		Seçilen listenin istenen sıradaki elemanının alınmasını sağlar.
9		Seçilen listenin istenen elemanını almayı sağlar.
10		Seçilen listenin uzunluğunu verir.
11		Seçilen listenin içinde istenen elemanın varlığını sorar.
12		Seçilen listeyi ekranda gösterir.
13		Seçilen listeyi ekranda gizler.



10. Uygulama

İşlem adımlarına göre 10 tane sayıyı 1 ile 10 arasında rastgele seçip bu sayıları ikişer saniye aralıklarla sahneye yazdıran uygulamayı listeler yardımıyla hazırlayınız (Görsel 1.46).



Görsel 1.46: Liste uygulaması

- 1. Adım:** Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.
- 2. Adım:** Projenizin adını “liste ile sayıları yazdırma” olarak belirleyiniz.
- 3. Adım:** Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.
- 4. Adım:** Değişkenler blokundan “sayacı” isimli değişken oluşturunuz.
- 5. Adım:** Değişkenler blokundan “sayılar” isimli liste oluşturunuz.
- 6. Adım:** Değişkenler blokundan “sayılar” isimli listenin “hepsini sil” blokunu ekleyiniz.
- 7. Adım:** 10 sayı seçileceği için kontrol blokundan “10 defa tekrarla” blokunu ekleyiniz.
- 8. Adım:** Değişkenler blokundan “şeyi sayılara ekle” blokunu ekleyiniz.
- 9. Adım:** Operatörler blokundan “1 ile 10 arasında rastgele sayı seç” blokunu “şeyi” yazan parametreye sürükleyip bırakınız.
- 10. Adım:** Değişkenler blokundan “değişkenim değişkenini 0 yap” blokunu ekleyiniz. “değişkenim” parametresini “sayacı” olarak seçiniz.
- 11. Adım:** Kontrol blokundan “10 defa tekrarla” blokunu ekleyiniz.
- 12. Adım:** Değişkenler blokundan “sayacı 1 ile değiştir” blokunu ekleyiniz.
- 13. Adım:** Görünüm blokundan “2 saniye boyunca Merhaba! de” blokunu ekleyiniz.
- 14. Adım:** “Merhaba! de” yazısı yerine operatörler blokundan “muz ile elma birleştir” blokunu ekleyiniz.
- 15. Adım:** Muz yerine operatörler blokundan “muz ile elma birleştir” blokunu ekleyiniz.
- 16. Adım:** On dördüncü adımdaki elma yerine değişkenler blokundan “öge 1 için sayılar” blokunu ekleyiniz.

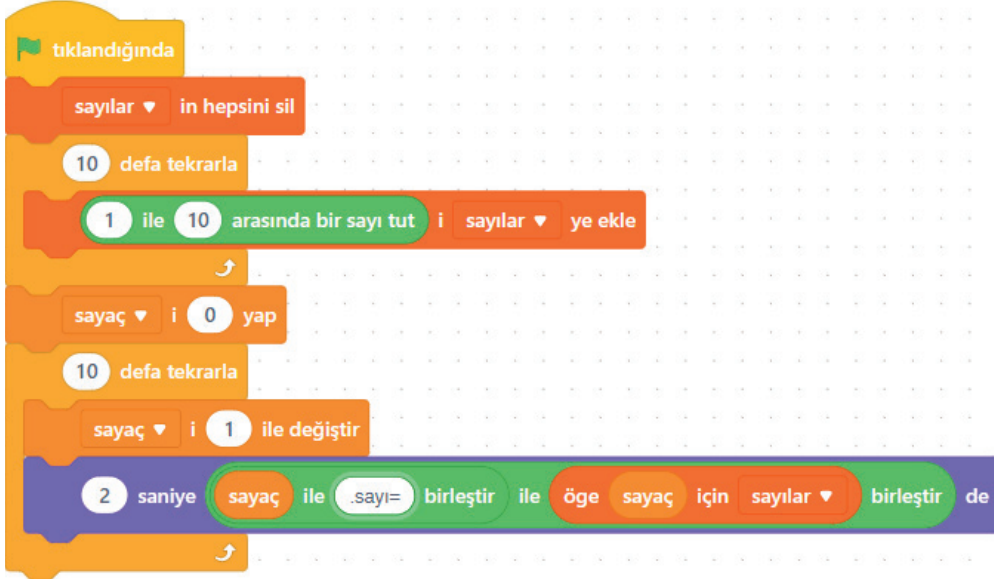


17. Adım: 1 yazan yere değişkenler blokundan “sayaç” değişkenini sürükleyip bırakınız.

18. Adım: On beşinci adımdaki muz yerine değişkenler blokundan “sayaç” değişkenini sürükleyip bırakınız.

19. Adım: On beşinci adımdaki elma yerine “.sayı=” yazınız.

Onuncu uygulamanın kod blokları Görsel 1.47’de verilmiştir.



Görsel 1.47: Liste uygulaması kod blokları



Sıra Sizde

Sayılar listesindeki en büyük sayıyı bulan uygulamayı hazırlayınız.



11. Uygulama

İşlem adımlarına göre çizgi izleyen robot uygulamasını hazırlayınız (Görsel 1.48).

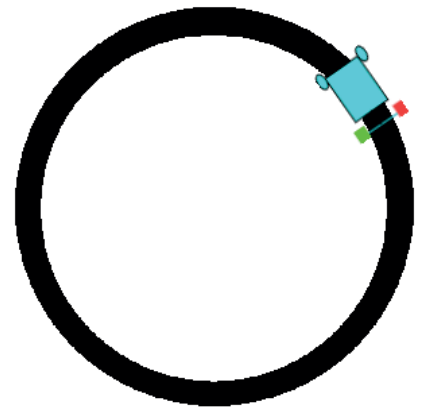
1. Adım: Blok tabanlı programlama editörünü açarak yeni bir proje oluşturunuz.

2. Adım: Projenizin adını “çizgi izleyen robot” olarak belirleyiniz.

3. Adım: Dekor çiz bölümünde yuvarlak bir pist çiziniz (Görsel 1.48).

4. Adım: Kukla çiz bölümünde çizgi izleyen robot çiziniz (Görsel 1.48).

5. Adım: Yeşil bayrağa tıkladığı zaman olay gerçekleşeceği için olaylar blokundaki “yeşil bayrağa tıkladığında” blokunu kodlama alanına sürükleyiniz.

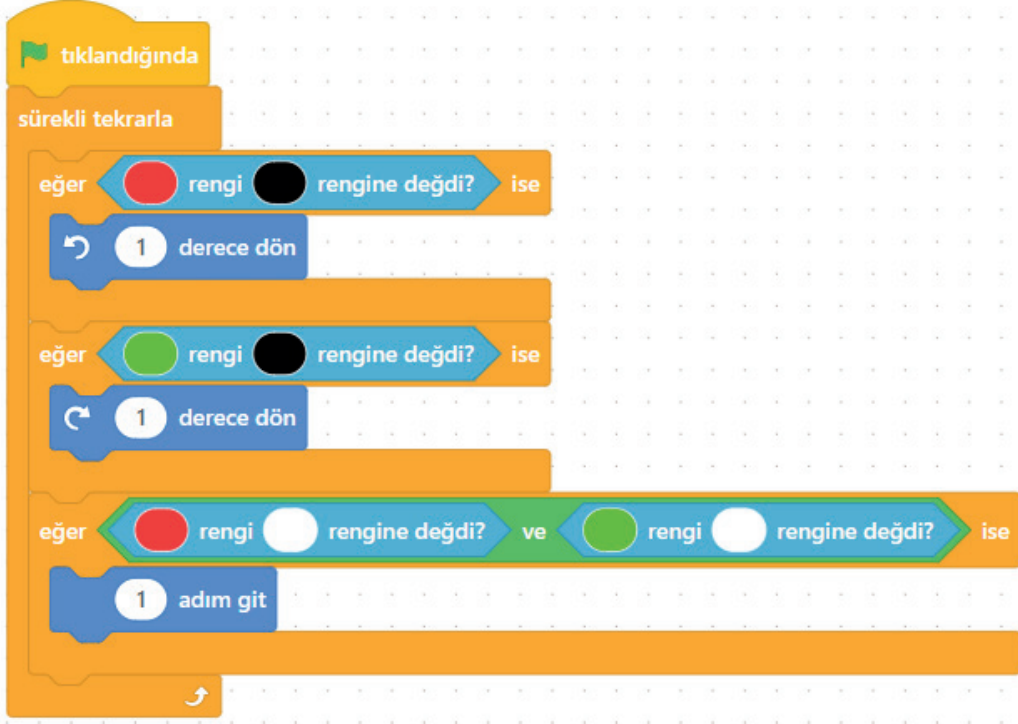


Görsel 1.48: Çizgi izleyen robot uygulaması



- 6. Adım:** Kontrol blokundan “sürekli tekrarlar” blokunu ekleyiniz.
- 7. Adım:** Kontrol blokundan “eğer ise” blokunu ekleyiniz.
- 8. Adım:** Algılama blokundan “renge rengine değişiyor mu” blokunu ekleyiniz.
- 9. Adım:** Sol sensör kontrolü için “kırmızı siyaha değişiyor mu” renklerini seçiniz.
- 10. Adım:** Hareket blokundan “1 derece sola dön” blokunu ekleyiniz.
- 11. Adım:** Kontrol blokundan “eğer ise” blokunu ekleyiniz.
- 12. Adım:** Algılama blokundan “renge rengine değişiyor mu” blokunu ekleyiniz.
- 13. Adım:** Sağ sensör kontrolü için “yeşil siyaha değişiyor mu” renklerini seçiniz.
- 14. Adım:** Hareket blokundan “1 derece sağa dön” blokunu ekleyiniz.
- 15. Adım:** Kontrol blokundan “eğer ise” blokunu ekleyiniz.
- 16. Adım:** Operatörler blokundan “ve” operatörü blokunu ekleyiniz.
- 17. Adım:** “ve” operatörüne “kırmızı beyaza değişiyor mu” ve “yeşil beyaza değişiyor mu” parametrelerini ekleyiniz.
- 18. Adım:** Hareket blokundan “1 adım git” blokunu ekleyiniz.

On birinci uygulamanın kod blokları Görsel 1.49’da verilmiştir.



Görsel 1.49: Çizgi izleyen robot uygulaması kod blokları



ÖLÇME VE DEĞERLENDİRME

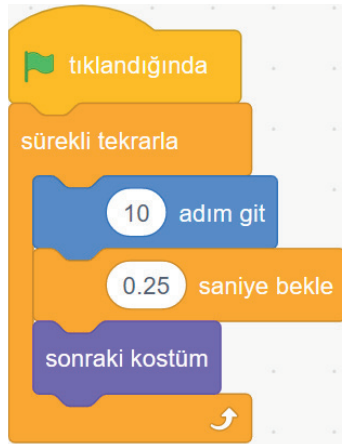
A) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

1. Aşağıdaki kod bloku seçeneklerdeki görevlerden hangisini gerçekleştirir?



- A) Yeşil bayrak tıkladığında başlangıç konumu olarak belirlenmiş x:185, y:115 konumuna gider. 9 saniye bekler ve x:185, y:125 konumuna gider.
- B) Yeşil bayrak tıkladığında başlangıç konumu olarak belirlenmiş x:-185, y:115 konumuna gider. 3 saniye bekler ve x:185, y:125 konumuna gider.
- C) Yeşil bayrak tıkladığında başlangıç konumu olarak belirlenmiş x:-185, y:-115 konumuna gider. 3 saniye bekler ve x:185, y:125 konumuna gider.
- D) Yeşil bayrak tıkladığında başlangıç konumu olarak belirlenmiş x:185, y:-115 konumuna gider. 9 saniye bekler ve x:185, y:125 konumuna gider.
- E) Yeşil bayrak tıkladığında başlangıç konumu olarak belirlenmiş x:-185, y:-115 konumuna gider. 3 saniye bekler ve x:185, y:-125 konumuna gider.

2. Panda kuklasına aşağıdaki kod bloku eklenirse panda kuklası seçeneklerden hangisinde yer alan hareketi yapar?



- A) Yeşil bayrak tıkladığında sürekli tekrarlayacak şekilde 10 adım gider, 0.25 saniye bekler, kostüm değiştirir.
- B) Yeşil bayrak tıkladığında belirlenmiş yönde sürekli tekrarlayacak şekilde 10 adım gider, 0.25 saniye bekler, kostüm değiştirir.
- C) Boşluk tuşuna basıldığında belirlenmiş yönde sürekli tekrarlayacak şekilde 10 adım gider, 0.25 saniye bekler, kostüm değiştirir.
- D) Boşluk tuşuna basıldığında sürekli tekrarlayacak şekilde 10 adım gider, 0.25 saniye bekler, kostüm değiştirir.
- E) Yeşil bayrak tıkladığında belirlenmiş yönde sürekli tekrarlayacak şekilde 10 adım gider, kostüm değiştirir.

3. Aşağıdaki kod bloku seçeneklerdeki hangi geometrik şekli çizer?

- A) Altıgen
- B) Beşgen
- C) Çember
- D) Kare
- E) Üçgen

```
when clicked, show message, move to x: -65, y: -62, erase everything, set pen color to orange, set pen thickness to 10, draw line, repeat 6 times: move 100 steps, wait 1 second, turn 60 degrees
```

4. Girilen sayının karesini alan uygulamayı yazmak için aşağıdaki kod bloku seçeneklerden hangisi ile tamamlanmalıdır?

- A) $\text{yanıt} * \text{yanıt}$ de
- B) $\text{yanıt} - \text{yanıt}$ de
- C) $\text{yanıt} / \text{yanıt}$ de
- D) $\text{yanıt} + \text{yanıt}$ de
- E) $\text{yanıt} \bmod \text{yanıt}$ de

5. Panda kuklası 0.16 saniyede 10 adım hareket ederek kenara geldiyse panda kuklasının düz bir şekilde geri dönme hareketini sürekli tekrarlayacak aşağıdaki kod blokları hangi sırayla tamamlanmalıdır?

- A) 2-3-4-5-6-7-1
- B) 2-7-4-1-6-3-5
- C) 3-4-1-2-6-7-5
- D) 5-4-1-2-6-7-3
- E) 7-4-3-5-1-2-6

6. Aşağıdaki kod bloklarında seçeneklerden hangisi “?” işaretli yere geldiği zaman 10 tane sayıyı 1 ile 10 arasında rastgele seçip bu sayıları ikişer saniye aralıklarla ekrana yazdırır?

```
when clicked
  clear numbers
  repeat 10 times
    pick a random number between 1 and 10
    add to numbers
  set counter to 0
  repeat 10 times
    ?
  wait 2 seconds
  append counter and number to text
```

- A) 1 saniye bekle
- B) 1 ile 10 arasında bir sayı tut
- C) sayaç i 0 yap
- D) şey i sayılar ye ekle
- E) sayaç i 1 ile değiştir

7. Aşağıdakilerden hangisi mBlock programında verilen bir işlemin dokuz kez tekrar etmesi istendiğinde kullanılacak kod bloğudur?

- A) 9 defa tekrarla
- B) 9 saniye bekle
- C) eğer tekrar = 9 ise
- D) tekrar i 9 yap
- E) tekrar i 9 ile değiştir

8. Aşağıdakilerden hangisi yazılım için yapılmış en doğru tanımdır?

- A) Elektronik bileşenlerden oluşan donanım sistemine, belirli bir işi yapması için yazılmış kodlara yazılım denir.
- B) Kullanıcı işlerini kolaylaştıran her türlü programa yazılım denir.
- C) Elektronik bileşenlerden oluşan donanım sistemine yazılım denir.
- D) Sistem uygulamalarını gerçekleştiren kodlara yazılım denir.
- E) Bilgisayarları çalıştırmak ve belirli görevleri yürütmek için kullanılan bir talimattır.

9. I. Oyun

II. Animasyon

III. Çizim

IV. Algılama

mBlock programı ile yukarıdakilerden hangisi veya hangileri yapılabilir?

- A) Yalnız I
- B) Yalnız IV
- C) I ve II
- D) I, II, III, IV
- E) II, III

10. Aşağıdakilerden hangisi mBlock kullanımını artırmıştır?

- A) Değişken tanımlamanın kolay olması
- B) Sürükle bırak yöntemiyle kodlamanın çok kolay geliştirilmesi
- C) En iyi oyun geliştirme platformu olması
- D) Çevrimiçi veya çevrimdışı kullanılabilmesi
- E) Çevrimiçi paylaşım özelliği sayesinde sosyalleşmeye imkân sağlanması

11. Aşağıdakilerden hangisi panda kuklasının kontrol edilmesi istendiğinde pandanın hareketini sağlamak için kullanılan bloklardır?

- A) Hareket blokları
- B) Ses tanıma blokları
- C) Renk algılama blokları
- D) Işık sensörü blokları
- E) Sıcaklık algılama blokları




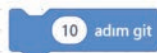

12. Aşağıdakilerden hangisi panda kuklasının kontrol edilmesi istendiğinde pandanın sesle tepki vermesini sağlamak için kullanılan bloklardır?

- A) Hareket blokları
- B) Ses tanıma blokları
- C) Işık sensörü blokları
- D) Renk algılama blokları
- E) Ultrasonik sensör blokları

13. Aynı kod satırlarının defalarca yazılması yerine aşağıdakilerden hangi kod bloku kullanılarak uygulamadaki kodlar daha sade hazırlanır?

- A) Hareket blokları
- B) Değişkenler
- C) Bloklarım
- D) Algılama blokları
- E) İşlemler blokları

14. Aşağıdaki kod bloklarından hangisi ile kukla 10 birim sola yer değiştirir?

- A)  y'yi -10 değiştir
- B)  x'i -10 değiştir
- C)  10 yönüne dön
- D)  10 adım git
- E)  10 derece dön

15. mBlock'ta kukla görünüm blokları aşağıdaki hangi amaca hizmet eder?

- A) Koddaki hataları düzeltir.
- B) Kodu daha estetik hâle getirir.
- C) Blokları daha iyi organize eder.
- D) Kuklanın hareketlerini programlar.
- E) Kuklanın görünümünü değiştirir.

16. Aşağıdaki hangi durumda kontrol kukla bloklarının kullanılması en uygun olur?

- A) Yalnızca bir kukla iletişim kutusu oluşturulmak istendiğinde
- B) Kuklanın hareketlerinin programı yazılırken
- C) Yalnızca kuklanın resimleri düzenlenirken
- D) Kuklanın etrafındaki engelleri algılayıp tepki vermesi istendiğinde
- E) Yalnızca kuklanın sesleri kontrol edilmek istendiğinde

17. Aşağıdakilerden hangisi bir işlemler kukla bloku değildir?

- A) Topla
- B) Çıkar
- C) Yön değiştir
- D) Çarp
- E) Böl

18. Aşağıdakilerden hangisi kukla bloklarından biri değildir?

- A) Hareket et
- B) Döngü
- C) Yön değiştir
- D) Bekle
- E) Ses çal

19. Kukla blokları kullanılarak aşağıdaki hangi projeler yapılabilir?

- A) Sadece resim galerisi oluşturulur.
- B) Sadece müzik aleti yapılır.
- C) Sadece kuklanın kostümleri değiştirilir.
- D) Robotik projelerden interaktif hikâyelere kadar geniş bir yelpazede kullanılır.
- E) Yalnızca kuklanın hareketleri kaydedilir.

20. Aşağıdaki hangi durumda algılama kukla bloklarının kullanılması en uygun olur?

- A) Yalnızca kukla hareketlerinin programlanması istendiğinde
- B) Kuklanın etrafındaki engelleri algılayıp tepki vermesi istendiğinde
- C) Yalnızca kukla seslerinin kontrol edilmesi istendiğinde
- D) Yalnızca bir kukla iletişim kutusu oluşturulması istendiğinde
- E) Kuklanın kostümlerini değiştirmek için herhangi bir zamanda

2. ÖĞRENME BİRİMİ

ROBOTİK PROGRAMLAMA PLATFORMLARI



KONULAR

- 2.1. ROBOTİK PROGRAMLAMA PLATFORMU YAPISI VE ÇEŞİTLERİ
- 2.2. ROBOTİK PROGRAMLAMA PLATFORMU

ANAHTAR KELİMELER

- IDE
- MEGA
- Mikrodenetleyici
- NANO
- UNO
- Uygulama kartı
- Port

NELER ÖĞRENECEKSİNİZ?

