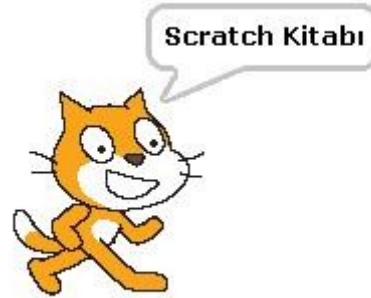


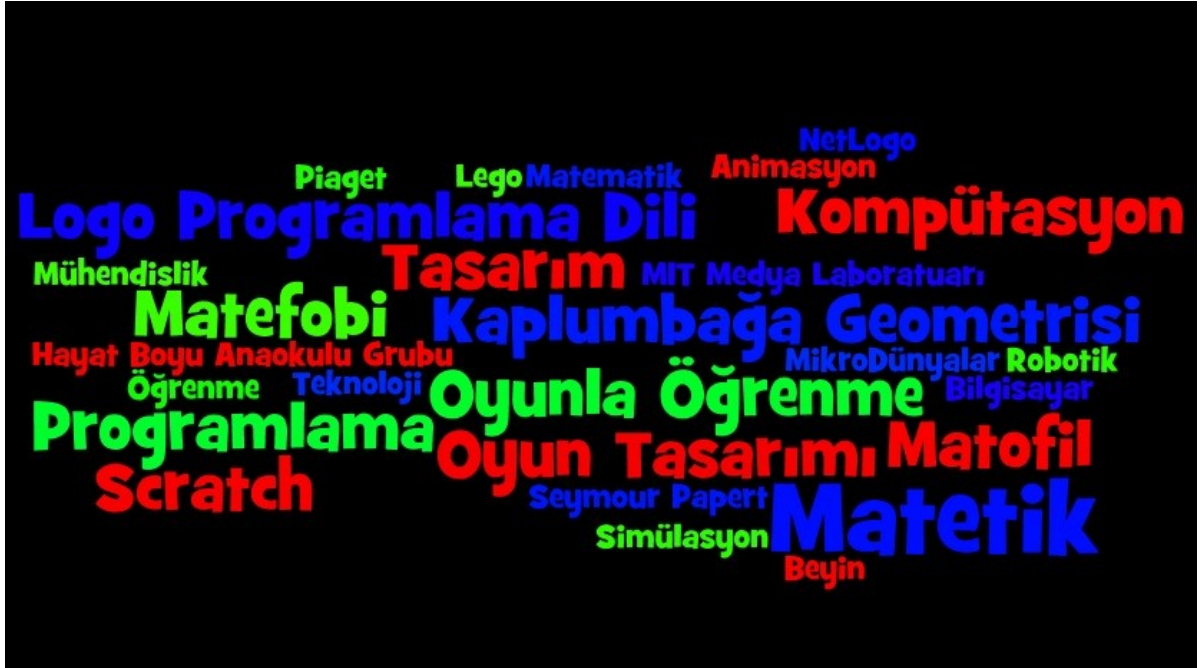
# SCRATCH





<http://scratchogren.blogspot.com/>





## Scratch İndir

<http://scratch.mit.edu/download>

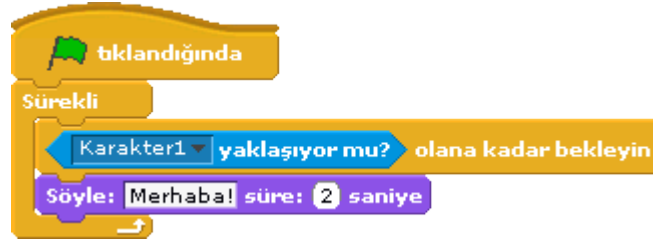
MIT Medya Laboratuvarında geliştirilen Scratch Logo programlama dilinin en gelişmiş örneklerinden birisidir. Scratch bir grafik programlama dilidir. Okuma yazma bilen ya da yeni öğrenmeye başlayan her çocuk Scratch'i kolaylıkla öğrenebilir. Scratch'i 40'dan fazla dilde kullanabilir ve yapılmış projeleri anında istediğimiz dile çevirebiliriz.