- Mikrodenetleyicinin fiziksel yapısı
- Mikrodenetleyici uygulama kartı bölümleri
- Mikrodenetleyici uygulama kartı çeşitleri
- IDE platformunu bilgisayara indirme ve kurma
- IDE platformu ile mikrodenetleyici uygulama kartı arasında bağlantı sağlama



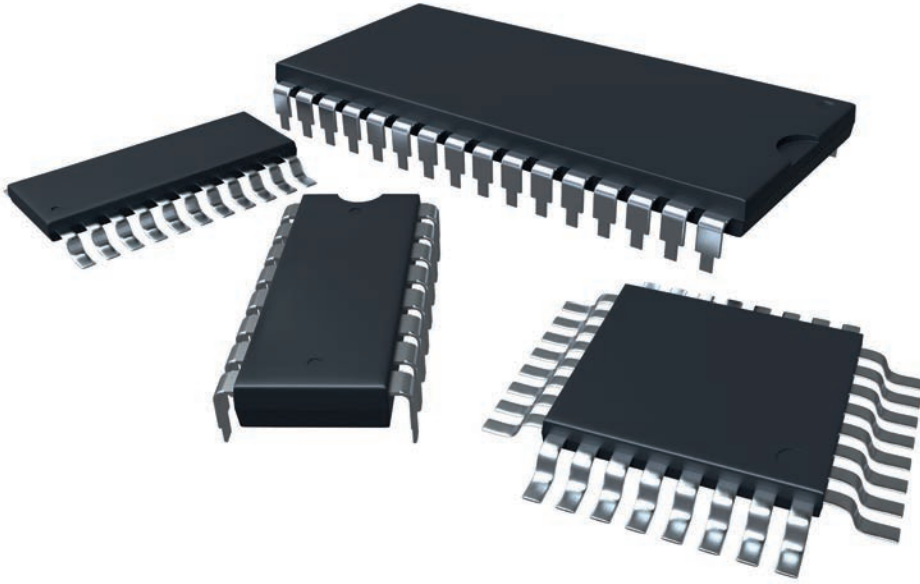


HAZIRLIK ÇALIŞMALARI

1. Elektronik parça ve sensörlerden oluşan tıbbi cihazlar istenen işlevi nasıl yerine getirir? Bu konudaki düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.
2. Açık kaynak kodlu yazılımların avantajları ve dezavantajları nelerdir?

2.1. ROBOTİK PROGRAMLAMA PLATFORMU YAPISI VE ÇEŞİTLERİ

Mikrodenetleyiciler, günlük yaşantıdaki veya sağlık sektöründeki cihazlarda yaygın olarak kullanılır. Mikrodenetleyicilerin temel işlevi; girişlerine bağlanan anahtar, sensör vb. elemanlar sayesinde gerekli ölçümleri almak, içinde bulunan yazılımla aldığı bu değerleri mantıksal işlemlere tabi tutmak ve çıkışlarına bağlanan elemanların çalışmasını kontrol etmektir. Mikrodenetleyiciler, üretici firmaların entegre üzerine yazdığı 16F824, ATmega328 gibi isimlerle anılır. Görsel 2.1'de çeşitli mikrodenetleyiciler görülür.



Görsel 2.1: Çeşitli mikrodenetleyici entegreleri

2.1.1. Mikrodenetleyicinin Fiziksel Yapısı

Mikrodenetleyiciler; içinde merkezî işlem birimi, hafıza, giriş/çıkış portlarının bulunduğu mikro bilgisayarlardır. Bazı mikrodenetleyicilerde analog/dijital (A/D) ve dijital/analog (D/A) dönüştürücüler de bulunur (Görsel 2.2).



Görsel 2.2: Mikrodenetleyicinin iç yapısı



MİB: Merkezî işlem birimidir. Bu birim, ilgili programları çalıştırmayı ve diğer birimler arasında haberleşmeyi sağlar.

RAM: Rastgele erişimli bellektir (Random Access Memory). Bu birimde geçici bilgiler tutulur.

ROM: Sadece okunabilen bellektir (Read Only Memory). Sistemin çalışması için gerekli programlar bu birimde saklanır.

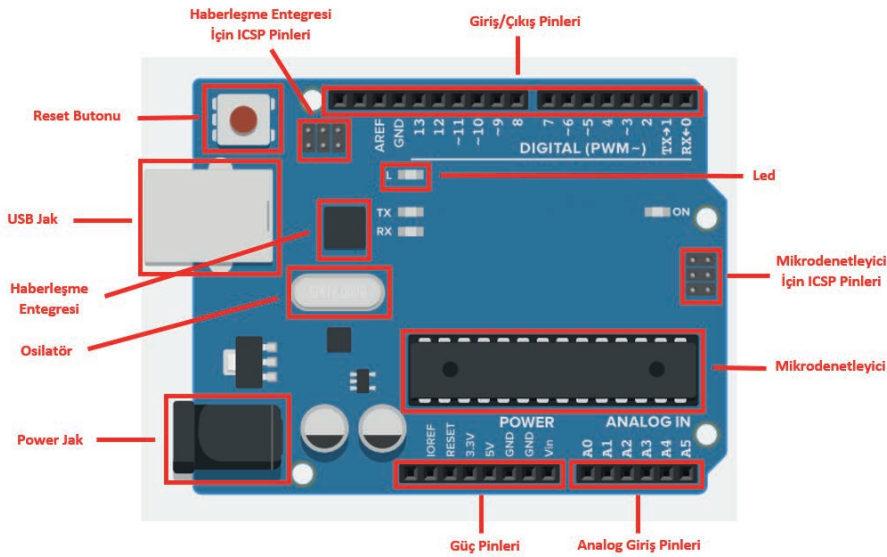
Giriş/Çıkış Portları: Giriş birimi, sensörler veya anahtarlardan gelen bilgilerin alındığı kısımdır. Çıkış birimi ise mikrodenetleyicinin yazılan programa göre çalışmasını kontrol ettiği elemanların bağlandığı bölümdür.

A/D Dönüştürücüler: Analog sinyallerin dijital sinyallere dönüştürüldüğü bölümdür.

D/A Dönüştürücüler: Dijital sinyallerin analog sinyallere dönüştürüldüğü bölümdür.

2.1.2. Mikrodenetleyici Uygulama Kartı

Mikrodenetleyiciler uygulamada tek başına kullanılmaz. Mikrodenetleyicinin çalışabilmesi için osilatör devresi, çevre birimlerinin bağlantıları, bilgisayarla haberleşmesi ve programlama için gerekli arabirimlerin olması gerekir. Bu arabirimlerin ve mikrodenetleyicinin bulunduğu tümleşik yapıya uygulama kartı denir. Görsel 2.3'te bir mikrodenetleyici uygulama kartı ve bu uygulama kartının bölümleri görülür.



Görsel 2.3: Mikrodenetleyici uygulama kartı

Reset Butonu: Mikrodenetleyicinin içinde işleyen programı durdurup yeniden başlatır.

USB Jak: Uygulama kartı, USB kablo ile bilgisayara bağlanarak mikrodenetleyici programlanabilir.

Haberleşme Entegresi: USB portu ile mikrodenetleyicinin haberleşmesini sağlar.

Osilatör: Mikrodenetleyicinin çalışabilmesi için gerekli olan saat pulsini üretir.

Power Jak: Uygulama kartı, haricî kaynaktan beslenecekse bu jak kullanılabilir ve 7-12 V doğru gerilimle besleme yapılabilir. Jakın merkezi pozitif olmalıdır.

IOREF: Mikrodenetleyicinin referans voltaj değerini sağlar.

RESET: Mikrodenetleyiciyi haricî olarak resetlemek için kullanılır.



3.3 V: 3.3 V maksimumu 50 mA doğru gerilim çıkış verir.

5 V: 5 V DC gerilim çıkış verir.

GND: Doğru gerilimle çalışan devrelerde eksi (-) ucu ifade eder.

VIN: Uygulama kartını beslemek için bu pin kullanılabilir. 7-12 V DC gerilimle besleme yapılabilir.

Analog Girişler: Analog sinyal üreten transduserlerin bağlandığı pinlerdir.

ICSP: Mikrodenetleyici ve haberleşme entegresini programlamak için kullanılacak pinlerdir.

Giriş/Çıkış Pinleri: Tanımlanmış giriş veya çıkış elemanların bağlantılarının yapıldığı kısımdır.

AREF: Analog girişler için referans gerilimidir.



Sıra Sizde

Mikrodenetleyici uygulama kartı üzerindeki PWM çıkış pinlerinin kullanımı ve bu pinlerin avantajları ile ilgili bir sunu hazırlayınız.

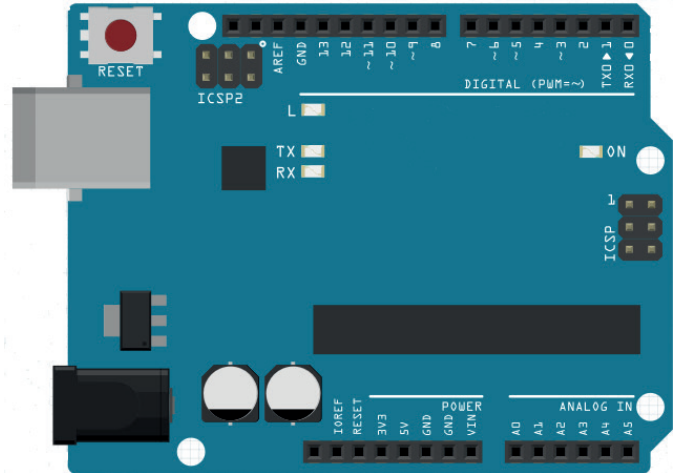
2.1.3. Mikrodenetleyici Uygulama Kartı Çeşitleri

Mikrodenetleyici uygulama kartlarının çok fazla çeşidi mevcuttur. Projelerde kartın boyutları, mikrodenetleyicinin bellek boyutu, giriş/çıkış sayısı, işlem hızı gibi özellikler dikkate alınarak seçim yapılır.

Yaygın olarak kullanılan mikrodenetleyici uygulama kartı çeşitleri ve teknik özellikleri şunlardır:

Mikrodenetleyici UNO Uygulama Kartının Teknik Özellikleri (Görsel 2.4)

- **Mikrodenetleyici:** ATmega328
- **Çalışma Gerilimi:** +5 V DC
- **Besleme Gerilimi:** 7-12 V DC
- **Dijital Giriş/Çıkış Pinleri:** 14 adettir (0-13) ve bunların 6 tanesi (3, 5, 6, 9, 10 ve 11 No.lu) PWM çıkışını destekler.
- **Analog Giriş Pinleri:** 6 adet (A0-A5)
- **Flash Hafıza:** 32 KB
- **SRAM:** 2 KB
- **EEPROM:** 1 KB
- **Saat Frekansı:** 16 MHz

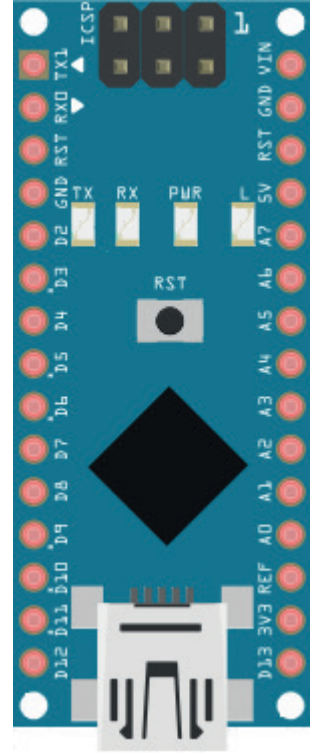


Görsel 2.4: Mikrodenetleyici UNO uygulama kartı



Mikrodenetleyici NANO Uygulama Kartının Teknik Özellikleri (Görsel 2.5)

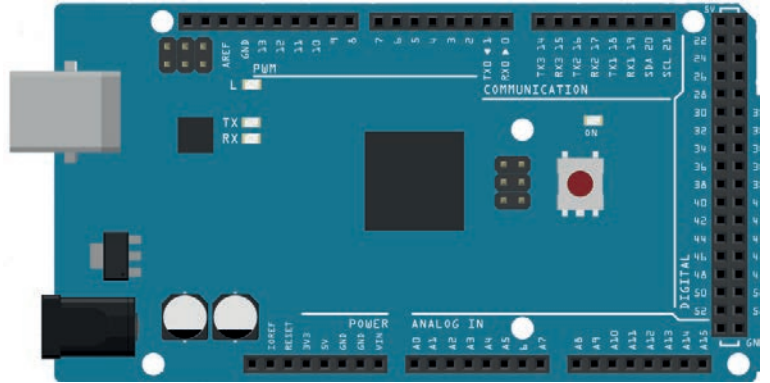
- Mikrodenetleyici: ATmega328
- Çalışma Gerilimi: +5 V DC
- Besleme Gerilimi: 7-12 V DC
- Dijital Giriş/Çıkış Pinleri: 14 adettir (0-13) ve 6 tanesi (3, 5, 6, 9, 10 ve 11 No.lu) PWM çıkışını destekler.
- Analog Giriş Pinleri: 8 adet (A0-A7)
- Flash Hafıza: 32 KB
- SRAM: 2 KB
- EEPROM: 1 KB
- Saat Frekansı: 16 MHz



Görsel 2.5: Mikrodenetleyici NANO uygulama kartı

Mikrodenetleyici MEGA Uygulama Kartının Teknik Özellikleri (Görsel 2.6)

- Mikrodenetleyici: ATmega 2560
- Çalışma Gerilimi: +5 V DC
- Besleme Gerilimi: 7-12 V DC
- Dijital Giriş/Çıkış Pinleri: 54 adettir (0-53) ve 15 tanesi (2-13 ve 44-46 No.lu) PWM çıkışını destekler.
- Analog Giriş Pinleri: 16 adet (A0-A15)
- Flash Hafıza: 256 KB
- SRAM: 8 KB
- EEPROM: 4 KB
- Saat Frekansı: 16 MHz



Görsel 2.6: Mikrodenetleyici MEGA uygulama kartı



Sıra Sizde

Mikrodenetleyici uygulama kartı seçiminde nelere dikkat edilmesi gerektiğini anlatan bir sunu hazırlayınız.



2.2. ROBOTİK PROGRAMLAMA PLATFORMU

Mikrodenetleyici uygulama kartının programlanması için geliştirilmiş platformlar vardır. Bunlardan biri de IDE platformudur. Tümlşik Geliştirme Ortamı da denilen IDE, İngilizce Integrated Development Environment kelimelerinin kısaltmasıdır.

2.2.1. İşletim Sistemine Uygun IDE Platformunu Bilgisayara İndirmek

Program kodlarının yazılıp derlenmesi ve mikrodenetleyici uygulama kartına yüklenebilmesi için kullanılan işletim sistemine uygun IDE platformu bilgisayara indirilir.

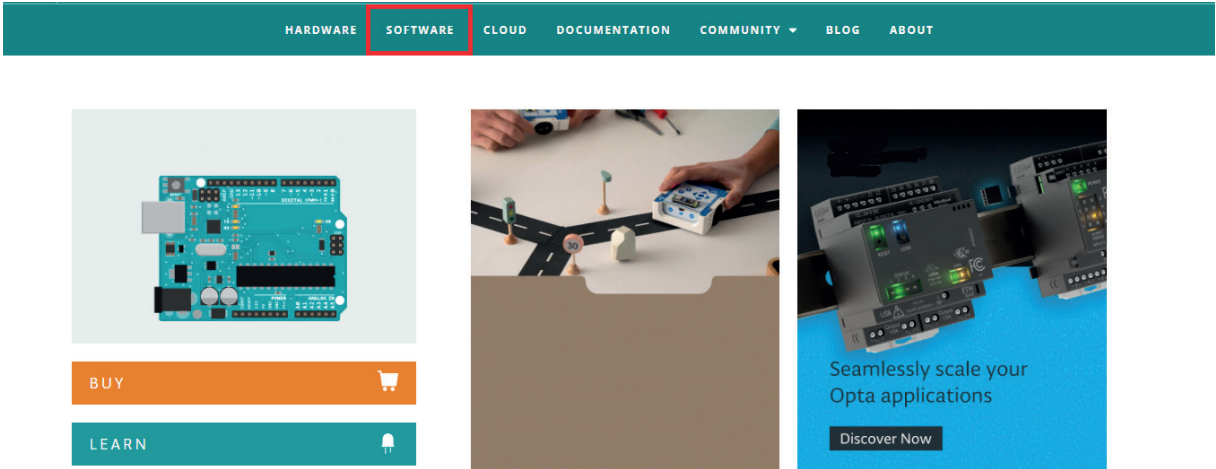


1. Uygulama

İşlem adımlarına göre IDE platformunu bilgisayara indiriniz.

1. Adım: İnternet tarayıcısının adres çubuğuna mikrodenetleyici uygulama kartının resmî internet adresini (<https://www.arduino.cc>) yazınız.

2. Adım: Açılan sayfada SOFTWARE menüsüne tıklayınız (Görsel 2.7).



Görsel 2.7: Mikrodenetleyici uygulama kartı resmî internet sayfası

3. Adım: Görsel 2.8'de verilen seçeneklerden kullanılan işletim sistemine uygun olanını seçiniz.



Görsel 2.8: İşletim sistemine uygun seçim yapılması



4. Adım: Görsel 2.9'da IDE platformu indirme sayfası görülür. Bu platform, açık kaynak kodludur. Bu nedenle platformun geliştirilmesi için bağış önerisi görülür. Bu öneriyi değerlendiriniz veya bağış yapmadan IDE platformunu indirmek için **JUST DOWNLOAD** seçeneğini seçiniz.



Görsel 2.9: IDE platformu indirme ekranı

5. Adım: Görsel 2.10'da sayfanın sağ üst köşesinde kurulum dosyasının indirildiğini gözlemleyiniz.



Görsel 2.10: Kurulum dosyasının indirilmesi



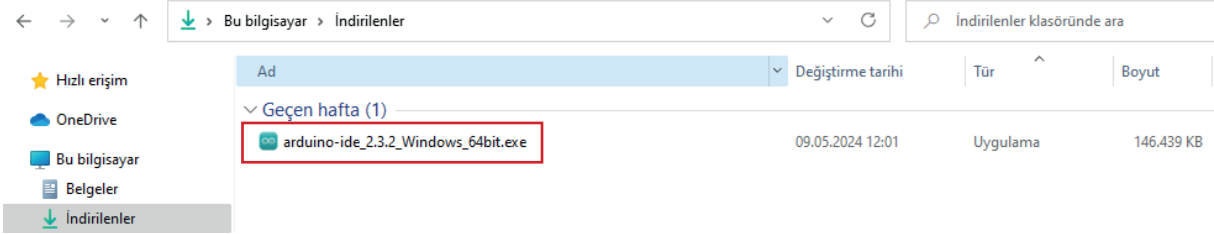
2.2.2. IDE Platformunu Bilgisayara Kurmak

Bilgisayara indirilen IDE platformunun kurulumu yapılmalıdır.



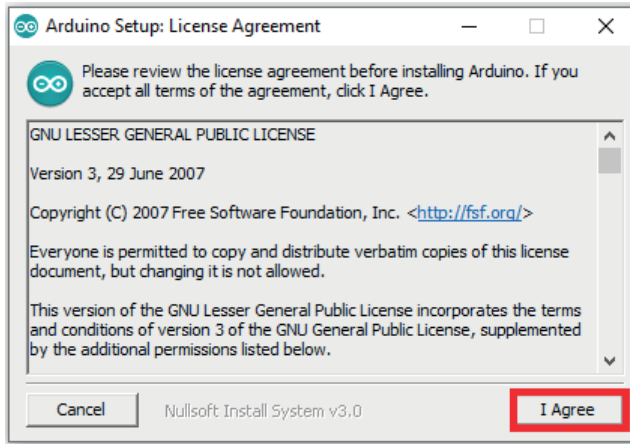
2. Uygulama

1. Adım: Bilgisayara indirilen kurulum dosyasının üzerine farenin sol tuşu ile çift tıklayınız (Görsel 2.11).



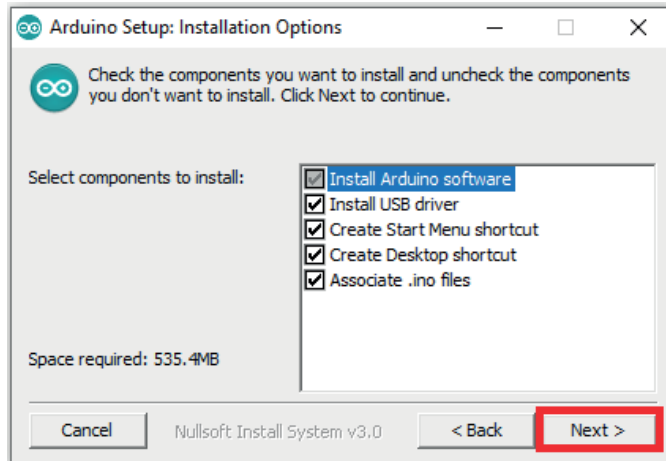
Görsel 2.11: Kurulum dosyası

2. Adım: Açılan **License Agreement (Lisans Anlaşması)** penceresinde **I Agree (Kabul ediyorum.)** butonu ile anlaşmayı kabul ediniz (Görsel 2.12).



Görsel 2.12: Lisans anlaşması

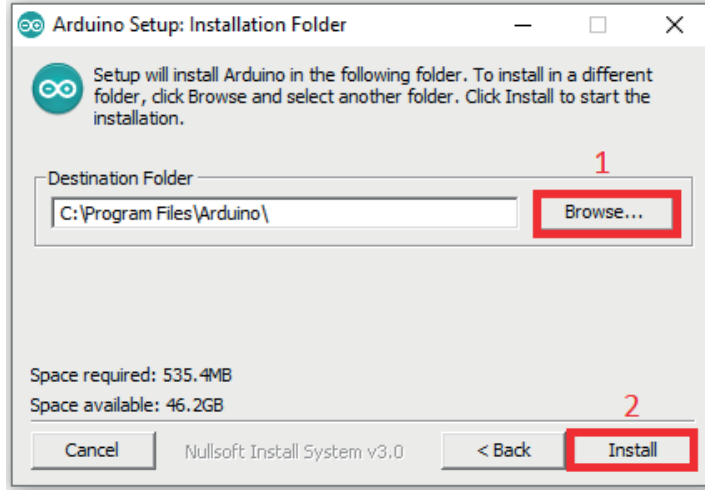
3. Adım: Açılan **Installation Options (Kurulum Seçenekleri)** penceresinde **Next (İleri)** butonuna basınız (Görsel 2.13).



Görsel 2.13: Kurulum seçenekleri

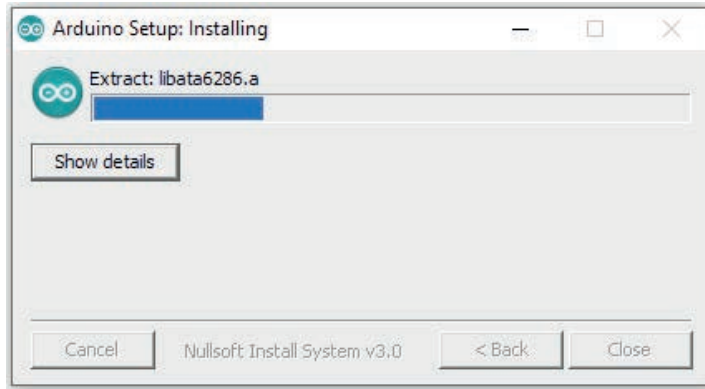


4. Adım: Açılan **Installation Folder (Kurulum Klasörü)** penceresinde **Browse (Araştır)** butonuna basarak programın nereye kuracağını ayarlayınız (Görsel 2.14). Ardından **Install (Yükle)** butonuna basarak kurulum işlemini başlatınız.



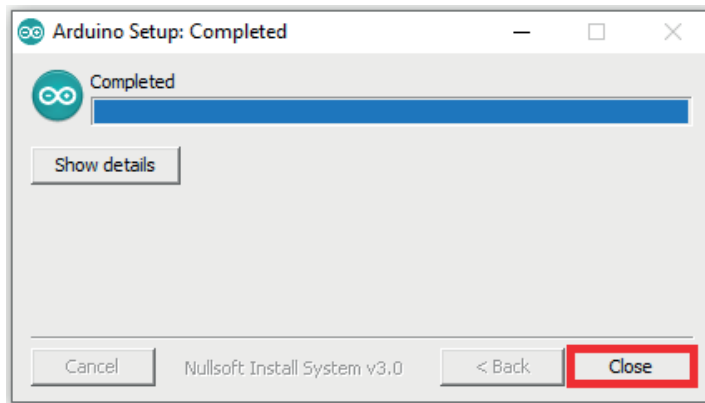
Görsel 2.14: Kurulum işlemi

5. Adım: Açılan **Installing (Yükleme)** penceresinde kurulum dosyalarını yükleyiniz (Görsel 2.15).



Görsel 2.15: Kurulum dosya yükleme

6. Adım: Açılan **Completed (Tamamlandı)** penceresinde kurulum dosyalarının yüklenmesi tamamlandıktan sonra **Close (Kapat)** butonuna basarak kurulum işlemini bitiriniz (Görsel 2.16).



Görsel 2.16: Kurulum dosya yükleme



2.2.3. IDE Platformu ve Mikrodenetleyici Uygulama Kartı Arasında Bağlantı Sağlamak

IDE platformunun kurulumu tamamlandıktan sonra mikrodenetleyici uygulama kartı arasında bağlantı sağlanır.

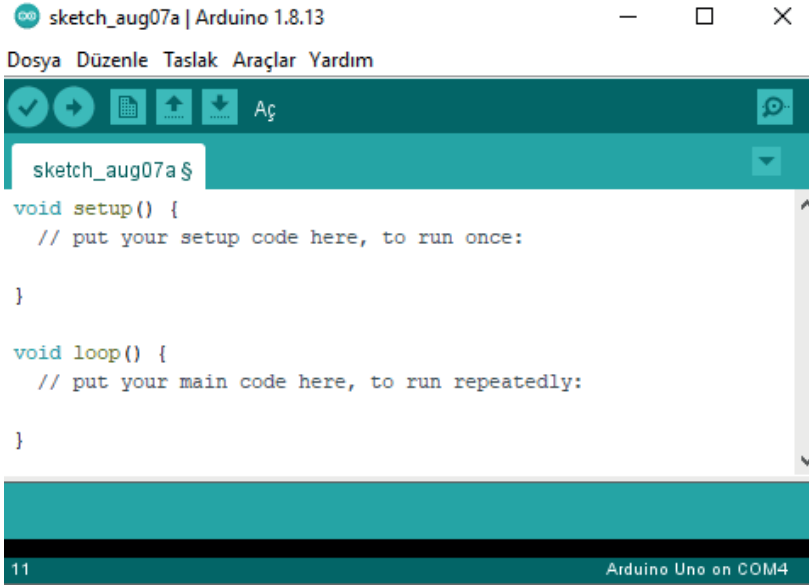


3. Uygulama

İşlem adımlarına göre IDE platformu ile mikrodenetleyici uygulama kartı arasında bağlantı sağlayınız.

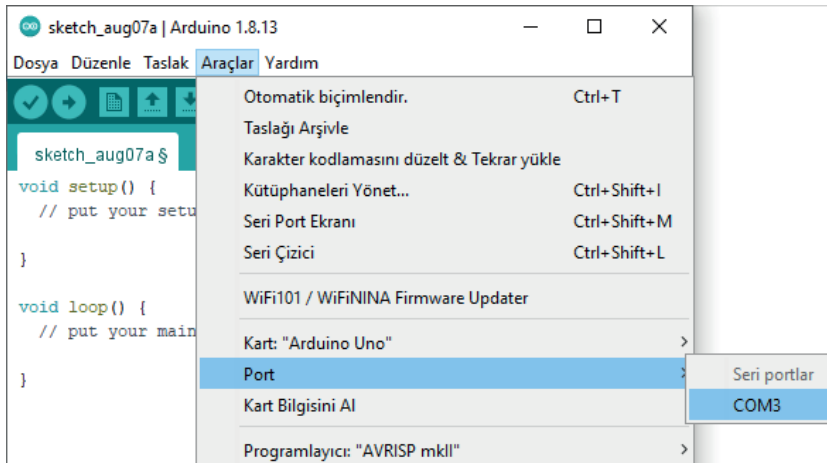
1. Adım: USB kablosu kullanarak mikrodenetleyici uygulama kartının bilgisayar bağlantısını yapınız.

2. Adım: Bilgisayara kurulan IDE platformunu açınız (Görsel 2.17).



Görsel 2.17: IDE platformu

3. Adım: IDE platformu açıldıktan sonra Görsel 2.18'de görüldüğü gibi USB portunun Araçlar menüsü Port sekmesinden bilgisayara bağlantının ayarlanması gerekir. Bu bağlantı portu, USB bağlantı noktalarına göre (COM1, COM2, COM3 vb.) değişiklik gösterebilir. Port ayarlamasını yaptıktan sonra IDE platformu ve mikrodenetleyici uygulama kartı arasındaki bağlantıyı sağlayınız.



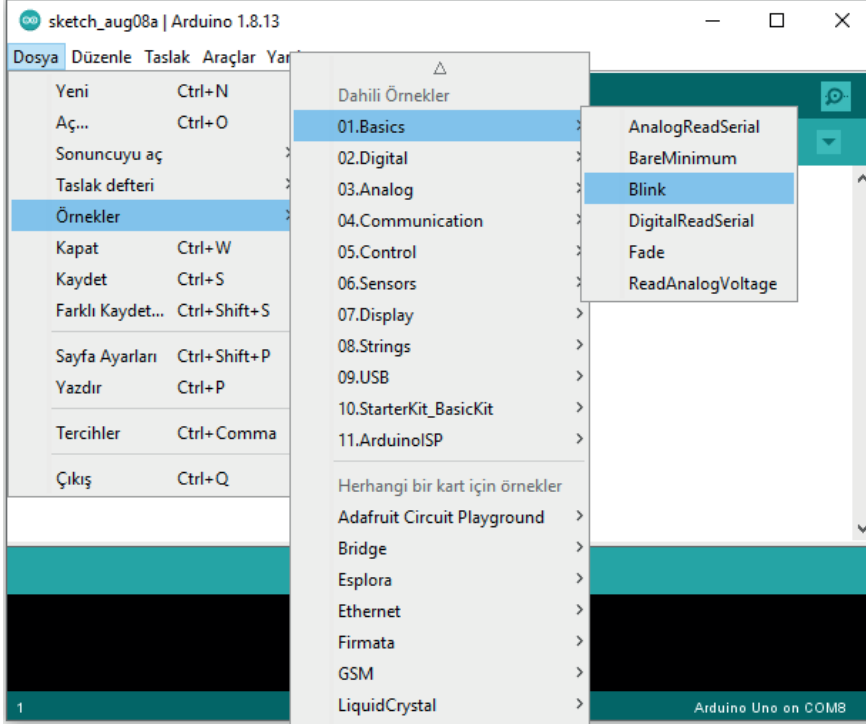
Görsel 2.18: Bağlantı portu seçimi



4. Uygulama

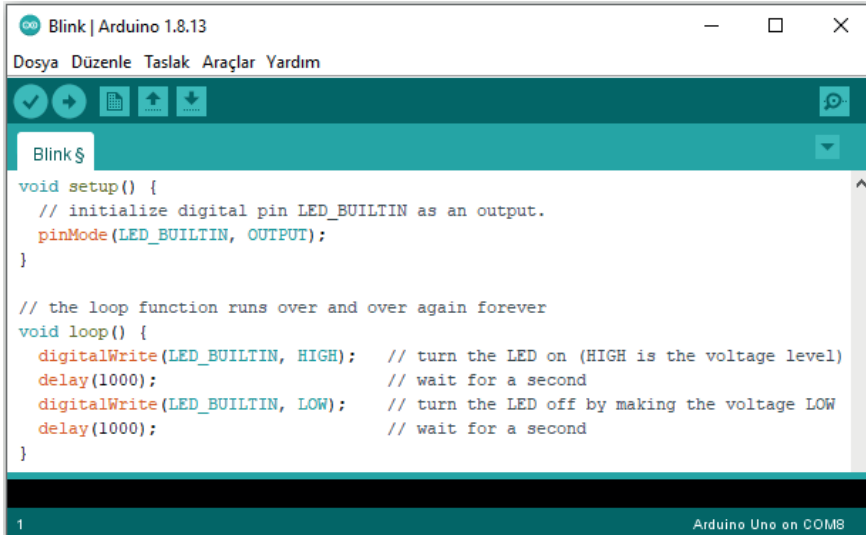
Mikrodenetleyici uygulama kartının çalışmasını kontrol etmek için IDE platformunun içindeki örnek kodlar denenebilir. Bunlardan biri, **Blink (Göz kırpma)** programıdır. Blink programı için mikrodenetleyici uygulama kartına ilave devre elemanlarının takılmasına gerek yoktur. İşlem adımlarına göre Blink programını çalıştırınız.

1. Adım: IDE platformundaki Blink programını açınız (Görsel 2.19).



Görsel 2.19: Blink programının açılması

2. Adım: IDE platformundaki Blink programının kodlarını inceleyiniz (Görsel 2.20). Mikrodenetleyici uygulama kartı üzerindeki 13 numaralı dijital giriş/çıkış portuna bağlı LED, 1 saniye aralıklarla yanıp sönecektir.

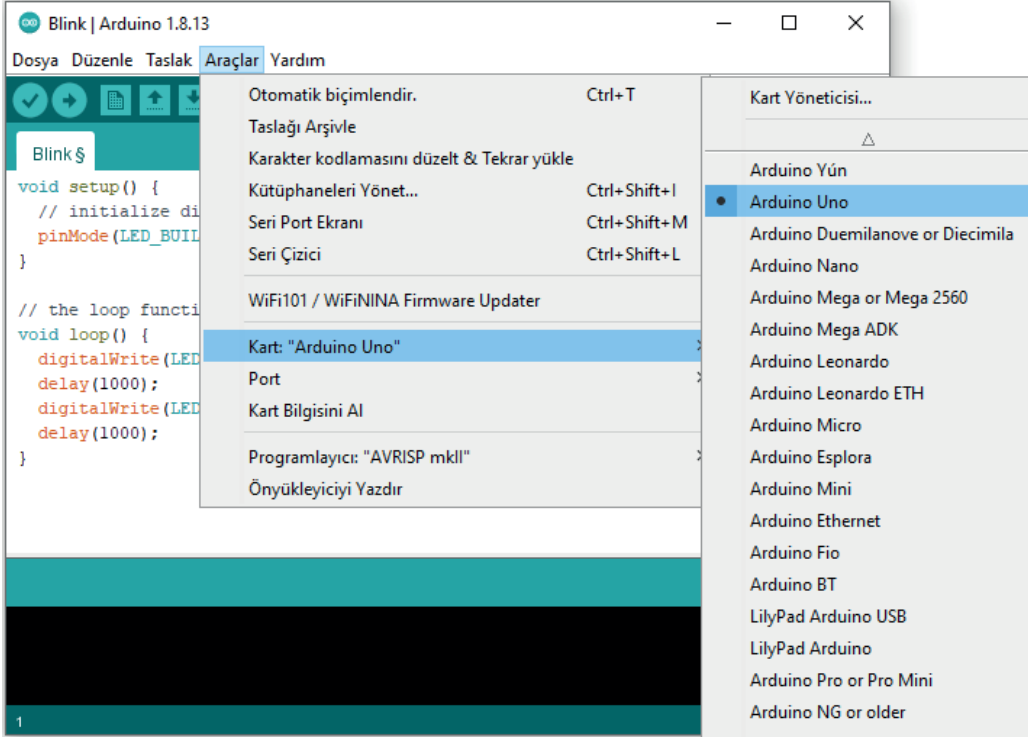


Görsel 2.20: Blink programının açılması



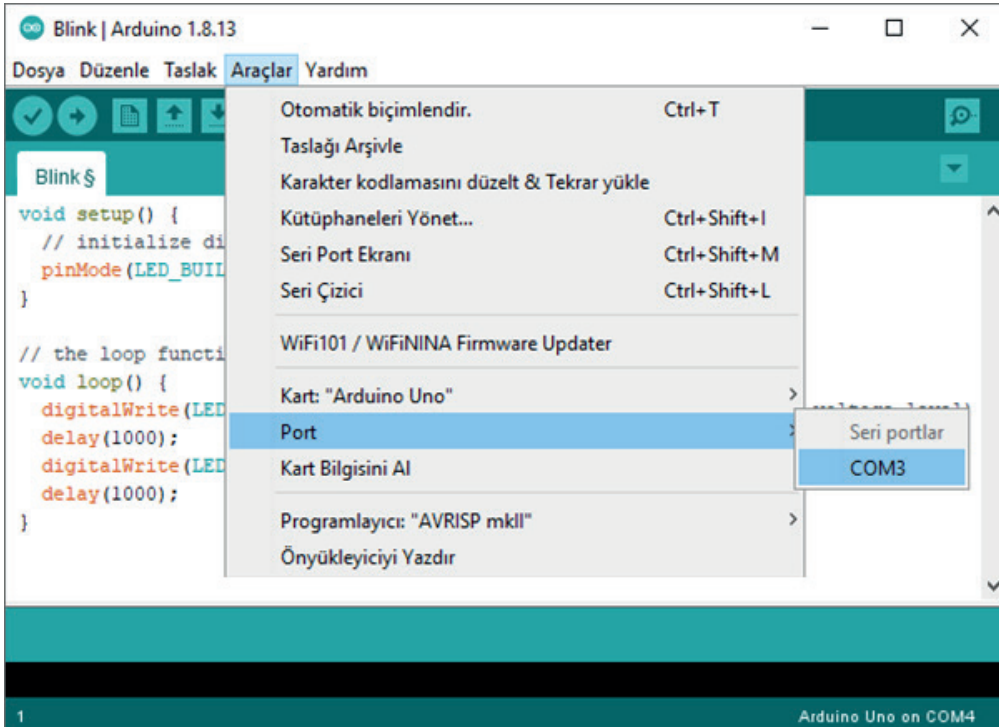
3. Adım: USB kablosu kullanarak mikrodenetleyici uygulama kartının bilgisayar bağlantısını yapınız.

4. Adım: Araçlar menüsünden mikrodenetleyici uygulama kartı türünü seçiniz (Görsel 2.21).



Görsel 2.21: Blink programının açılması

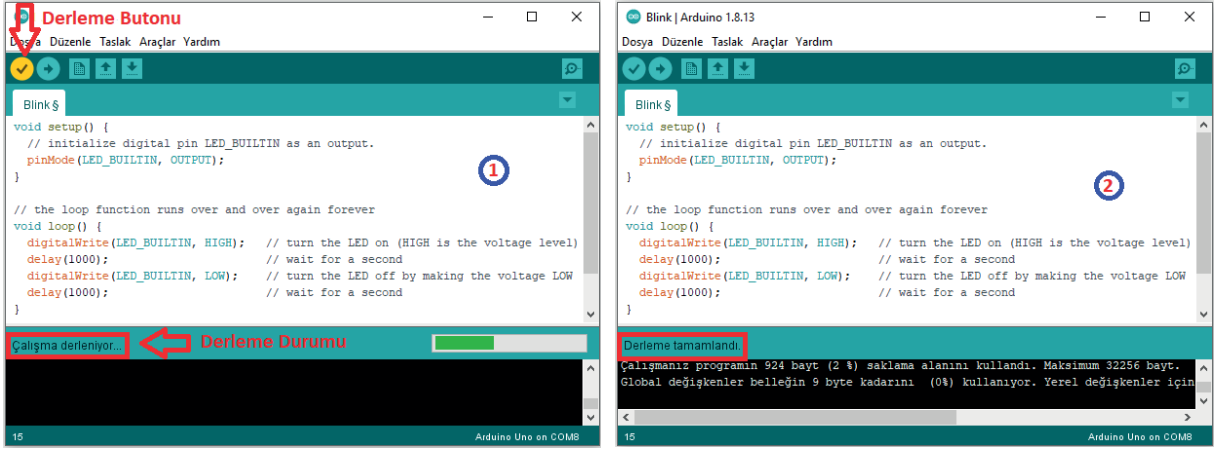
5. Adım: Araçlar menüsünden mikrodenetleyici uygulama kartının bilgisayara bağlandığı USB portunu seçiniz (Görsel 2.22).