Scratch ile çocuklar teknolojiyi çok daha akıcı bir şekilde kullanmayı ve teknoloji ile yaratmayı öğreneceklerdir. Çocukların kullandığı multi-medya araçlarının çoğu daha önceden başkaları tarafından yapılır ve bir paket olarak çocukların önüne gelir. Çocuklar multi-medya ürünlerini sadece tüketirler.

Scratch çocukların kendi multi-medya tasarımlarını yapmalarını, karşılaştıkları gerçek hayat problemleri karşısında teknolojiyi kullanarak yaratıcı çözümler ortaya koymalarını ve kendilerini değişik şekillerde ifade ederek 21. yüzyıl becerilerini kazanmalarını sağlar.

Scratch ile bilgisayar programı yazmak için kodların bloklar halinde üst üste ve yan yana bir araya getirilmesi gerekir. Çek-bırak özelliği sayesinde Scratch ile programlama öğrenmek çok kolaydır.

Scratch ile resim, ses, müzik gibi çeşitli medya araçlarını bir araya getirebilir, kendi animasyonlarımızı, bilgisayar oyunlarımızı tasarlayabilir ya da interaktif hikayeler anlatabiliriz.



## Scratch:

**Alçak tabanlıdır:** Başlaması kolaydır.

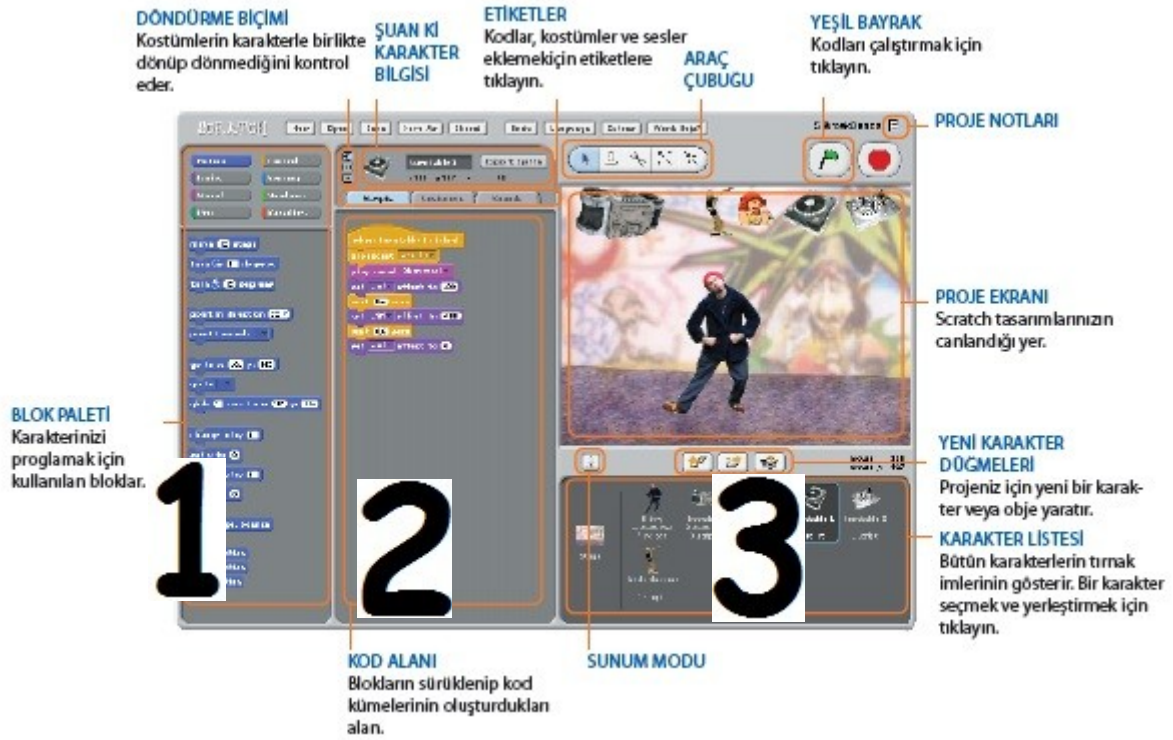
**Geniş duvarlıdır:** Çok çeşitli projeler yapılabilir.