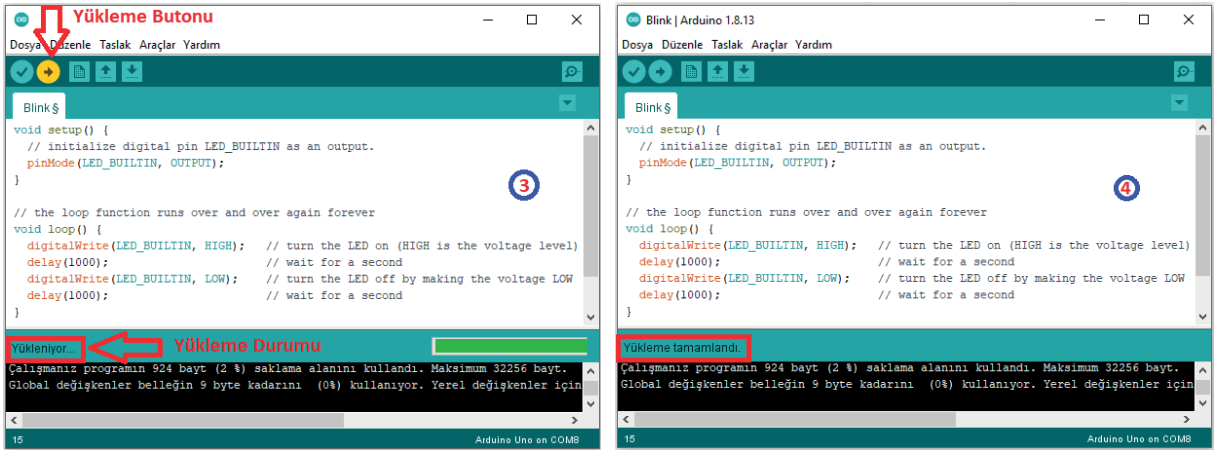
Görsel 2.22: Bağlantı portu seçimi



6. Adım: Derleme butonuna basınız (Görsel 2.23). Hata yoksa Yükleme butonuna basarak programın mikrodnetleyici uygulama kartına yüklenmesini sağlayınız. Yükleme tamamlandığında “Yükleme tamamlandı.” bilgisi IDE platformunun sol alt köşesinde verilir (Görsel 2.24). Mikrodnetleyici uygulama kartı üzerindeki Blink LED’inin 1 saniye aralıklarla yanıp söndüğünü gözlemleyiniz.



Görsel 2.23: Programın derlenmesi



Görsel 2.24: Mikrodnetleyici uygulama kartına programın yüklenmesi



Sıra Sizde

IDE platformundaki örnek uygulamalardan birini açarak mikrodnetleyici uygulama kartına yükleyiniz.



ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantezlerin içine yargılar doğru ise “D”, yanlış ise “Y” yazınız.

- () Merkezî işlem birimi, ilgili programları çalıştırmayı ve diğer birimlerle haberleşmeyi sağlar.
- () ROM, geçici bilgilerin tutulduğu bellektir.
- () Yazılan programın mikrodenetleyiciye yüklenmesi için USB portu kullanılır.
- () Mikrodenetleyici uygulama kartının haricî besleme gerilimi 3-5 voltur.
- () Mikrodenetleyici uygulama kart çeşitlerinin dijital giriş/çıkış pin sayıları aynıdır.
- () IDE platformu, program kodlarının yazıldığı yerdir.
- () NANO, mikrodenetleyici uygulama kartı çeşitlerinden biridir.
- () Platformdaki örnek uygulamalar Dosya menüsünden açılır.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

9. Aşağıdakilerden hangisi kalıcı bilgilerin tutulduğu bellek türü değildir?

- RAM
- ROM
- PROM
- EPROM
- EEPROM

10. Aşağıdakilerden hangisi mikrodenetleyici uygulama kartı çeşitlerinden değildir?

- UNO
- MİKRO
- NANO
- MEGA
- ATMega

11. Aşağıdakilerden hangisi program kodlarının yazıldığı platformun adıdır?

- MİB
- ROM
- IDE
- USB
- RAM

12. Aşağıdakilerden hangisi IDE platformunda bağlantı için port seçiminin yapıldığı menünün adıdır?

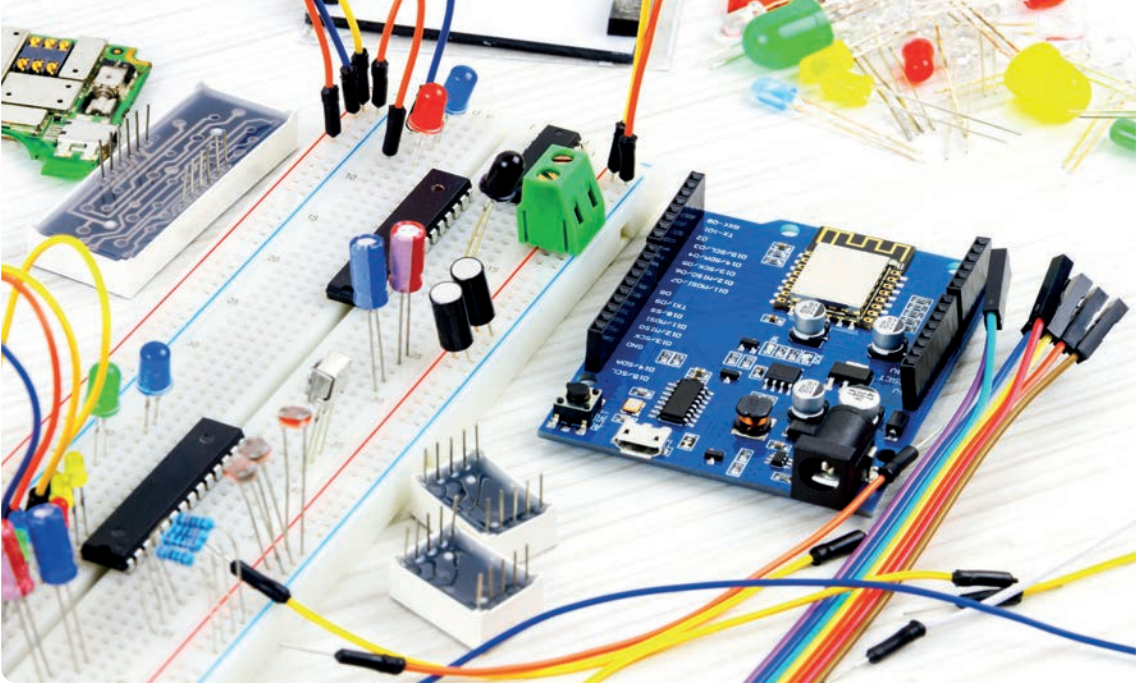
- Dosya
- Düzenle
- Taslak
- Araçlar
- Yardım

13. Aşağıdakilerden hangisi platformdaki örnek uygulamaların açıldığı menüdür?

- Dosya
- Düzenle
- Taslak
- Araçlar
- Yardım

3. ÖĞRENME BİRİMİ

PROGRAMLAMA PLATFORMU UYGULAMALARI



KONULAR

- 3.1. LED İLE YAPILAN UYGULAMALAR
- 3.2. BİYOMEDİKAL SENSÖR UYGULAMALARI
- 3.3. BİYOMEDİKAL CİHAZ MODÜLLERİNDEKİ UYGULAMALAR

ANAHTAR KELİMELE

- Buzzer
- Devre tasarım ve simülasyon programı
- Direnç
- LED
- Nabız sensörü
- Sensör
- Ultrasonik mesafe sensörü

NELER ÖĞRENECEKSİNİZ?

- LED yapısı ve elektronik devrelerdeki bağlantısı
- Ohm Kanunu
- Dirençlerin renk kodları
- Breadboard üzerine elektronik devrelerin kurulumu
- Devre tasarım ve simülasyon programıyla devre kurup çalıştırma
- Mikrodenetleyici ile LED uygulamaları
- Mesafe sensör yapısı ve uygulamaları
- Buzzer yapısı ve uygulamaları
- Nabız sensör yapısı ve kullanımı
- Işığa duyarlı dirençlerin kullanımı
- Sıcaklık sensörü kullanımı





HAZIRLIK ÇALIŞMALARI

1. Elektriksel büyüklükler ve birimleri hakkındaki bilgilerinizi arkadaşlarınızla paylaşınız.
2. Algılayıcı, dijital, analog, giriş ve çıkış kavramları size ne ifade ediyor?

3.1. LED İLE YAPILAN UYGULAMALAR

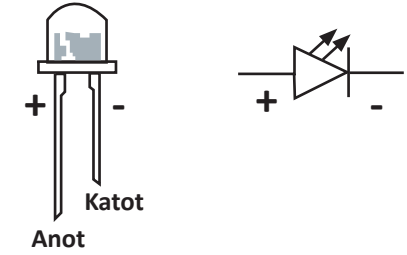
Tıbbi cihazlardaki ölçümler sonucunda yapılan bilgilendirme ve uyarılarda genellikle LED kullanılır. Günümüzdeki tıbbi cihazlarda mikrodenetleyicilerden yaygın olarak yararlanılır. Mikrodenetleyicilerde gerekli programlar yazılıp uygun bağlantılar sağlandıktan sonra LED'lerin çalışma zamanları, parlaklık ayarları ve LED'lerin sıralı yanmaları gibi işlemler yapılabilir. Ayrıca çıkışların durumlarını gözlemlemek için de LED kullanılabilir.

3.1.1. LED Yapısı

Doğru yönde DC gerilim uygulandığında ışık yayan yarı iletken elemana **LED diyot** denir. Kısaca **LED** olarak da ifade edilir.

Görsel 3.1'de görüldüğü gibi LED'in **anot (+)** ve **katot (-)** olmak üzere iki bacağı vardır. LED'in ışık yayması için gerilim kaynağının pozitif (+) ucu anot, negatif (-) ucu katot bacağına bağlanmalıdır. Ters bağlantıda LED ışık yaymaz.

Görsel 3.2'de çeşitli renklerde LED'ler gösterilmiştir. LED'lerin rengine göre çalışma gerilimi ve akım değerleri Tablo 3.1'de verilmiştir. LED'lere çalışma geriliminin üstünde bir gerilim uygulanacaksa seri olarak bir ön direnç bağlanmalıdır. Aksi durumda LED'ler bozulur.



Görsel 3.1: LED bacakları ve sembolü



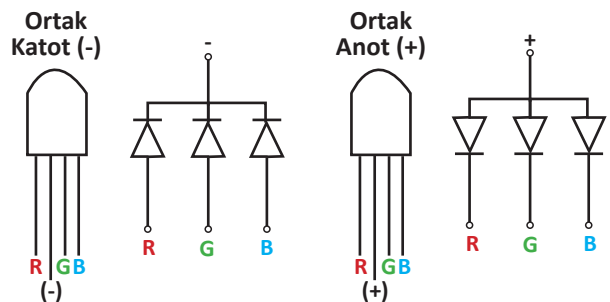
Görsel 3.2: LED

Tablo 3.1: LED Çalışma Gerilim ve Akım Değerleri

LED Renkleri	Çalışma Gerilimi (V)	Çalışma Akımı (mA)
Kırmızı	1,5 V	15 mA
Sarı	1,8 V	15 mA
Yeşil	2,2 V	15 mA
Mavi	3 V	30 mA
Beyaz	3 V	30 mA

3.1.2. LED Çeşitleri

Üç ana renkli LED'in tek kılıf içinde paketlenmesiyle oluşan LED'e **RGB LED** denir. RGB ismi, renklerin baş harflerinin (**R**ed "kırmızı", **G**reen "yeşil", **B**lue "mavi") birleşmesinden oluşmuştur. Ortak anotlu ve ortak katotlu olmak üzere iki çeşidi vardır (Görsel 3.3).



Görsel 3.3: RGB LED çeşitleri



Kızılötesi ışık yayan LED'lere **infrared LED** denir. Bu tarz LED'lerin yanıp söndüğü insan gözü ile algılanamaz. Televizyonların uzaktan kumandalarında infrared LED kullanılır (Görsel 3.4). Kumandanın tuşuna basıldığında LED'in ışık yaydığı ancak kamerayla görülebilir.



Görsel 3.4: Infrared LED

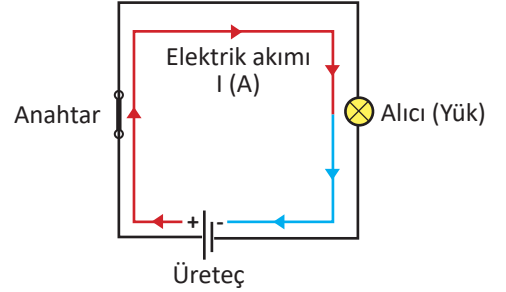


Araştırma

Infrared LED'in ışık yaydığını cep telefonu kamerasıyla gözlemleyerek kullanım alanları ile ilgili araştırma yapınız.

3.1.3. Elektrik Akımı

Elektrik yüklerinin hareket etmesine **elektrik akımı** denir. Elektrik akımı, kapalı bir elektrik devresinde üreticinin etkisiyle oluşan elektron hareketliliği ile gerçekleşir. Elektrik devresinden akımın geçişi Görsel 3.5'te görülür. Elektrik akımı I harfi ile gösterilir. Elektrik akımının birimi amperdir (A).



Görsel 3.5: Elektrik devresinden akımın geçişi

3.1.4. Elektrik Akım Birimleri ve Dönüşümleri

Elektronik devrelerde akımın daha küçük birimleri kullanıldığı için elektrik akım birimlerinin dönüşümleri yapılmalıdır. Elektrik akım birimleri ve çarpanları Tablo 3.2'de verilmiştir.

Tablo 3.2: Akım Birimleri ve Çarpanları

Akım Birimleri	Birimlerin Kısaltması	Çarpan
Kiloamper	kA	10^3 (1000)
Amper	A	1
Miliamper	mA	10^{-3} (0,001)
Mikroamper	μ A	10^{-6} (0,000001)

Örnek

Verilen akım birimlerinin dönüşümünü yapınız.

$$2,5 \text{ kA} = \dots\dots\dots \text{A}$$

$$2,5 \text{ kA} = 2,5 \cdot 10^3 = 2500 \text{ A}$$

$$12 \text{ mA} = \dots\dots\dots \text{A}$$

$$12 \text{ mA} = 12 \cdot 10^{-3} = 0,012 \text{ A}$$

$$0,04 \text{ A} = \dots\dots\dots \mu\text{A}$$

$$0,04 \mu\text{A} = 0,04 \cdot 10^6 = 40000 \mu\text{A}$$



Sıra Sizde

Verilen akım birimlerinin dönüşümlerini yapınız.

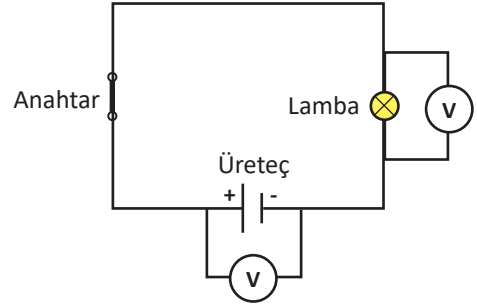
$$0,21 \text{ A} = \dots\dots\dots \text{mA}$$

$$0,01 \text{ kA} = \dots\dots\dots \text{A} \quad 58000 \mu\text{A} = \dots\dots\dots \text{A}$$



3.1.5. Gerilim

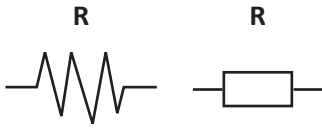
Bir elektrik devresinde iki nokta arasındaki potansiyel farka **gerilim** denir (Görsel 3.6). Gerilim, V harfi ile gösterilir. Gerilim birimi volt (V). Elektrik devresinde akım geçişinin sağlanabilmesi için elektrik kaynağının uçları arasında bir gerilimin olması gerekir.



Görsel 3.6: Elektrik devresinde gerilim

3.1.6. Direnç

Elektrik akımına karşı gösterilen zorluğa **direnç** denir. Direnç, R harfi ile gösterilir. Direnç birimi ohmdur (Ω). Görsel 3.7'de direncin sembolü, Görsel 3.8'de değerleri farklı dirençler görülür.



Görsel 3.7: Direnç sembolü



Görsel 3.8: Dirençler

3.1.7. Direnç Renk Kodları

Direnç değerlerini ölçmeden bulmak amacıyla dirençlerin üzerinde renk bantları vardır. Her rengin rakamsal bir karşılığı mevcuttur (Görsel 3.9).

Renk	Siyah	Kahverengi	Kırmızı	Turuncu	Sarı	Yeşil	Mavi	Mor	Gri	Beyaz
Rakamsal Karşılığı	0	1	2	3	4	5	6	7	8	9

Görsel 3.9: Dirençlerde renkler ve rakamsal karşılıkları

Direnç değeri, rengin direnç üzerindeki sırası ve rakamsal karşılığına göre belirlenir. Renk sıralaması yapılırken tolerans renk sağa gelecek şekilde direnç tutulur. Tolerans rengin diğer renk şeritleriyle arasındaki mesafe daha fazladır. Direnç renkleri soldan başlayarak sıralanır.

- Dört renkli dirençte ilk iki rengin rakamsal karşılığı yazılır. Üçüncü renk çarpan, dördüncü renk toleranstır (Görsel 3.10).

1.BANT		2.BANT		ÇARPAN		TOLERANS	
0	SIYAH	0	SIYAH	0	SIYAH	SIYAH	
1	KAHVERENGİ	1	KAHVERENGİ	1	KAHVERENGİ	KAHVERENGİ	±%1
2	KIRMIZI	2	KIRMIZI	2	KIRMIZI	KIRMIZI	±%2
3	TURUNCU	3	TURUNCU	3	TURUNCU	TURUNCU	
4	SARI	4	SARI	4	SARI	SARI	
5	YEŞİL	5	YEŞİL	5	YEŞİL	YEŞİL	±%0,5
6	MAVİ	6	MAVİ	6	MAVİ	MAVİ	±%0,25
7	MOR	7	MOR			MOR	±%0,1
8	GRİ	8	GRİ			GRİ	±%0,05
9	BEYAZ	9	BEYAZ			RENKSİZ	±%20
					ALTIN	10 ⁻¹ =0,1	±%5
					GÜMÜŞ	10 ⁻² =0,01	±%10

Görsel 3.10: Dört renkli dirençlerin değerinin bulunması



Örnek

Görsel 3.11'deki direncin değerini bulunuz.



Görsel 3.11: Direnç

Çözüm

Kahverengi - Mavi - Kırmızı - Altın

$$1 \quad 6 \quad 10^2 \quad \%5$$

$$16 \times 10^2 = 1600 \Omega \pm \%5$$

Örnek

Değeri 10 kΩ ve toleransı ± %10 olan dört renkli direncin renklerini bulunuz.

Çözüm

$$10 \text{ k}\Omega = 10 \times 10^3 = 10 \times 1000 = 10.000 \Omega \pm \%10$$

$$1 \quad 0 \quad 10^3 \quad \%10$$

Kahverengi - Siyah - Turuncu - Gümüş

- Beş renkli dirençlerde ilk üç rengin rakamsal karşılığı yazılır. Dördüncü renk çarpan, beşinci renk toleranstır (Görsel 3.12).



1.BANT		2.BANT		3.BANT		ÇARPAN		TOLERANS	
0	SIYAH	0	SIYAH	0	SIYAH	0	SIYAH	SIYAH	
1	KAHVERENGİ	1	KAHVERENGİ	1	KAHVERENGİ	1	KAHVERENGİ	KAHVERENGİ	±%1
2	KIRMIZI	2	KIRMIZI	2	KIRMIZI	2	KIRMIZI	KIRMIZI	±%2
3	TURUNCU	3	TURUNCU	3	TURUNCU	3	TURUNCU	TURUNCU	
4	SARI	4	SARI	4	SARI	4	SARI	SARI	
5	YEŞİL	5	YEŞİL	5	YEŞİL	5	YEŞİL	YEŞİL	±%0,5
6	MAVİ	6	MAVİ	6	MAVİ	6	MAVİ	MAVİ	±%0,25
7	MOR	7	MOR	7	MOR			MOR	±%0,1
8	GRİ	8	GRİ	8	GRİ			GRİ	±%0,05
9	BEYAZ	9	BEYAZ	9	BEYAZ			RENKSİZ	±%20
							ALTIN $10^{-1}=0,1$	ALTIN	±%5
							GÜMÜŞ $10^{-2}=0,01$	GÜMÜŞ	±%10

Görsel 3.12: Beş renkli dirençlerin değerinin bulunması

Örnek

Görsel 3.13'teki direncin değerini bulunuz.



Görsel 3.13: Direnç

Çözüm

Mavi - Yeşil - Kırmızı - Kahverengi - Altın

$$6 \quad 5 \quad 2 \quad 10^1 \quad \%5$$

$$652 \times 10^1 = 6520 \Omega \pm \%5$$



Örnek

Değeri 270Ω ve toleransı $\pm \%1$ olan beş renkli direncin renklerini bulunuz.

Çözüm

$$270 \times 10^0 = 270 \times 1 = 270 \Omega \pm \%1$$

2 7 0 10^0 $\%1$

Kırmızı - Mor - Siyah - Siyah - Kahverengi



Sıra Sizde

1. Görsel 3.14'te renkleri verilen dirençlerin değerini bulunuz.



Görsel 3.14: Dirençler

2. Değerleri verilen dirençlerin renk kodlarını bulunuz.

$220 \Omega \pm \%5$ (Dört renkli) $576 \text{ k}\Omega \pm \%10$ (Beş renkli)

3.1.8. Ohm Kanunu

Bir elektrik devresinde gerilim, akım ve direnç arasındaki bağıntıya **Ohm Kanunu** denir (Görsel 3.15).