**Yüksek tavanlıdır:** Basit bilgilerle karmaşık projeler yapılabilir.



Scratch'in gelişmiş ve kullanması kolay bir arayüzü vardır. Scratch arayüzü 3 ana bölmeden oluşur.

1. Blok Paletinde karakterleri programlamak için kullanılan bloklar vardır.
2. Kodlama alanı bilgisayar programını yazdığımız yerdir. Buraya blokları çekip bırakarak programımızı yazarız.
3. Proje ekranı bizim sahnemizdir. Tasarladığımız her şey Scratch'in sahnesinde hayat bulur.



**Değişkenler yaratabilir ve bu değişkenleri program içinde kullanabiliriz.**



**Sürekli ve Tekrar kodları ile belli komutlar dizisini tekrarlayabiliriz.**



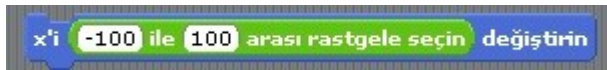
**Sor ve Bekle kodu ile klavye ile bilgi girişi yapılır. Yanıt kodu bilgiyi hafızaya alır.**



**Program yazmak için atacağımız adımları sistematik olarak düşünmemiz gerekir.**



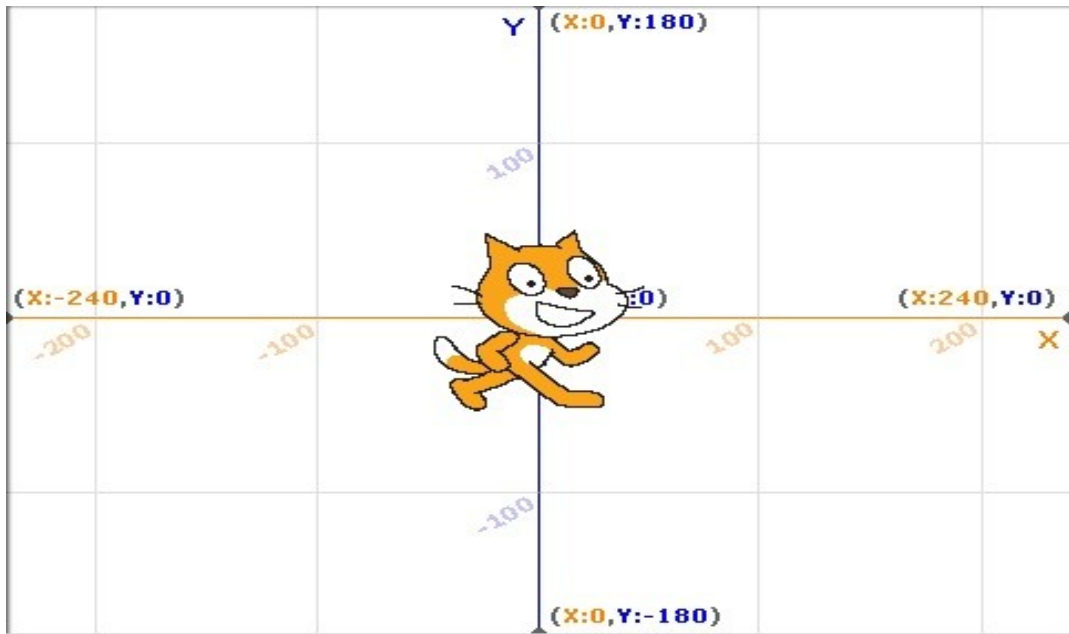
**Rastgele Sayılar ile belirlenen aralıktaki sayılardan rastgele seçilir.**