	$V=I \times R$	$I=V/R$	$R=V/I$

Görsel 3.15: Ohm Kanunu

**Örnek**

Görsel 3.16'daki devrede dolaşan akımın kaç amper olduğunu bulunuz.

Çözüm

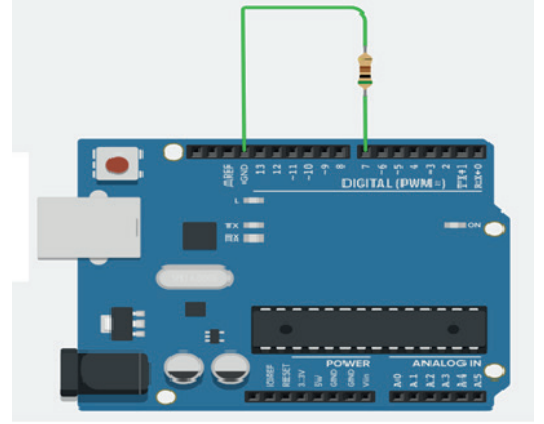
Direncin Renkleri: Yeşil-Siyah-Kahverengi-Altın

Direnc Değeri

$$R = 50 \times 10^1 = 500 \Omega \pm \%5$$

Mikrodenetleyici, dijital çıkış uçları aktif olduğunda 5 V gerilim verir.

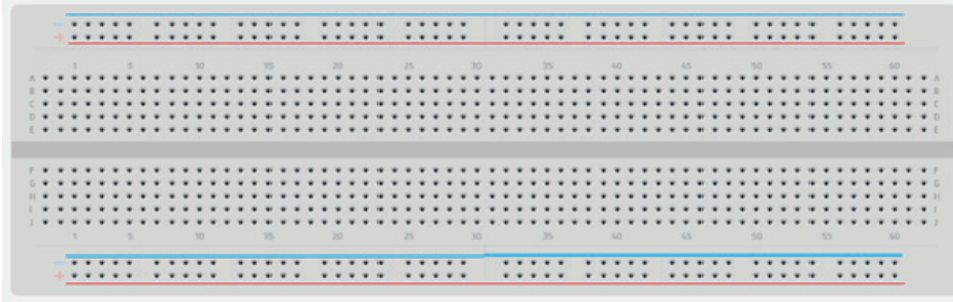
$$V = 5 \text{ V} \quad I = V/R \quad I = 5/500 \quad I = 0,01 \text{ A}$$



Görsel 3.16: Elektrik devresi

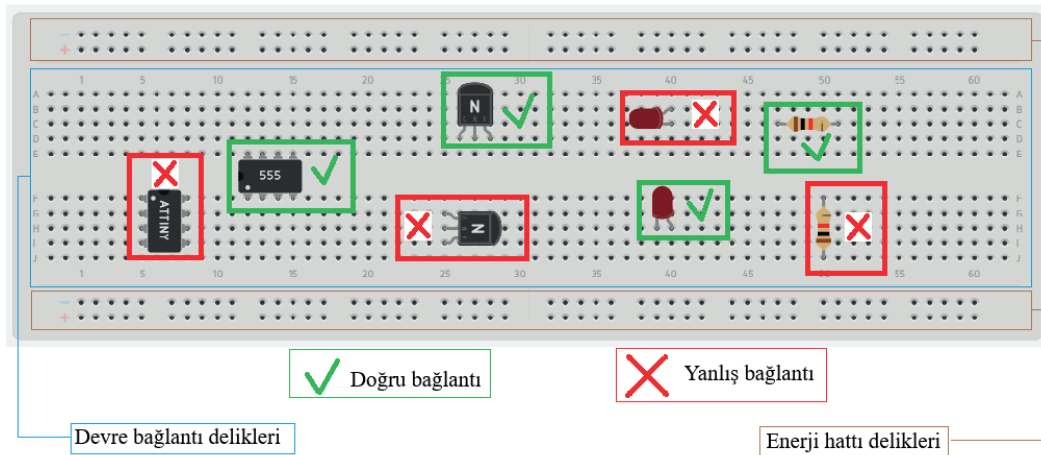
3.1.9. Breadboard ve Yapısı

Elektronik devrelerde lehimleme işlemi gerçekleştirilmeden elemanların ve devrenin bağlantılarının kolayca yapılmasına olanak sağlayan deneme kartlarına **breadboard** denir. Yalıtım için plastik bir kılıfı bulunan breadboard, elektronik elemanların ve iletkenlerin girebileceği delikli bir yapıya sahiptir (Görsel 3.17).



Görsel 3.17: Breadboard

Enerji bağlantı (kırmızı "+", mavi "-") delikleri satır olarak (çizgi boyunca), devre bağlantı yerinde ise bir sütun ve beş satır kısa devre olarak birbirine bağlıdır. Breadboard üzerindeki elemanların bağlantıları Görsel 3.18'de görülür.

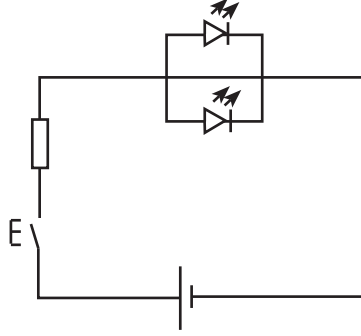


Görsel 3.18: Breadboard kullanımı



Sıra Sizde

Görsel 3.19'da verilen devreyi breadboard üzerinde kurunuz.



Görsel 3.19: Elektrik devresi

3.1.10. Devre Tasarım ve Simülasyon Programını Kullanmak

Basit ve orta seviye mikrodenetleyici uygulamaları için devre tasarım ve simülasyon programı ile çeşitli devreler kurulabilir. Böylece devrenin fiziksel kurulumunda önce simülasyonu gerçekleştirilerek devrenin çalışması kontrol edilebilir. Program, çevrimiçi ortamda çalışır. Bu yüzden herhangi bir program kurmaya gerek yoktur. Programın kullanımı, birinci uygulamada verilmiştir.



1. Uygulama

İşlem adımlarına göre devre tasarım ve simülasyon programına giriş yapınız.

1. Adım: Devre tasarım ve simülasyon programına giriş yapmak için öncelikle oturum açmanız gerekir. İnternet tarayıcınıza <https://www.tinkercad.com/> yazınız. Giriş yap butonuna tıklayarak Görsel 3.20'deki seçeneklerden kendinize uygun olanı ile hesap oluşturunuz.

Tekrar hoş geldiniz

Tinkercad'i ne amaçla kullanacaksınız?

Okulda

Eğitimciler

Sınıf Kodu olan öğrenciler

Öğrenci hesapları

Kendi kendinize

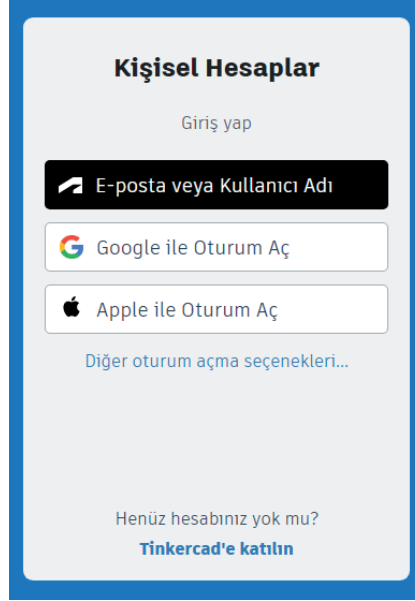
Kişisel hesaplar

Henüz hesabınız yok mu?
[Tinkercad'e katılın](#)

Görsel 3.20: Devre tasarım ve simülasyon programı hesap oluşturma ekranı

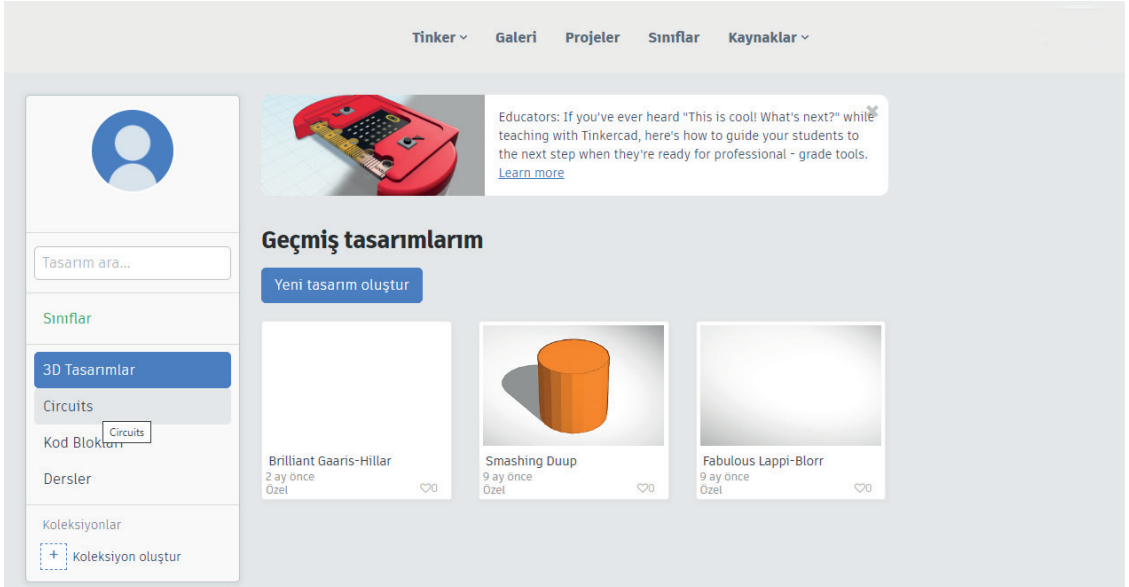


2. Adım: Görsel 3.21'de belirtilen mail hesaplarınız (Gmail, Apple) varsa onları kullanarak, kaydolmadan uygulamaya giriş yapınız.



Görsel 3.21: Uygulamaya kayıt ekranı

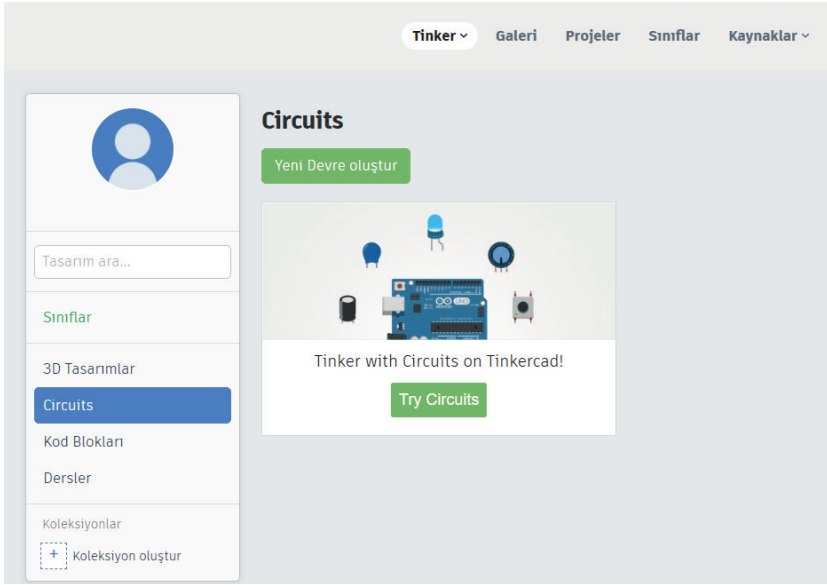
3.Adım: Devre tasarım ve simülasyon uygulamasına giriş yapıldığında ekrana 3D tasarım ortamı gelir. Devre tasarım ve simülasyon uygulaması için **Circuits (Devreler)** seçeneğine tıklayınız (Görsel 3.22).



Görsel 3.22: Uygulama ekranı



4. Adım: Görsel 3.23'teki gibi devre oluşturma ekranı ile karşılaşılır. Uygulamanın bu kısmında yeni devre oluşturabilir veya daha önce hazırlanan devrelerde düzenleme yapabilirsiniz. Yeni Devre oluştur butonuna tıklayıp uygulamada kodların yazıldığı, devre tasarımının ve simülasyonun yapıldığı kısma geçiniz.

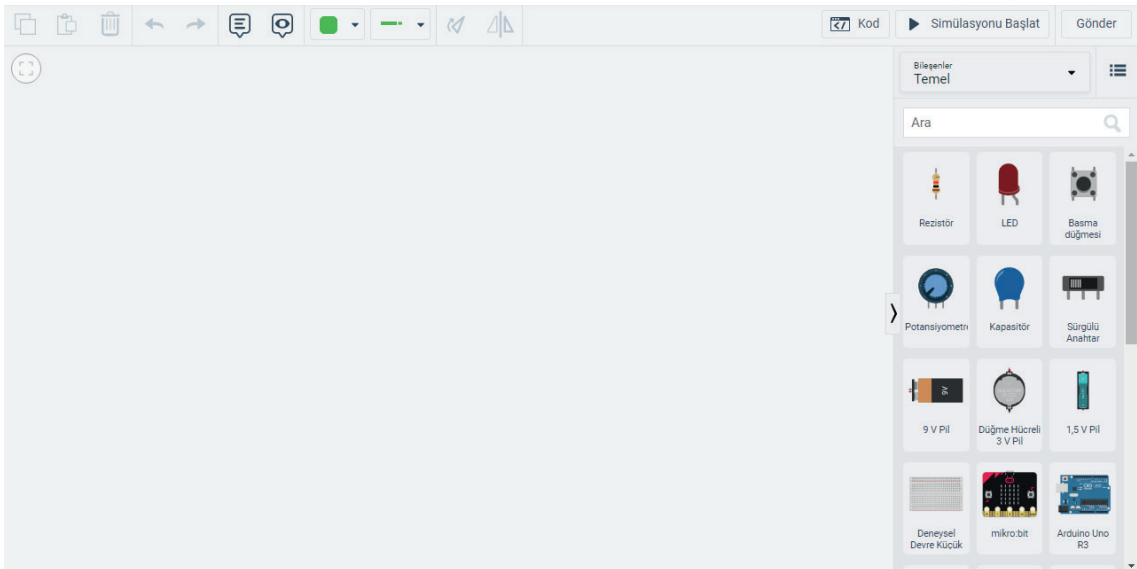


Görsel 3.23: Devre oluşturma ekranı

5. Adım: Kuracağınız devredeki elemanları sürükleyip bırak yöntemiyle devre tasarım alanına getiriniz. Devre bağlantılarını yaptıktan sonra Simülasyonu Başlat butonu ile devrenizin çalışmasını kontrol ediniz (Görsel 3.24).

Not

Mikrodenetleyici uygulamalarında simülasyon başlatılmadan önce kod alanına girilerek ilgili kodlar yazılır.



Görsel 3.24: Devre oluşturma ekranı



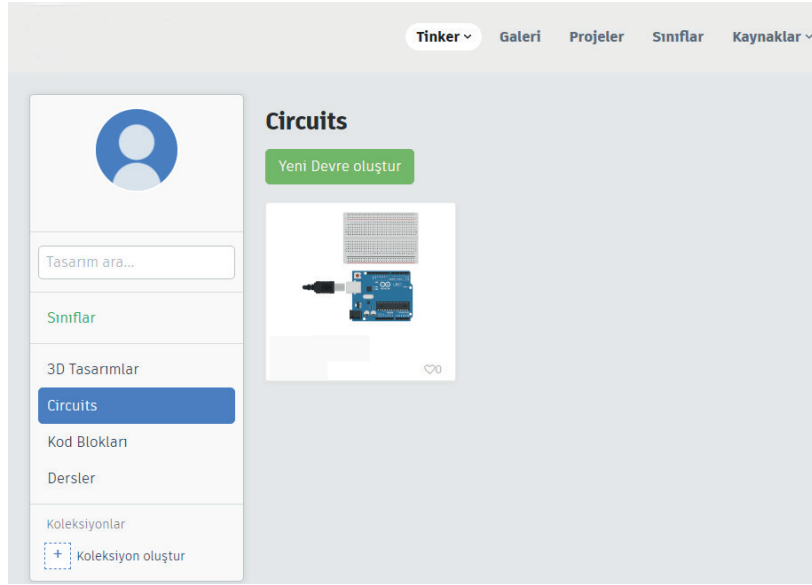
2. Uygulama

İşlem adımlarına göre devre tasarım ve simülasyon programını kullanarak, belirli saniye aralıklarla LED yakıp söndürme uygulamasını yapınız.

Not

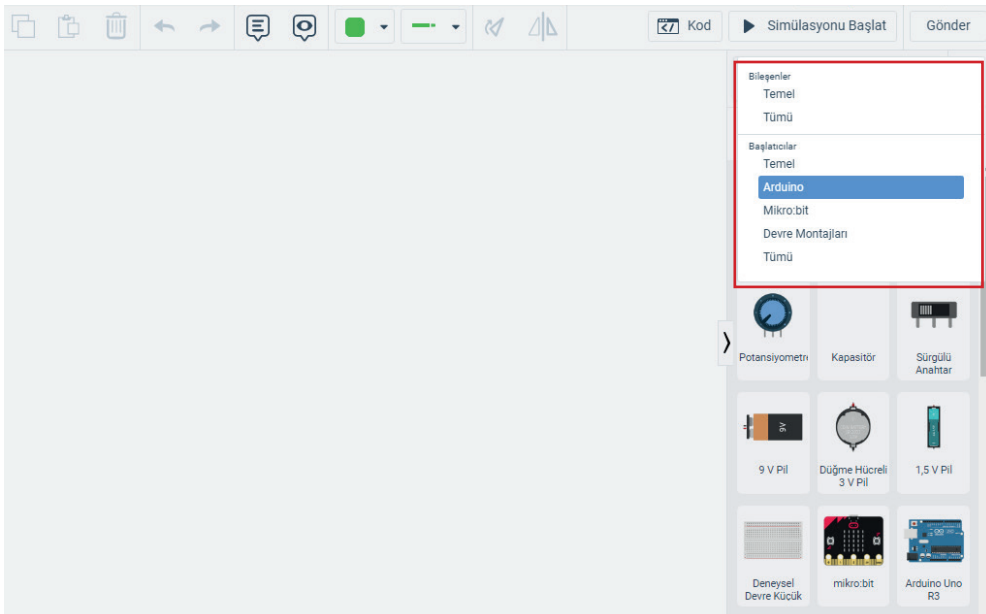
LED'lerin uzun bacağı anot, kısa bacağı katottur. LED'in anot bacağı mikrodenetleyicinin dijital pinine, katot bacağı ise GND pinine bağlanmalıdır. Mikrodenetleyicinin dijital pinleri +5 V gerilim verir. LED'in çalışma geriliminden 5 V büyük olduğu için ön direnç kullanılması zorunludur.

1. Adım: Devre tasarım ve simülasyon programının Circuits ekranında Yeni Devre oluştur butonuna tıklayınız (Görsel 3.25).



Görsel 3.25: Yeni Devre oluştur ekranı

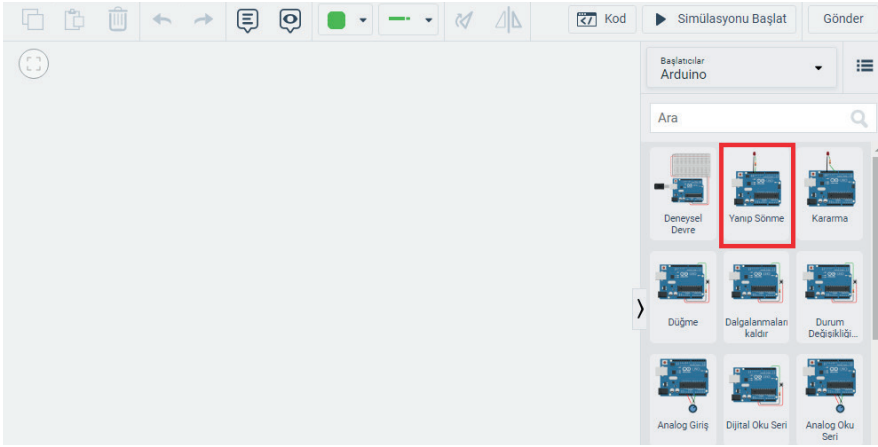
2. Adım: Bileşenler kısmından Arduino seçeneğine tıklayınız (Görsel 3.26).



Görsel 3.26: Mikrodenetleyici devreleri seçimi



3. Adım: Mikrodenetleyici devreleri arasında Yanıp Sönme uygulamasını seçiniz (Görsel 3.27).



Görsel 3.27: LED'in yanıp sönme devresi

Not

Görsel 3.28'de görüldüğü gibi LED'in anot bacağı ön direnç üzerinden 13 No.lu pine, katot bacağı ise mikrodenetleyicinin GND pinine bağlanmıştır. Ön direncin değeri şu şekilde hesaplanır:

$V_{\text{ç}}$: Mikrodenetleyicinin dijital çıkış pinlerinin aktif (HIGH) olduğunda verdiği gerilimin değeri

V_{LED} : Kırmızı LED'in çalışma gerilimi

I : Kırmızı LED'in çalışma akımı

R : Ön direncin değeri

$$V_{\text{ç}} = 5 \text{ V}$$

$$R = (V_{\text{ç}} - V_{\text{LED}}) / I$$

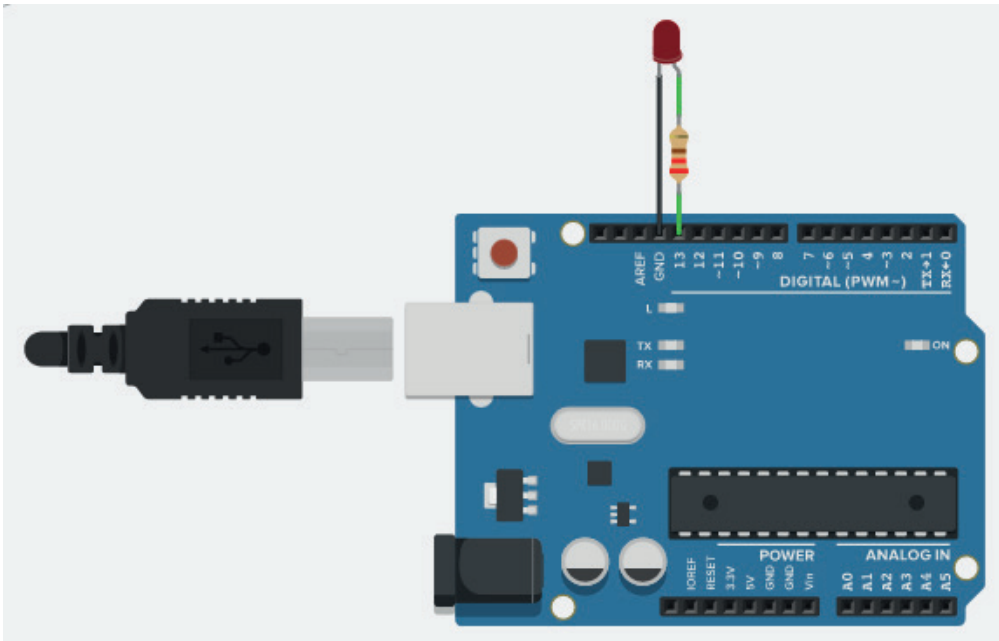
$$V_{\text{LED}} = 1,5 \text{ V}$$

$$R = (5 - 1,5) / 0,015$$

$$I = 15 \text{ mA} = 15 \cdot 10^{-3} = 0,015 \text{ A}$$

$$R = 233,33 \text{ } \Omega \text{ (En yakın 220 ohm direnç seçilir.)}$$

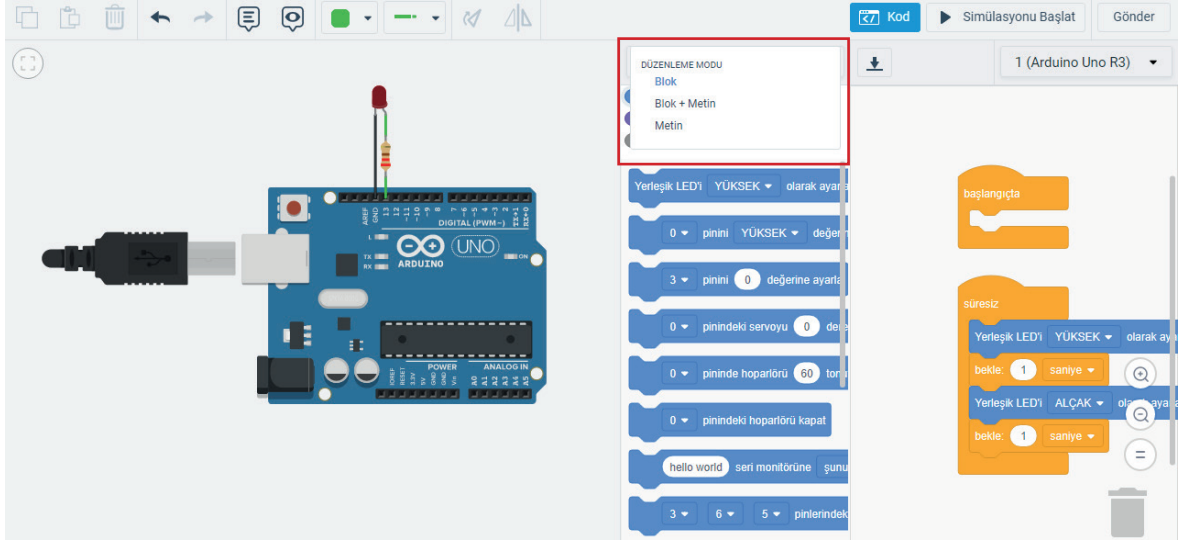
220 Ω Direncin Renk Kodları: Kırmızı-Kırmızı-Kahverengi-Altın



Görsel 3.28: Tasarım alanına devrenin gelmesi



4. Adım: Programdaki hazır kodları açmak için kod sekmesinden Metin seçeneğini tıklayınız (Görsel 3.29).



Görsel 3.29: Metin kodlarını açma

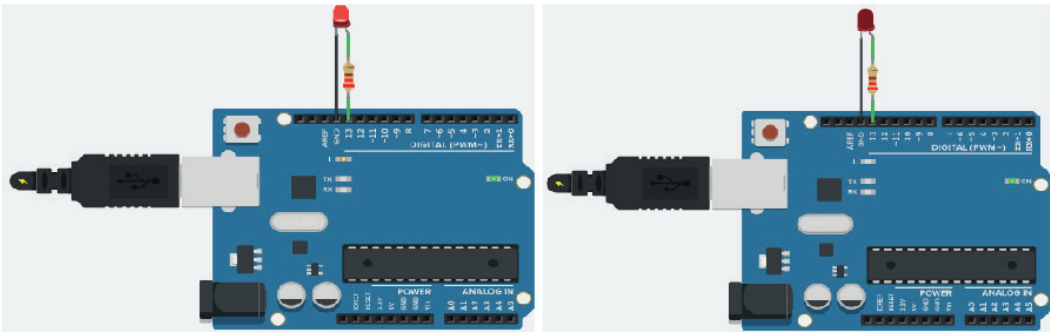
5. Adım: Görsel 3.30'daki kodları inceleyiniz ve mikrodenetleyicinin 13 No.lu pinine LED bağlanacağı için çıkış olarak ayarlandığını gözlemleyiniz.

```
void setup ( ) //Fonksiyon bir kez çalışır. Pinlerin giriş
{
    // veya çıkış olarak belirlendiği kısımdır.
    pinMode(13 , OUTPUT); // Mikrodenetleyicinin 13 No.lu pini
    // çıkış olarak ayarlanmıştır.
}

void loop ( ) // Fonksiyon sonsuz döngü sağlar.
{
    digitalWrite(13 , HIGH); //13 No.lu pin 5 V çıkış verir. LED ışık yayar.
    delay(1000); // 1000 milisaniye ( 1 saniye ) bekler.
    digitalWrite(13 , LOW); //13 No.lu pin 0 V çıkış verir. LED söner.
    delay(1000); // 1000 milisaniye ( 1 saniye ) bekler.
}
```

Görsel 3.30: Kod satırları

6. Adım: Simülasyonu Başlat butonuna basarak, devredeki LED'in birer saniye aralıklarla yanıp söndüğünü gözlemleyiniz (Görsel 3.31).



Görsel 3.31: Devrenin simülasyonu



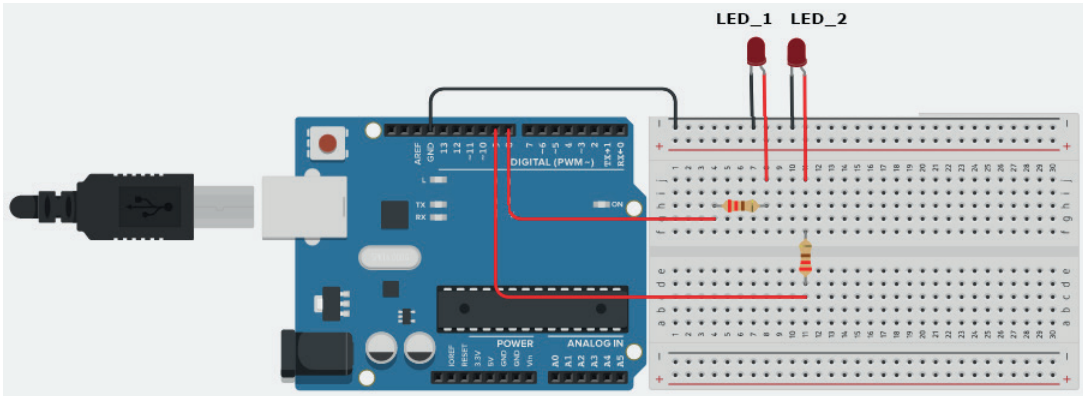
3. Uygulama

İşlem adımlarına göre flip flop uygulamasını yapınız. Flip flop, iki LED'in bulunduğu ve sırayla yanıp söndüğü devrelerdir.

Gerekli Devre Elemanları

- 2 adet LED
- 2 adet 220 Ω (Kırmızı, Kırmızı, Kahverengi, Altın) direnç
- 1 adet mikrodenetleyici
- 1 adet breadboard
- Bağlantı kabloları

1. Adım: Görsel 3.32'deki bağlantıyı yapınız. LED'lerin katot uçlarını mikrodenetleyicinin GND ucuna, anot uçlarını ön direnç üzerinden mikrodenetleyicinin dijital 8 ve 9 No.lu pinlerine bağlayınız.



Görsel 3.32: Devre bağlantı şeması

2. Adım: Mikrodenetleyici IDE programını açarak Görsel 3.33'teki kod satırlarını yazınız. İki LED olduğundan dolayı LED'lerin karışmaması için LED'lere LED_1 ve LED_2 isimlerini veriniz.

Not

Kod satırlarına öncelikle değişken tanımlama ve değer atama ile başlanır. Değişkenler programın tamamında kullanılacaksa en baştan tanımlanmalıdır. Sadece fonksiyonun birinde kullanılacaksa ilgili fonksiyonun başında tanımlama yapılabilir.

```
int LED_1=8; //LED_1 adında değişken tanımlanarak 8 değeri atanır.
int LED_2=9; //LED_2 adında değişken tanımlanarak 9 değeri atanır.
void setup ()
{
  pinMode(LED_1 , OUTPUT); // 8 No.lu pin, çıkış pini olarak ayarlanır.
  pinMode(LED_2 , OUTPUT); // 9 No.lu pin, çıkış pini olarak ayarlanır.
}
void loop ()
{
  digitalWrite(LED_1 , HIGH); //8 No.lu pin 5 V çıkış verir. LED_1 ışık yayar.
  digitalWrite(LED_2 , LOW); //9 No.lu pin 0 V çıkış verir. LED_2 söner.
  delay(1000); // 1000 milisaniye ( 1 saniye ) bekler.
  digitalWrite(LED_1 , LOW); //8 No.lu pin 0 V çıkış verir. LED_1 söner.
  digitalWrite(LED_2 , HIGH); //9 No.lu pin 5 V çıkış verir. LED_2 ışık yayar.
  delay(1000); // 1000 milisaniye ( 1 saniye ) bekler.
}
```

Görsel 3.33: Kod satırları



3. Adım: Kodların yazım işleminden sonra hata kontrolü yapınız. Kod yazımında hata yoksa programı mikrodenetleyiciye yükleyerek devrenin çalışmasını test ediniz.



Sıra Sizde

Görsel 3.32'deki devreyi kurunuz. Kurulumu yaparken mikrodenetleyicinin 5 ve 6 No.lu pinlerini kullanınız. LED'lerin yanma sürelerini iki saniye olacak şekilde ayarlayan kodları IDE programına yazınız. Programı mikrodenetleyiciye yükleyerek devrenin çalışmasını gözlemleyiniz.

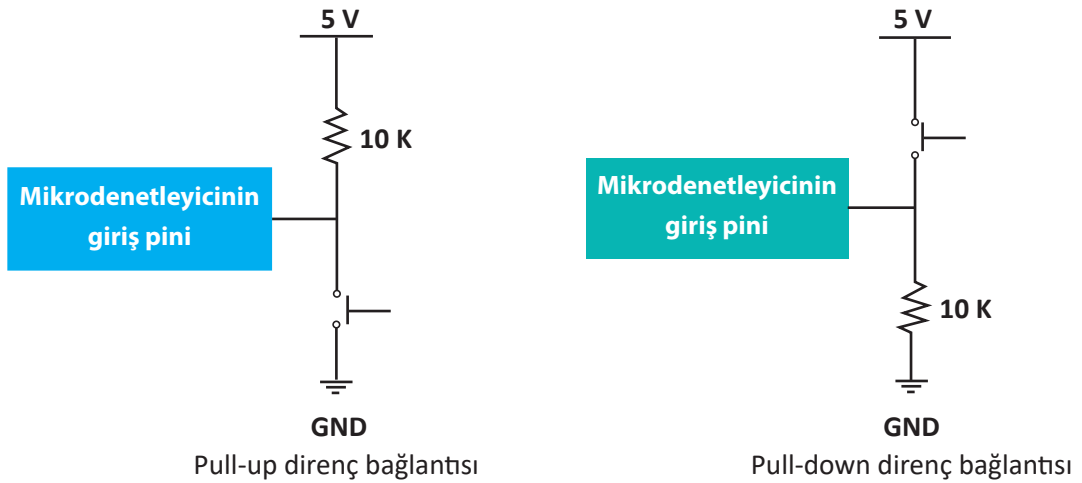


4. Uygulama

İşlem adımlarına göre bir butonla üç LED'i sırasıyla çalıştıran uygulamayı yapınız.

Not

Anahtarlar ve butonlar, elektrik devrelerinde alıcının çalışmasını kontrol etmek amacıyla kullanılan devre elemanlarıdır. Anahtarlar, butonlar ve sensörler mikrodenetleyicinin dijital pinlerine bağlanır. Bu pinler, giriş pini olarak tanımlanmalıdır. Mikrodenetleyicide giriş olarak tanımlanan pinlerin uçları boşa kalırsa bu pinler kararsız çalışır. Bunu önlemek amacıyla pull-up ve pull-down direnç bağlantısı kullanılır (Görsel 3.34).



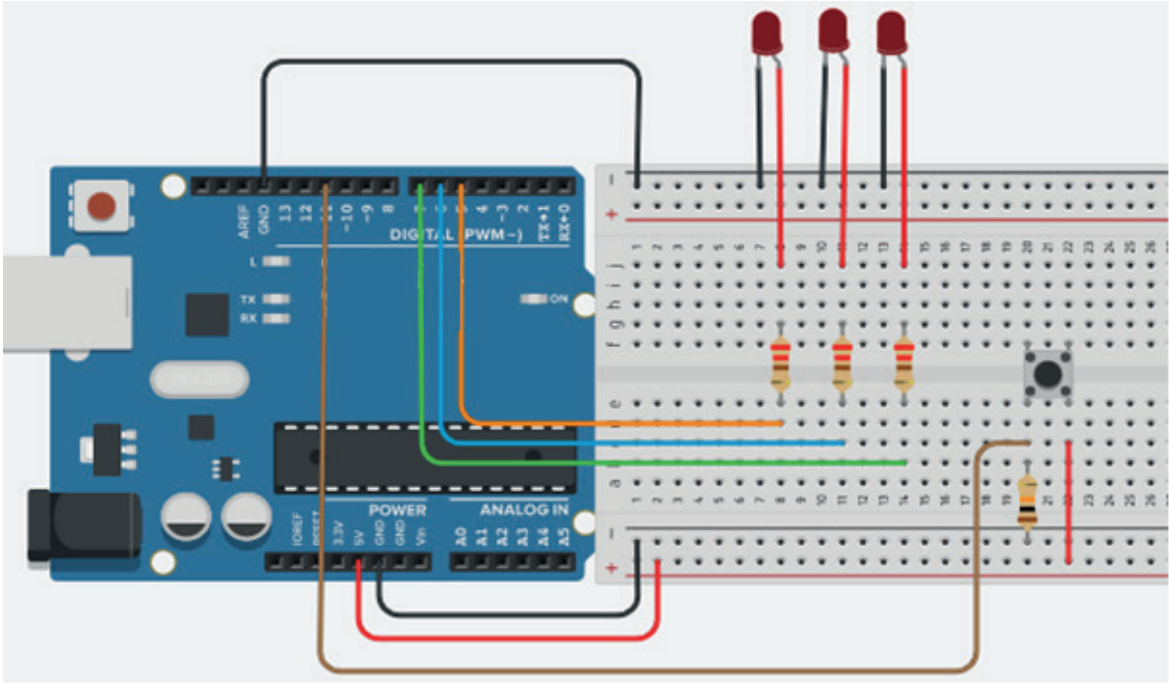
Görsel 3.34: Pull-up ve pull-down direnç bağlantıları

Gerekli Devre Elemanları

- 3 adet LED
- 1 adet buton
- 1 adet 220 Ω (Kırmızı, Kırmızı, Kahverengi, Altın) direnç
- 1 adet 10 K (Kahverengi, Siyah, Turuncu, Altın) direnç
- 1 adet mikrodenetleyici
- 1 adet breadboard
- Bağlantı kabloları



1. Adım: Butona bastığınızda LED'lerin sırasıyla yanıp söndüğü, butondan elinizi çektiğinizde LED'lerin hepsinin aynı anda yanıp söndüğü Görsel 3.35'teki bağlantıyı kurunuz.



Görsel 3.35: Devre bağlantı şeması

Not

Buton bağlantısında pull-down direnç unutulmamalıdır.

2. Adım: Mikrodenetleyici IDE programını açarak Görsel 3.36'daki kod satırlarını yazınız.

Not

Kod satırlarındaki if komutu ile belirlenen bir şart kontrol edilir. Bu şart doğruysa if komutunun altındaki ilk kod veya ilk kaşlı ayraç ({ }) arasındaki kodlar işlenir. Şart doğru değilse bu komutlar işlenmez. Şartın doğru olmadığı durumlarda else komutunun altındaki ilk kod veya ilk kaşlı ayraç ({ }) arasındaki kodlar işlenir. Kontrol edilecek şart sayısı ikiden fazla ise if, else if komutları kullanılır.

```
if (şart) // şart doğruysa      if (şart) // şart yanlışsa
..... // kod işlenir.        ..... // kod atlanır.
```

```
if (şart) // şart doğruysa      if (şart) // şart yanlışsa
{
  ..... // kod işlenir.        ..... // kod atlanır.
  ..... // kod işlenir.        ..... // kod atlanır.
}
```

Programda if komutuyla butonun basılı olma durumundaki, else komutuyla basılı olmama durumundaki LED'lerin yanıp sönmeye şekli belirlenir.



Programda **for (int i=5;i<=7;i++)** döngüsü sayesinde aynı kodları tekrar yazmaya gerek kalmaz. for döngüsünün kullanımı şu şekildedir:

- **int i=5** döngünün başlangıç değeridir.
- **i<=7** döngünün koşuludur. i'nin değerinin 7 ve 7'den küçük olduğu her durumda for döngüsünün altındaki kodlar işlenir.
- **i++** sayaç değeridir. Döngüdeki kodların hepsi işlendiğinde sayaç değerindeki işlem yapılır ve değişkenin yeni değeri belirlenir. Burada i'nin değeri her seferinde bir artırılmıştır.

3. Adım: Kodların yazım işleminden sonra hata kontrolü yapınız. Kod yazımında hata yoksa programı mikrodenetleyiciye yükleyerek devrenin çalışmasını test ediniz.