**Ve, Ya da, Değil Boolean Mantığına örnektir.**



Scratch sahnesi aslında bir koordinat düzlemidir. Bütün animasyonlarımız ve bilgisayar oyunlarımız bu koordinat düzlemi üzerinde tasarlanır. Scratch karakteri olan kedi program açıldığı zaman (0,0) noktasındadır. Ekranın üstünde karakterlerimizin x ve y değerlerini görebiliriz.



Çocuklar x ve y değerlerini 10'ar birim değiştirerek ekrandaki karakteri ok tuşları ile ileri-geri ya da aşağıya-yukarıya hareket ettirebilir ve koordinat sistemi hakkında kalıcı bilgi edinirler.





Çoęu insan için bilgisayar programlamak zor ve uzmanlık gerektiren bir iř olduęunu düşünür. JAVA ya da C++ gibi geleneksel programlama dilleri için bunlar doğrudur ve çoęu insan için bu dilleri öğrenmek zordur. Örneęin, JAVA dili ile yazılmıř ařaęıdaki program çalıştırıldıęı zaman bilgisayar ekranında "merhaba dünya" yazısının çıkmasını saęlar.

Scratch bilgisayarların daha güçlenmesi ve arayüzlerin gelişmesinden faydalanarak bütün bunları deęiřtirir ve bilgisayar programla becerisini ve bilgisini çok geniş kitlelere ve özellikle de çocuklara ulařtırır. Aynı programı JAVA yerine Scratch ile yazmak çok daha kolaydır. Boolean mantıęı da farklı deęildir.

```
class Hello  
{
```



# Boolean Expressions

Scratch v. C



`(x < y)`

`((x < y) && (y < z))`

## Hareket Et

Hareket bloğunun üzerine tıklayarak "10 adım gidin" bloğunu yazılar alanına çekip bırakın.



"10 adım gidin" bloğunun üzerine tıklayarak kediye hareket ettirebilirsiniz.



## Ses



Bateri bloğunu çekerek hareket blokunun altına yerleştirin.

Bloklara tıklayıp dinleyin. Eğer ses gelmiyorsa bilgisayarınızın sesinin açık olduğundan emin olun..

Bateri blokunun içindeki oku tıklayarak değişik davullar çalabilirsiniz.



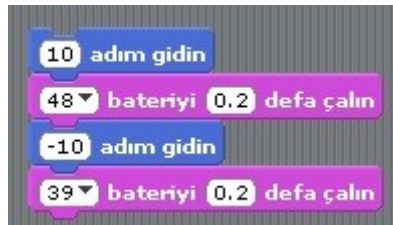
## Dans Edin

Bir hareket bloğunu daha çekerek Bateri blokunun altına yerleştirin. Hareket bloğunun içine tıklayarak (-) eksi işareti yerleştirin ya da (-10) yazın.



Herhangi bir bloku tıklayarak blok kümenizi çalıştırabilirsiniz.

Kümenize bir Bateri bloku daha ekleyerek farklı bir davul seçin.



## Tekrar ve Tekrar

Kontrol panelini açın.



Kontrol Panelinden Sürekli bloğunu alarak kümenizi içine yerleştirin. Kümenizi tıklayarak Sürekli bloğunun içinden çıkartabilirsiniz. Herhangi bir bloku tıklayarak blok kümenizi çalıştırabilirsiniz.



## Yeşil Bayrak

Kontrol Panelinden



blokunu alarak blok kümenizin en üstüne yerleştirin.