Sıra Sizde

Görsel 3.35'teki devreyi kurunuz. Devredeki butonun konumuna göre LED'lerin yanıp sönmeye animasyonları farklı olacak şekilde kodları IDE programına yazınız. Programı mikrodenetleyicinize yükleyerek devrenin çalışmasını gözlemleyiniz.

```
int buton=11;
void setup ()
{
  pinMode(5 , OUTPUT);
  pinMode(6 , OUTPUT);
  pinMode(7 , OUTPUT);
  pinMode(11 , INPUT);
}
void loop ()
{
  if ( digitalRead (buton) == 1)
  {
    for ( int i=5; i<=7; i++)
    {
      digitalWrite( i , HIGH);
      delay(1000);
      digitalWrite( i , LOW);
      delay(1000);
    }
  }
  else
  {
    for ( int i=5; i<=7; i++)
    digitalWrite( i , HIGH);
    delay(1000);
    for ( int i=5; i<=7; i++)
    digitalWrite( i , LOW);
    delay(1000);
  }
}
```

Görsel 3.36: Kod satırları

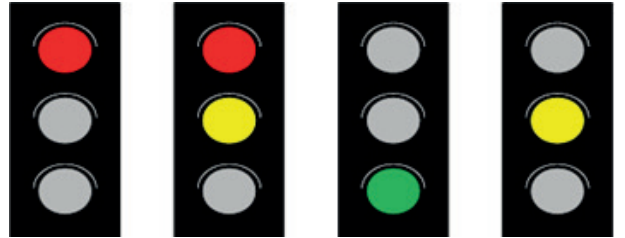


5. Uygulama

İşlem adımlarına göre bir yolda araç ve yaya geçişini düzenleyen trafik lambası uygulamasını yapınız. Görsel 3.37'de araç için trafik lambalarının çalışma sıralaması verilmiştir.

Gerekli Devre Elemanları

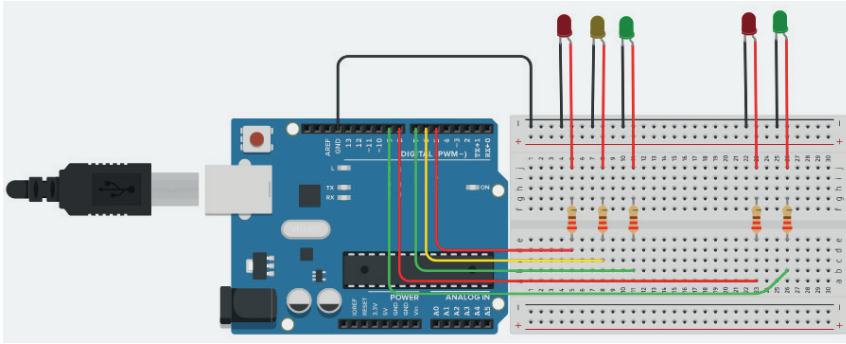
- 2 adet kırmızı LED
- 2 adet yeşil LED
- 1 adet sarı LED
- 5 adet 220 Ω (Kırmızı, Kırmızı, Kahverengi, Altın) direnç
- 1 adet mikrodenetleyici
- 1 adet breadboard
- Bağlantı kabloları



Görsel 3.37: Trafik lambalarının çalışma sıralaması



1. Adım: Görsel 3.38'deki bağlantıyı kurunuz. Araç ve yayalar için kullanılan LED'leri anlaşılır olması için birbirinden ayrı gruplandırınız.



Görsel 3.38: Trafik lambası uygulama devresi

2. Adım: Araç ve yaya geçişini düzenleyen trafik lambası uygulama devresinin kodları Görsel 3.39'da verilmiştir. Kodları IDE programında yazınız.

3. Adım: Kodların yazım işleminden sonra hata kontrolü yapınız. Kod yazımında hata yoksa programı mikrodenetleyiciye yükleyerek devrenin çalışmasını test ediniz.



Sıra Sizde

Dört kollu kavşak noktasındaki trafik lambası uygulama devresini kurarak kodlarını IDE programına yazınız. Programı mikrodenetleyicinize yükleyerek devrenin çalışmasını gözlemleyiniz.

```
int ak=5; // ak (araç kırmızı)
int as=6; // as (araç sarı)
int ay=7; // ay (araç yeşil)
int yk=8; // yk (yaya kırmızı)
int yy=9; // yy (yaya yeşil)
void setup ( )
{
  pinMode(ak , OUTPUT);
  pinMode(as , OUTPUT);
  pinMode(ay , OUTPUT);
  pinMode(yk , OUTPUT);
  pinMode(yy , OUTPUT);
}
void loop ( )
{
  digitalWrite( ak , HIGH);
  digitalWrite( as , LOW);
  digitalWrite( ay , LOW);
  digitalWrite( yk , HIGH);
  digitalWrite( yy , LOW);
  delay(2000);
  digitalWrite( yk , LOW);
  digitalWrite( yy , HIGH);
  delay(6000);
  digitalWrite( yk , HIGH);
  digitalWrite( yy , LOW);
  delay(1000);
  digitalWrite( as , HIGH);
  delay(2000);
  digitalWrite( ak , LOW);
  digitalWrite( as , LOW);
  digitalWrite( ay , HIGH);
  delay(8000);
  digitalWrite( as , HIGH);
  digitalWrite( ay , LOW);
  delay(2000);
}
```

Görsel 3.39: Kod satırları



3.2. BİYOMEDİKAL SENSÖR UYGULAMALARI

Günümüzde kullanılan tıbbi cihazlar ile yapılan ölçümlerde birçok farklı sensör kullanılır. Sensörler, bulunduğu ortamdaki değişiklikleri (ısı, ışık, ses, basınç, manyetik vb.) algılayıp bu değişikliklere göre dijital veya analog değer üreten elemanlardır (Görsel 3.40).



Görsel 3.40: Sensör çeşitleri



Araştırma

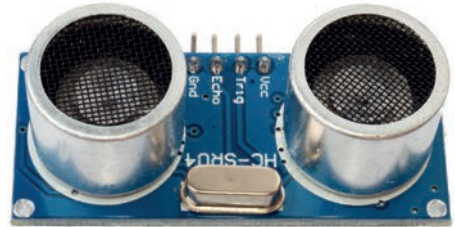
Tıbbi cihazlarda kullanılan sensör çeşitlerini araştırınız ve araştırma sonucunuzu bir sunum olarak hazırlayınız.

3.2.1. Mesafe Sensörü Uygulaması

Elektronik devreler ve tıbbi cihazlarda mesafe ölçmek ve engel algılamak için mesafe sensörleri kullanılır. Mesafe sensörlerinin birçok çeşidi bulunur. Bu bölümde HC-SR04 ultrasonik mesafe sensörü kullanılacaktır.

Mesafe sensörünün üzerinde bir verici ve bir alıcı olmak üzere iki hoparlör bulunur. Verici hoparlörden insan kulağının duyamayacağı frekansta (40 KHz) ses dalgaları yayımlanır. Bu ses dalgalarının bir yüzeye çarpıp alıcı hoparlöre geri gelmesi arasında geçen zamana göre mesafe ölçümü yapılabilir.

Mesafe sensörü 2 ile 400 cm arası ölçüm yapabilir. Besleme gerilimi 5 voltur. Mesafe sensörünün Vcc, Trig, Echo ve Gnd olmak üzere dört bacağı bulunur. Vcc (+) ve Gnd (-) besleme bacaklarıdır (Görsel 3.41). Trig bacağına enerji uygulanarak aktif hâle getirildiğinde verici hoparlörden ses dalgası yayımlanır. Yayımlanan bu ses dalgaları bir yüzeyden yansıyarak alıcı hoparlöre geri geldiğinde Echo bacağına gerilim uygulanır. Ses dalgalarının yayımlanıp, bir yüzeye çarptıktan sonra geri gelmesi arasında geçen süre kullanılarak sensörün önünde bulunan engelin mesafesi hesaplanır.



Görsel 3.41: Ultrasonik mesafe sensörü

Not

Ses dalgalarının yaklaşık hızı saniyede 340 m/sn.dir.



Araştırma

İnsan kulağının duyabileceği ses frekans aralıklarını araştırınız ve araştırma sonucunuzu bir sunum olarak hazırlayınız.



6. Uygulama

İşlem adımlarına göre devre şemasını devre tasarım ve simülasyon programında çizerek mesafe sensörü kullanımını sağlayan bir çalışma gerçekleştirebilirsiniz.

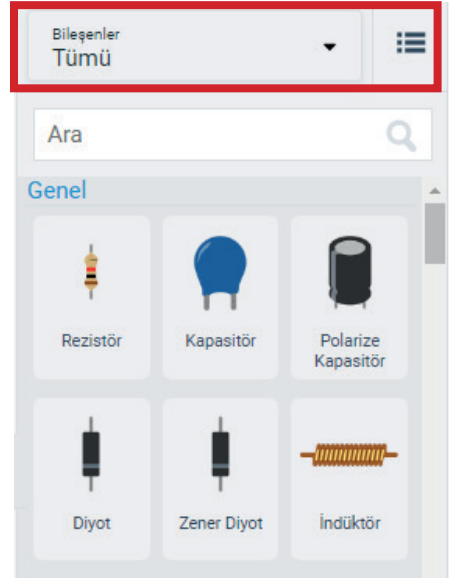
1. Adım: İnternet tarayıcınızın adres satırına <https://www.tinkercad.com/> yazıp üyelik bilgileriniz ile siteye giriş yapınız.

2. Adım: Circuits ekranından **Yeni Devre oluştur** seçeneğine tıklayınız.

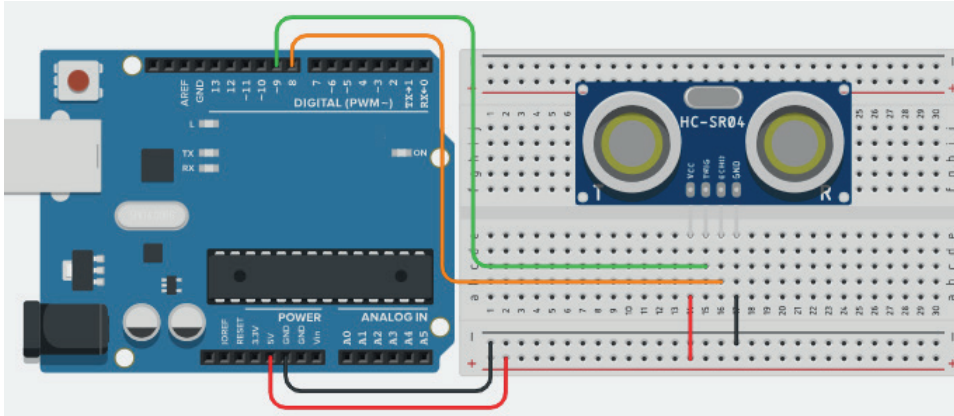
3. Adım: Görsel 3.42'deki **Bileşenler** panelinden gerekli devre elemanlarını tasarım alanına sürükleyerek alınız.

4. Adım: Görsel 3.43'te gösterildiği gibi devrenin bağlantısını yapınız.

Devre kurulumu yaparken mesafe sensörünün Vcc pinini mikrodenetleyicinin 5 V, GND pinini mikrodenetleyicinin GND, Echo pinini mikrodenetleyicinin D8, Trig pinini mikrodenetleyicinin D9 portuna bağlayınız.



Görsel 3.42: Bileşenler paneli



Görsel 3.43: Mesafe sensörü uygulama devresi

5. Adım: Devre tasarım ve simülasyon programındaki Kod sekmesinin altında yer alan Metin seçeneğine Görsel 3.44'teki kodları yazınız.

Not

Program kodları incelendiğinde “digitalWrite (trig, HIGH), delay (1) ve digitalWrite (trig, LOW)” fonksiyonları ile 1 milisaniye süresince ses dalgası üretilir.

Üretilen bu ses dalgası engеле çarpıp yansiyarak mesafe sensörüne geri döner. Bu süreçte geçen zaman, pulseIn fonksiyonu ile hesaplanır ve “sure” değişkenine aktarılır. Engel ile mesafe sensörü arasındaki uzaklık bulunurken sesin geliş gidiş zamanından dolayı sürenin yarısı alınır, sesin saniyedeki hızı ile çarpılarak mesafe hesaplanır ve “mesafe” değişkenine aktarılır.

```
int   trig=9;
int   echo=8;
void  setup ( )
{
  pinMode(trig , OUTPUT);
  pinMode(echo , INPUT);
  Serial.begin(9600);
}
void  loop ( )
{
  digitalWrite(trig , HIGH);
  delay(1);
  digitalWrite(trig , LOW);
  int   sure = pulseIn(echo , HIGH);
  int   mesafe = (sure / 2)*0.034;
  Serial.println(mesafe);
  delay(100);
}
```

Görsel 3.44: Kod satırları



3. Adım: Simülasyonu Başlat seçeneği ile devre çalıştırıldığında mesafe ölçümünün yapıldığını ve sonuçların seri port ekranında gösterildiğini gözlemleyiniz.

Programın seri port çıktısı Görsel 3.45'te görülür.

```

1  byte echo=8;
2  byte trig=9;
3  void setup() {
4  Serial.begin(9600);
5  pinMode(trig,OUTPUT);
6  pinMode(echo,INPUT);
7  }
8  void loop() {
9  digitalWrite(trig,HIGH);
10 delay(100);
11 digitalWrite(trig,LOW);
12 int sure=pulseIn(echo,HIGH);
13 int mesafe=(sure/2)*0.034;
14 Serial.print("Mesafe=");
15 Serial.println(mesafe);
16 delay(1);
17 }

```

Seri Monitör

```

Mesafe=16
Mesafe=15
Mesafe=13
Mesafe=13
Mesafe=13
Mesafe=13
Mesafe=13
Mesafe=13
Mesafe=15
Mesafe=17
Mesafe=18
Mesafe=20
Mesafe=23
Mesafe=25
Mesafe=27
Mesafe=27
Mesafe=29

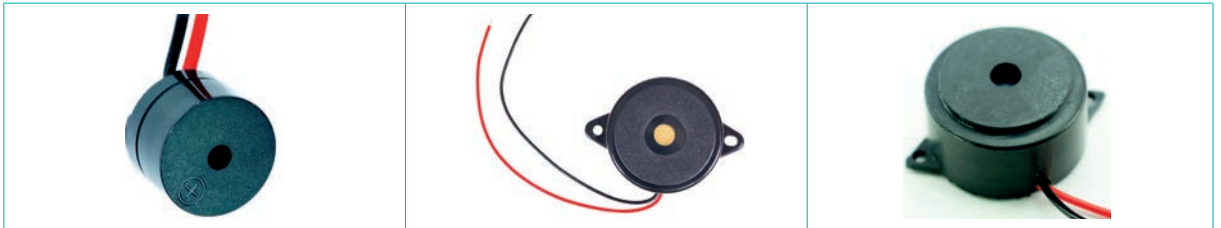
```

Görsel 3.45: Mesafe sensörü uygulamasının ekran çıktısı

3.2.2. Buzzer Uygulaması

Buzzer devre elemanı ufak bir hoparlör olarak düşünülebilir. Buzzerdan hoparlör gibi yüksek ve kaliteli bir ses elde edilemez. Buzzer ancak “bip” sesini çıkarabilir.

Buzzer bağlantısı yapılırken (+) ucu mikrodenetleyicinin dijital portlarından birine, (-) ucu GND portuna bağlandığında buzzerden ses elde edilir (Görsel 3.46).



Görsel 3.46: Buzzer çeşitleri



7. Uygulama

İşlem adımlarına göre buzzerın 1 saniye aralıklarla çalışmasını sağlayan uygulamayı yapınız.

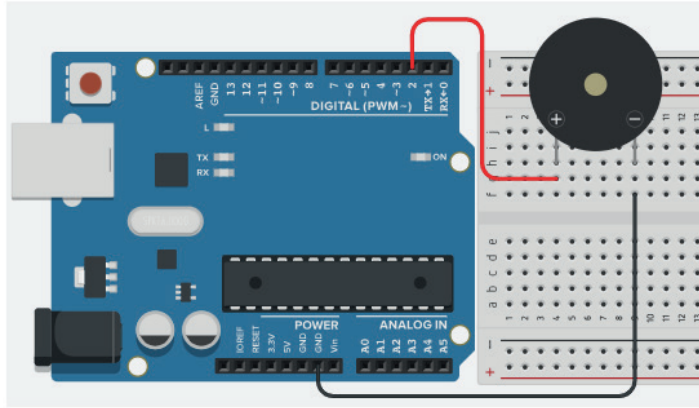
Gerekli Devre Elemanları

- 1 adet mikrodenetleyici kart
- 1 adet breadboard
- 1 adet buzzer
- Bağlantı kabloları

1. Adım: Buzzerın (-) pinini mikrodenetleyicinin GND, (+) pinini ise mikrodenetleyicinin D2 portuna Görsel 3.47’de gösterildiği gibi bağlayınız.

Not

Buzzerın bağlantıları dikkatli yapılmalıdır. Ters bağlanırsa buzzer ses çıkarmaz.



Görsel 3.47: Buzzer uygulama devresi

2. Adım: Mikrodenetleyici IDE programını açarak Görsel 3.48’deki kod satırlarını yazınız. Buzzerın çalma süresini “delay” fonksiyonu ile ayarlayınız.

3. Adım: Kodların yazım işlemi bittikten sonra hata kontrolünü yapınız. Kodlarda hata yoksa programı mikrodenetleyici karta yükleyerek devrenin çalışmasını test ediniz.



Sıra Sizde

Yedinci uygulamadaki buzzerı mikrodenetleyicinin 7 No.lu portuna bağlayarak breadboard üzerinde kurunuz. Buzzerdan 3 saniye ses gelecek, 2 saniye ses gelmeyecek şekilde IDE programında kodları yazıp, mikrodenetleyici karta yükleyerek devrenin çalışmasını test ediniz.

```
int buzzer=2;
void setup ()
{
  pinMode(buzzer , OUTPUT);
}
void loop ()
{
  digitalWrite(buzzer , HIGH);
  delay(1000);
  digitalWrite(buzzer , LOW);
  delay(1000);
}
```

Görsel 3.48: Kod satırları



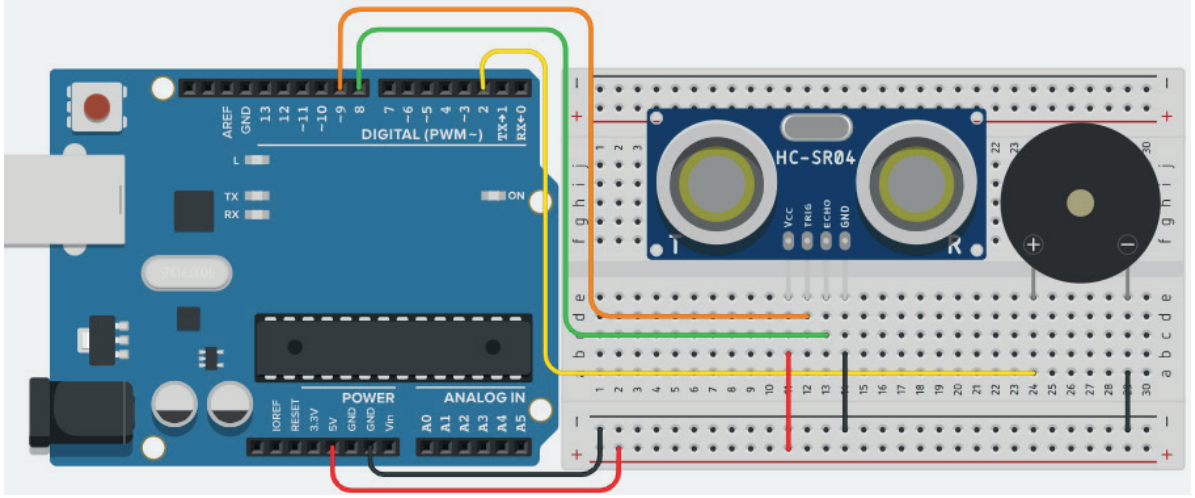
8. Uygulama

İşlem adımlarına göre mesafe sensörü ve buzzerın birlikte kullanımını sağlayarak park sensörü uygulamasını yapınız.

Gerekli Devre Elemanları

- 1 adet mikrodenetleyici kart
- 1 adet breadboard
- 1 adet mesafe sensörü
- 1 adet buzzer
- Bağlantı kabloları

1. Adım: Görsel 3.49'da gösterildiği gibi devre bağlantısını yapınız.



Görsel 3.49: Park sensörü uygulama devresi



Araştırma

Araçlarda kullanılan park sensörlerinin çalışmasını araştırınız. Edindiğiniz bilgileri sınıfta arkadaşlarınızla paylaşınız.

Not

Mesafe belirlendikten sonra “if - else” koşul yapısı ile uygun şart belirlenir ve buzzerdan ilgili uyarı tonu alınır.



2. Adım: Mikrodenetleyici IDE programını açarak Görsel 3.50'deki kod satırlarını yazınız.

```
int trig=9;
int echo=8;
int buzzer=2;
void setup ( )
{
  pinMode(trig , OUTPUT);
  pinMode(echo , INPUT);
  pinMode(buzzer , OUTPUT);
  Serial.begin(9600);
}
void loop ( )
{
  digitalWrite(trig , HIGH);
  delay(1);
  digitalWrite(trig , LOW);
  int sure = pulseIn(echo , HIGH);
  int mesafe = (sure / 2)*0.034;
  Serial.println(mesafe);
  if (mesafe>=50)
  {
    digitalWrite(buzzer , HIGH);
    delay(2000);
    digitalWrite(buzzer , LOW);
    delay(2000);
  }
  else if (mesafe>=40)
  {
    digitalWrite(buzzer , HIGH);
    delay(1500);
    digitalWrite(buzzer , LOW);
    delay(1500);
  }
  else if (mesafe>=30)
  {
    digitalWrite(buzzer , HIGH);
    delay(1000);
    digitalWrite(buzzer , LOW);
    delay(1000);
  }
  else if (mesafe>=20)
  {
    digitalWrite(buzzer , HIGH);
    delay(500);
    digitalWrite(buzzer , LOW);
    delay(500);
  }
  else if (mesafe>=10)
  {
    digitalWrite(buzzer , HIGH);
    delay(100);
    digitalWrite(buzzer , LOW);
    delay(100);
  }
}
```

Görsel 3.50: Kod satırları



3. Adım: Kodların yazım işlemi bittikten sonra hata kontrolünü yapınız. Kodlarda hata yoksa programı mikrodenetleyici karta yükleyerek devrenin çalışmasını test ediniz.



Sıra Sizde

Sekizinci uygulamada mesafe ölçümünü gösteren sensöre 5 adet LED ilave ederek devreyi breadboard üzerinde kurunuz. Engele 50 cm kaldığında 1 LED, 40 cm kaldığında 2 LED, 30 cm kaldığında 3 LED, 20 cm kaldığında 4 LED ve 10 cm kaldığında 5 LED yanacak şekilde IDE programında kodları yazıp, mikrodenetleyiciye yükleyerek devrenin çalışmasını test ediniz.

3.2.3. Nabız Sensörü Uygulaması

Nabız sensörü, medikal sensörlerden biridir ve parmak ucundan nabız ölçümü yapılmasını sağlar. Nabız sensörü, LED ve ışık sensörünün birleşiminden oluşmuştur. Nabız sensörü, parmaktan yansıyan ışık şiddetini analog sinyale çevirir (Görsel 3.51). Kalbin her atışında damarlardaki kan basıncı değiştiği için parmaktan yansıyan ışık miktarı ve sensörün üreteceği değer değişir.



Görsel 3.51: Nabız sensörü

Besleme gerilimi 5 voltur. (+), (-) ve S (Sinyal) olmak üzere üç bacağı bulunur. (+) bacağı mikrodenetleyicinin 5 V pinine, (-) bacağı mikrodenetleyicinin GND pinine ve S bacağı ise mikrodenetleyicinin analog giriş (A0-A5) portlarından birine bağlanır. Nabız sensörü, nabız atışına göre sinyal pini üzerinden 0 ile 1023 arasında değer üretir. Mikrodenetleyiciye yazılan kod ile bu değerler kullanılarak dakikadaki nabız sayısı ölçülür.



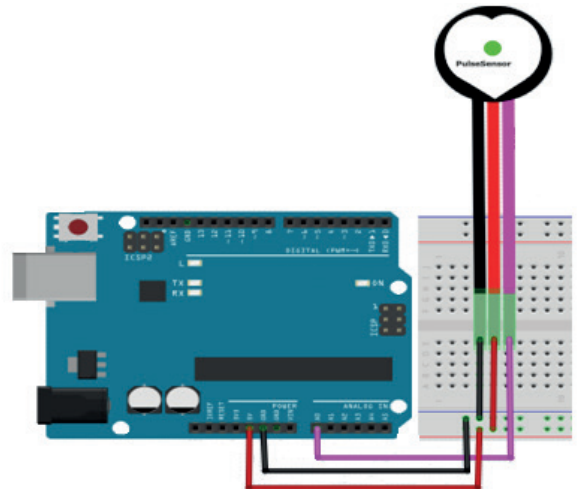
9. Uygulama

İşlem adımlarına göre nabız sensörü uygulamasını yapınız.

Gerekli Devre Elemanları

- 1 adet mikrodenetleyici kart
- 1 adet breadboard
- 1 adet nabız sensörü
- Bağlantı kabloları

1. Adım: Görsel 3.52'de gösterildiği gibi devre bağlantısını yapınız.



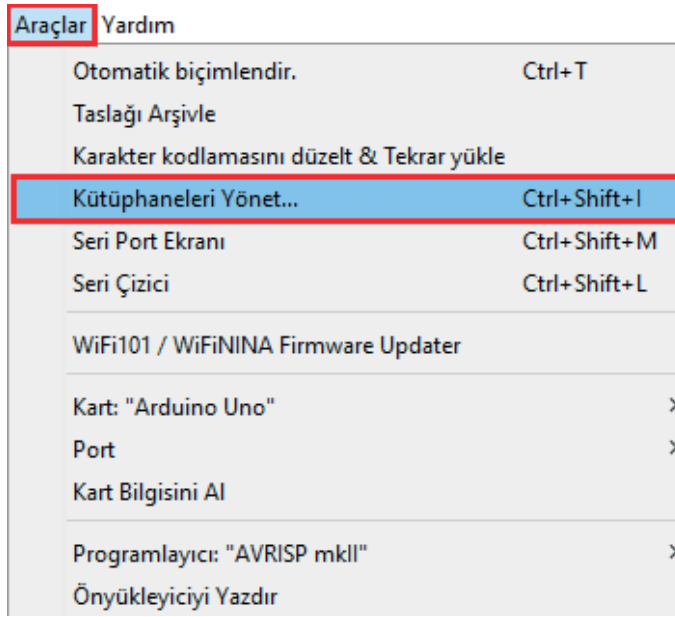
Görsel 3.52: Nabız sensörü uygulama devresi

Not

Mikrodenetleyici IDE programında kod yazılırken nabız sensörünü kullanabilmek için yapılması gereken ilk iş, sensör kütüphanesinin kurulmasıdır.

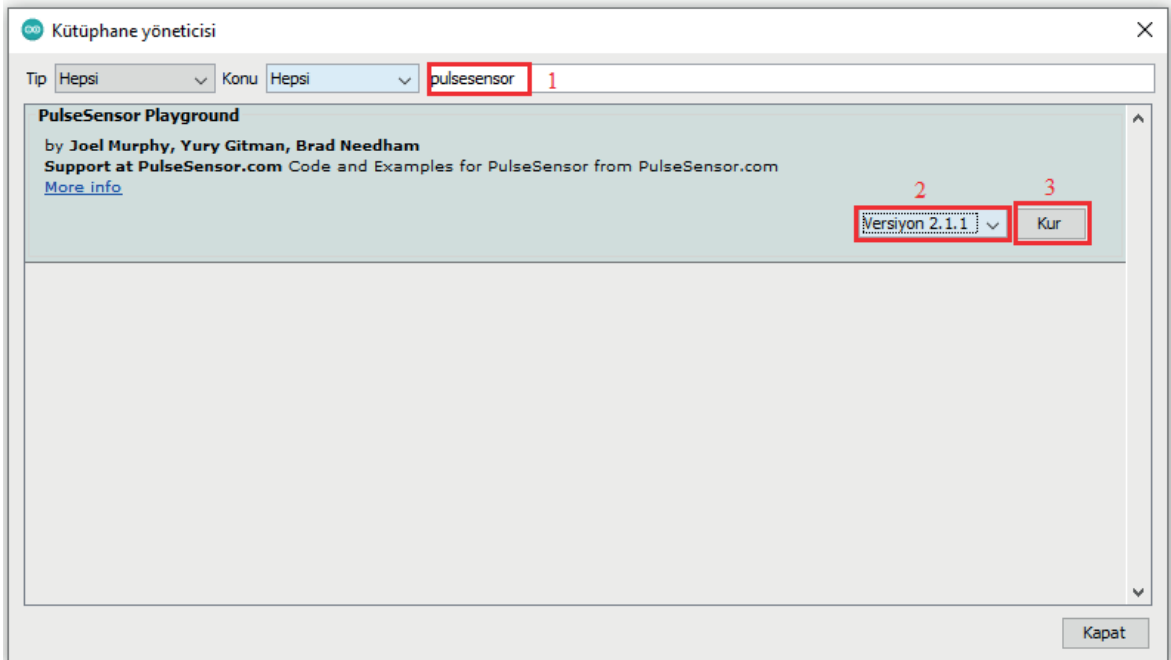


2. Adım: Görsel 3.53'te gösterildiği gibi IDE programından **Araçlar** menüsü içinde **Kütüphaneleri Yönet** seçeneğine tıklayınız.



Görsel 3.53: Kütüphaneleri Yönet penceresi

3. Adım: Görsel 3.54'te 1 numaralı alana pulsesensor yazınız. 2 numaralı alandan en güncel versiyonu seçiniz ve ardından 3 numaralı alandan Kur butonuna basarak kurulumu tamamlayınız.



Görsel 3.54: Kütüphane yöneticisi penceresinde sensör kütüphanesinin aranması



4. Adım: Mikrodenetleyici IDE programını açarak Görsel 3.55'teki kod satırlarını yazınız.

Not

Hesaplamalarda kullanılmak üzere bir eşik değeri belirlenir. Kod satırları incelendiğinde bu değerin 550 olarak kabul edildiği görülür ancak denemeler sonucunda bu değerin değiştirilmesi gerekebilir. if döngüsü ile nabız değerinin olup olmadığı kontrol edilir. Nabız değeri varsa seri monitöre yazdırılır.

```
#define USE_ARDUINO_INTERRUPTS true
#include "PulseSensorPlayground.h"
const int pulse = A0;
int esik_degeri = 550;
PulseSensorPlayground pulseSensor;
void setup ()
{
  Serial.begin (9600);
  pulseSensor.analogInput (pulse);
  pulseSensor.setThreshold (esik_degeri);
  pulseSensor.begin ();
}
void loop ()
{
  int nabiz = pulseSensor.getBeatsPerMinute ();
  if ( pulseSensor.sawStartOfBeat ( ) ) {
    Serial.println( "Nabızınız ");
    Serial.print( "BPM: ");
    Serial.println( nabiz );
  }
  delay(100);
}
```

Görsel 3.55: Kod satırları

5. Adım: Kodların yazım işlemi bittikten sonra hata kontrolünü yapınız. Kodlarda hata yoksa programı mikrodenetleyici karta yükleyerek devrenin çalışmasını test ediniz.



Araştırma

İnsanlarda olması gereken nabız değerlerini araştırınız ve araştırma sonucunuzu bir sunum olarak hazırlayınız.



Sıra Sizde

Normal bir insanda olması gereken değerler dışında bir nabız ölçümü yapılması sonucunda dokuzuncu uygulamaya bir kırmızı LED ve buzzer ekleyerek, uyarı vermesini sağlayan devreyi tasarlayıp breadboard üzerinde kurunuz. Ayrıca IDE programında kodları yazıp, mikrodenetleyiciye yükleyerek devrenin çalışmasını test ediniz.



3.3. BİYOMEDİKAL CİHAZ MODÜLLERİNDEKİ UYGULAMALAR

Biyomedikal cihaz teknolojilerindeki uygulamalar; sağlık sektöründe hastaların teşhis, tedavi ve takibinde kullanılmak üzere çeşitli teknolojilerin tasarımı ve geliştirilmesi için olanak sağlar.

3.3.1. Işık Etkili Direnç Sensörü Uygulaması

Işık etkili direnç, İngilizce **Light Dependent Resistor** kelimelerinin baş harflerinin birleşmesinden oluşan **LDR** ile ifade edilir. LDR'nin diğer adı fotodirençtir. LDR'ler, üzerine düşen ışık şiddeti ile ters orantılı olarak direnç değerleri değişen devre elemanlarıdır. Işık şiddeti arttıkça direnç değeri azalır, ışık şiddeti azaldığında ise direnç değeri artar. Görsel 3.56'da LDR görülür.



Görsel 3.56: LDR

Biyomedikal cihazlarda ortam ışığını algılama ve ışık şiddeti değerlerindeki değişime göre tetkik ve teşhis süreçlerinde LDR'lerin kullanıldığı alanlar bulunur. Bu cihazlara nabız sensörleri örnek verilebilir.



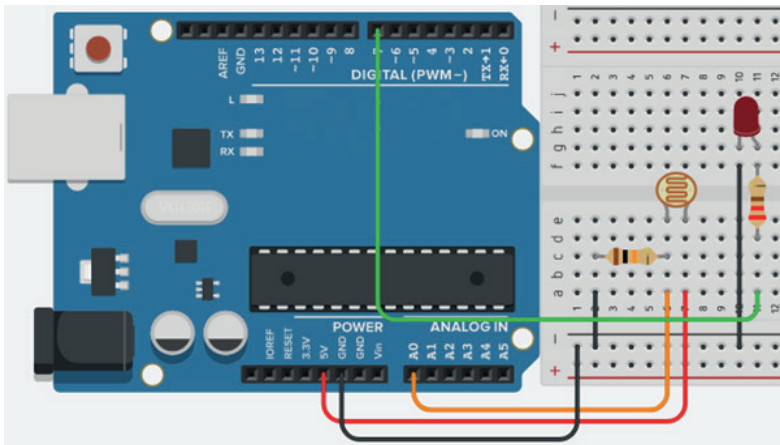
10. Uygulama

İşlem adımlarına göre LDR'den gelen değere uygun şekilde LED'i açıp kapatma uygulamasını yapınız.

Gerekli Devre Elemanları

- 1 adet mikrodenetleyici kart
- 1 adet breadboard
- 1 adet LDR
- 1 adet LED
- 1 adet 220 Ω (Kırmızı, Kırmızı, Kahverengi, Altın) direnç
- 1 adet 10 K Ω (Kahverengi, Siyah, Turuncu, Altın) direnç
- Bağlantı kabloları

1. Adım: Görsel 3.57'de gösterildiği gibi devre şemasını devre tasarım ve simülasyon programı üzerinde çözümleniz.



Görsel 3.57: LDR








ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerde parantezlerin içine yargılar doğru ise "D", yanlış ise "Y" yazınız.

1. () Elektrik akımına karşı gösterilen zorluğa direnç denir.
2. () Gerilimin birimi amperdir.
3. () LED bacaklarının bağlantı yönü önemli değildir.
4. () Programda bekleme yapmak için delay komutu kullanılır.
5. () LM35, analog sinyal üretir.

B) Aşağıdaki soruları dikkatlice okuyarak doğru seçeneği işaretleyiniz.

6. Aşağıdakilerden hangisi LED devre elemanının sembolüdür?

- A) 
- B) 
- C) 
- D) 
- E) 

7. Aşağıdakilerden hangisi 0,23 kA'dır?

- A) 0,00023 A
- B) 230 A
- C) 23 A
- D) 230000 A
- E) 2300 A

8. Üzerinde turuncu-turuncu-kahverengi-altın renkleri olan direncin değeri kaç ohmdur?

- A) 22 Ω
- B) 33 Ω
- C) 220 Ω
- D) 330 Ω
- E) 440 Ω

9. Basit bir elektrik devresinde yükün değeri 20Ω , yük üzerinden geçen akım 3A olduğuna göre üretcin değeri kaç volttur?
- A) 17 V
B) 23 V
C) 40 V
D) 50 V
E) 60 V
10. Aşağıdaki direnç çeşitlerinden hangisinin değeri ışık ile ayarlanabilir?
- A) POT
B) LDR
C) NTC
D) PTC
E) LM35
11. Aşağıdaki fonksiyonlardan hangisi 3 numaralı dijital pini çıkış olarak ayarlar?
- A) digitalRead (3);
B) analogRead (3);
C) pinMode (3, OUTPUT);
D) delay (3);
E) pinMode (3, INPUT);
12. Aşağıdakilerden hangisi mikrodenetleyici IDE programında A0 numaralı analog pindeki değeri okuyan fonksiyondur?
- A) pinMode (A0);
B) digitalRead (A0);
C) analogRead (A0);
D) delay (A0);
E) analogWrite (A0);
13. Aşağıdakilerden hangisi programın 2 saniye beklemesi için kullanılan fonksiyondur?
- A) delay (2);
B) delay (20);
C) delay (200);
D) delay (2000);
E) delay (20000);
14. Aşağıdakilerden hangisi mikrodenetleyici IDE programında 5 numaralı digital pindeki değeri okuyan fonksiyondur?
- A) Analogread (5);
B) digitalRead (5);
C) analogRead (5);
D) digitalWrite (5);
E) analogWrite (5);

15. Aşağıdakilerden hangisi mikrodenetleyici IDE programında pinlerin giriş veya çıkış olarak ayarlandığı fonksiyondur?

- A) digitalWrite ()
- B) if ()
- C) void loop ()
- D) for ()
- E) void setup ()

16. Aşağıdakilerden hangisi mikrodenetleyici IDE programında değişken tanımlamaya yarayan fonksiyondur?

- A) delay (3000);
- B) analogRead (5);
- C) int robotik;
- D) pinMode (6 , INPUT);
- E) digitalRead (5);

17. Yanda verilen kod satırlarındaki hangi fonksiyon 5 numaralı pine bağlı LED'in ışık yaymasını sağlar?

- A) digitalWrite (LED_1 , HIGH);
- B) digitalWrite (LED_2 , HIGH);
- C) int LED_1=5;
- D) digitalWrite (LED_1 , LOW);
- E) pinMode (LED_1 , OUTPUT);

```
int LED_1=5;
int LED_2=6;
void setup ( )
{
  pinMode (LED_1 , OUTPUT);
  pinMode (LED_2 , OUTPUT);
}
void loop ( )
{
  digitalWrite (LED_1 , HIGH);
  digitalWrite (LED_2 , LOW);
  delay (1000);
  digitalWrite (LED_1 , LOW);
  digitalWrite (LED_2 , HIGH);
  delay (1000);
}
```

18. Aşağıdakilerden hangisi mikrodenetleyici IDE programında iki defa döngü sağlayan fonksiyondur?

- A) if (buton==1)
- B) for (int i=10; i<=11; i++)
- C) for (int i=8; i<=4; i--)
- D) pinMode (2, INPUT)
- E) else

19. Mikrodenetleyici uygulama kartı ile LED yakıp söndürülecektir. LED'in katot bacağı, uygulama kartının hangi pinine bağlanmalıdır?

- A) GND
- B) 5 V
- C) 8
- D) A0
- E) RESET

20. Aşağıdakilerden hangisi mikrodenetleyici IDE programında sonsuz döngü sağlayan fonksiyondur?

- A) digitalRead (8);
- B) else
- C) for ()
- D) void setup ()
- E) void loop ()

CEVAP ANAHTARLARI

1. ÖĞRENME BİRİMİ

A	1. C	2. B	3. A	4. A	5. B
	6. E	7. A	8. A	9. D	10. B
	11. A	12. B	13. C	14. B	15. E
	16. D	17. C	18. B	19. D	20. B

2. ÖĞRENME BİRİMİ

A	1. Doğru	2. Yanlış	3. Doğru	4. Yanlış	5. Yanlış
	6. Doğru	7. Doğru	8. Doğru		

B	9. A	10. E	11. C	12. D	13. A
---	------	-------	-------	-------	-------

3. ÖĞRENME BİRİMİ

A	1. Doğru	2. Yanlış	3. Yanlış	4. Doğru	5. Doğru
---	----------	-----------	-----------	----------	----------

B	6. C	7. B	8. D	9. E	10. B
	11. C	12. C	13. D	14. B	15. E
	16. C	17. A	18. B	19. A	20. E

KAYNAKÇA

- Biyomedikal Cihaz Teknolojileri Alanı Çerçeve Öğretim Programı, Ankara, 2020.
- TDK Türkçe Sözlük, Ankara, 2023.
- TDK Yazım Kılavuzu, Ankara, 2021.

Kaynakça TDK'ye göre düzenlenmiştir.

GENEL AĞ KAYNAKÇASI VE GÖRSEL KAYNAKÇASI

Materyalin genel ağ kaynakçasına ve görsel kaynakçasına buradan ulaşılır.
Karekoda ulaşılabilmesi durumunda aşağıdaki linki kullanabilirsiniz.



<http://kitap.eba.gov.tr/karekod/Kaynak.php?KOD=2683>

NOT