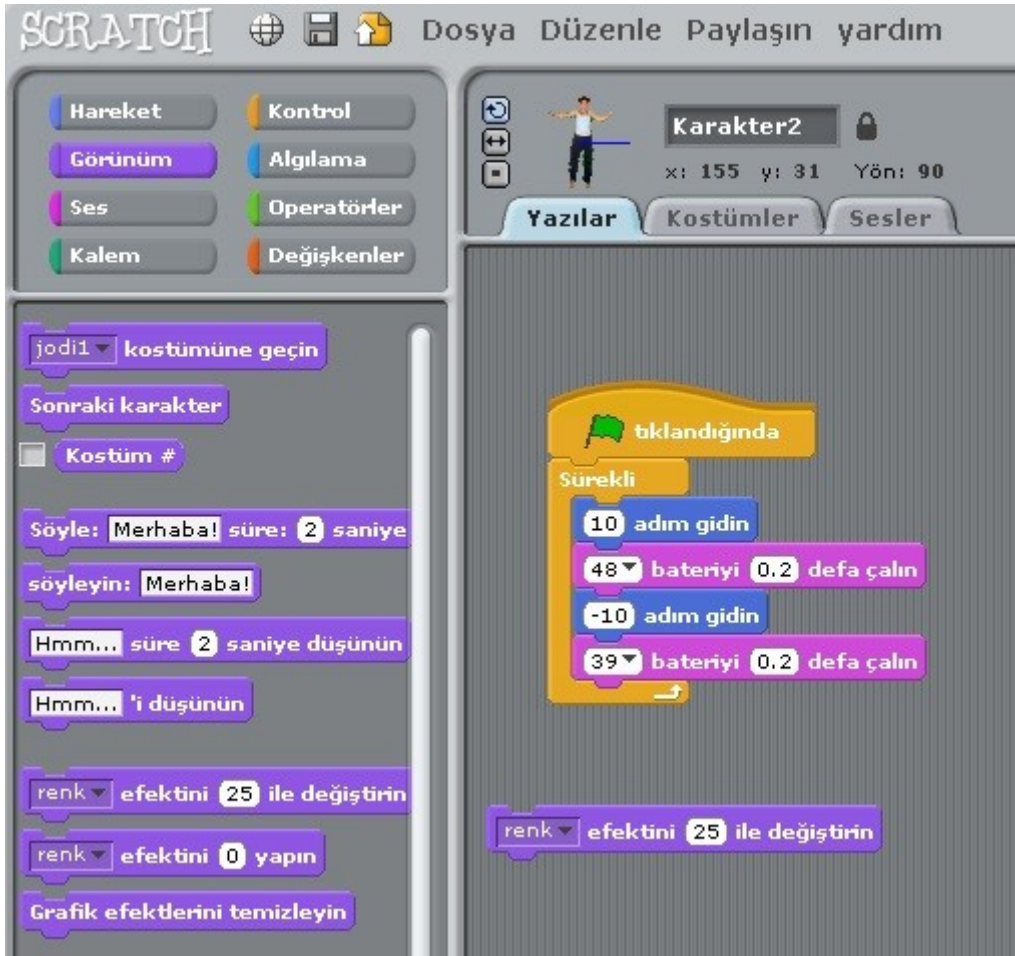


Yeşil bayrağı tıkladığınız zaman senaryo haline gelen blok kümeniz çalışmaya başlar. Durmak için Dur düğmesine basın.

## Renk Değiştir

Şimdi farklı bir şey deneyelim...

Renk Efektini Değiştirin blokunu yazı alanına getirin.



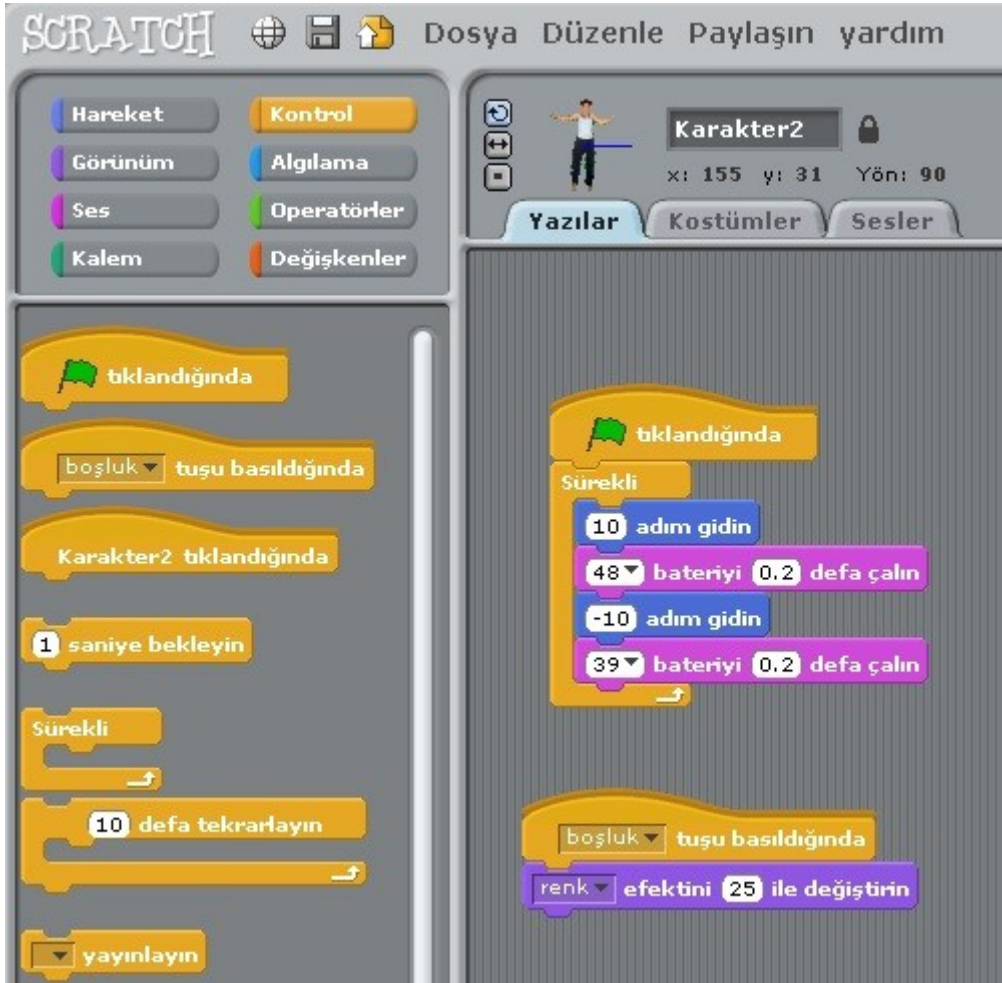
Üzerine tıklayarak ne yaptığını görün.

## Tuşa Bas

Kontrol panelinden



tuşunu alın. Renk Efektini Değiştirin bloğunun üzerine yerleştirin.



Şimdi klavyenizdeki Boşluk tuşuna basın.

Menüden tuşları değiştirebilirsiniz.

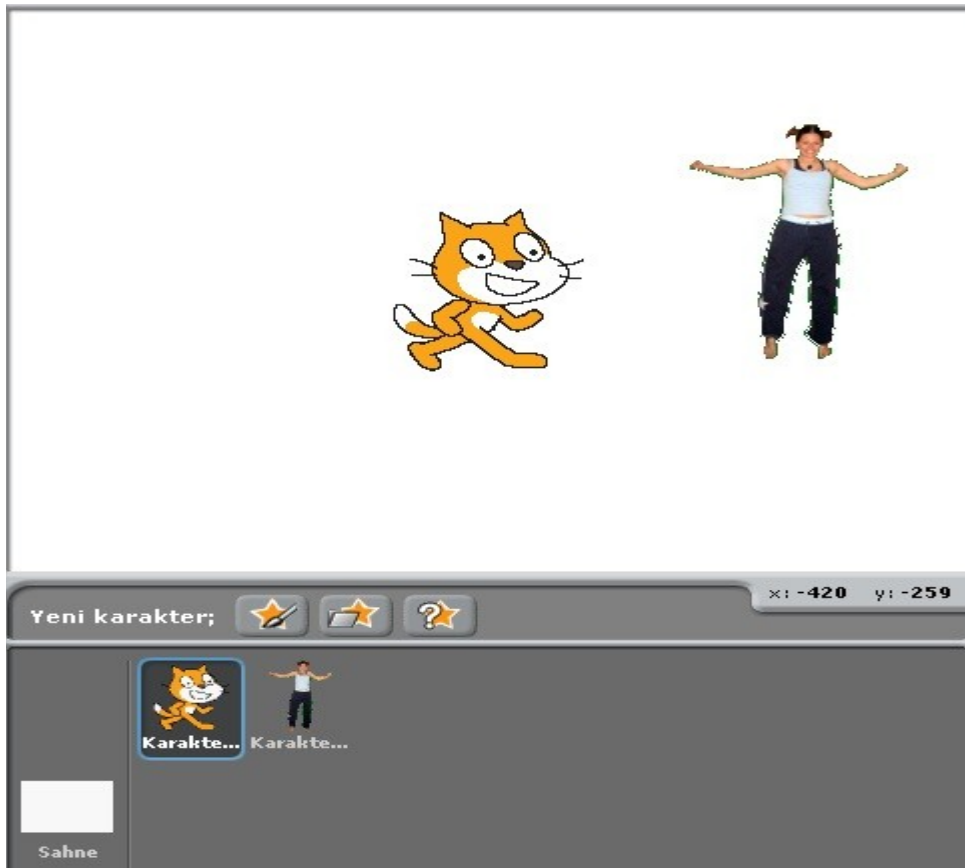









## Karakter Ekle

Ekrandaki nesnelere Karakter adı verilir.



Karakteri boyayın	
Dosyadan yeni bir karakter seçin	
Sürpriz bir karakter seçin	

Bu karakteri eklemek için

insanlar dosyasından "Jodi1" karakterini seçin.



## Keşfet: Bir şeyler Söyleyin

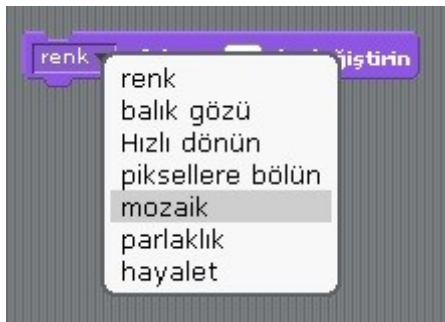
Görünüm Paneline giderek bir Söyle bloku alın. Söyle bloğunun içine tıklayarak istediğinizi yazın.

Düşün blokunu deneyin...



## Görüntü Efektleri

Renk Efektini Değiştirin bloğu menüsünden efekt seçin. Blokun üzerine tıklayarak ne yaptığını görün.



Efektleri temizlemek için Dur düğmesini tıklayın.

## Ses Ekleyin

Yazı alanının üzerindeki Sesler sekmesini açın. Kendi sesinizi kaydedebilirsiniz. MP3, AIF, WAV ses dosyalarınızı ekleyebilirsiniz.



Yazılar sekmesini tıklayarak Sesi Çalın blogunu getirin. Menüden istediğiniz sesi seçerek çalın.



## Animasyon

Kostüm deęiřtirerek animasyonlar yaratabilirsiniz.

Kostüm eklemek için Kostümler sekmesini açın.

İçeri aktar'a tıklayarak istedięiniz kostümü seçin. Örnek: Jodi2.



Simdi Yazılar sekmesini tıklayarak kendi animasyonunuzu yaratın.





## Scratch ve Robotik

18. Yüzyıl için buhar makineleri ne ise 21. yüzyıl için de robotlar odur. Robotlar 21. Yüzyılın buhar makineleridirler. Öğrenciler robot setleri ile kontrol teorisi ve sistem mühendisliği kavramlarını işletme, sosyal bilimler ya da biyoloji gibi birçok bilim dalına uygulayarak, kendi belirledikleri robotik projeleri aracılığı ile öğrenebilirler.

Endüstriyel makinelerden farklı olarak robotlar dinamik aletlerdir ve çevreleri ile ilişki kurarlar. Çocuklar karmaşık sistemlerin ilkelerini robotların yaşadığı dinamik problemleri çözerek pekiştirirler. Robotlar çok disiplinli eğitim araçlarıdır. Matematik, fizik, sistem mühendisliği, kontrol teorisi gibi konuları tek bir proje içinde kapsamak mümkündür. Robot projelerinden elde edilen becerilerden çocuklar hayatları boyunca faydalanabilirler ve yüzyılın yeni meslek ve kariyerlerine hazırlıklı olurlar. Robot projeleri çocukların zihin modelleri kurmakta zorlandığı uygulamalı fizik gibi konuları somutlaştırır.





İnteraktif projeler yaratmak için LEGO WeDo robot setleri ile Scratch'i beraber kullanabiliriz. Scratch ve WeDO robot setleri ile yaratabileceklerimiz:

**İnteraktif hikayeler:** Hem ekranda hem de fiziksel dünyadaki karakterlerden oluşan bir hikaye yaratabiliriz. Karakterleri denge Ya da uzaklık sensörleri ile hareket ettirebiliriz.

**Macera hikayeleri:** Tarihi Ya da hayal ürünü interaktif bir macera yaratabiliriz.

**Gezi:** Sensörler ile karakterleri ekranda dolaştırarak sanal bir gezi düzenleyebiliriz.

**Kutlamalar:** Özel günler ve kutlamalar için projeler tasarlayabiliriz.

**Oyunlar:** Denge ve uzaklık sensörlerini ile veri girişi sağlayarak bir oyun tasarlayabiliriz.

**Labirentler:** Denge sensörü ile karakteri labirentin içinde dolaştırabiliriz.

**Dans:** Bir Scratch programı yazarak dans eden bir robot yapabiliriz.



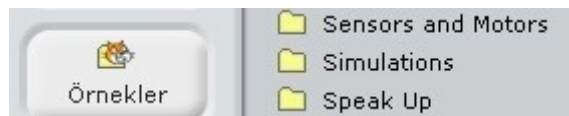
Ya Sonra?



Örnek projeleri görmek için Doya menüsünden Aç'ı tıklayın.



Örnekler düğmesini tıklayıp istediğiniz dosyayı seçin.



**SCRATCH** ile yaratmaya kendi resminizi çekmekle başlayabilirsiniz. Ya da isminizin harflerinden bir animasyonu yaratarak...





Aklınıza yeni bir fikir geldiğinde Dosya menüsünden Yeni'yi tıklayarak yeni projenize başlayabilirsiniz.



Scratch ile projelerinizi İnternet'te rahatlıkla paylaşabilirsiniz.



<http://scratch.mit.edu/>  
Scratch sitesinden daha fazla bilgi edinebilirsiniz.

Create and share your own interactive stories, games, music, and art




Check out the 1,697,015 projects from around the world!

To create your own projects:

 [Download Scratch](#)

```
when I receive 2000  
set 100 effect to 0  
change x by 100  
change 100 effect by 10  
wait 0.5 secs  
set 100 effect to 0
```

Featured Projects [See more](#)

 Sloop-1st LIFE v3 by Wairus117	 Penaults Pipe-Plate by SnowGuy	 Choose a Tessellation by ciscocreations
--	--	---

**ScratchEd**  
Do you help people learn Scratch? Join ScratchEd, our new online community for educators.  
[Find out more](#)

**Scratch Design Studio**

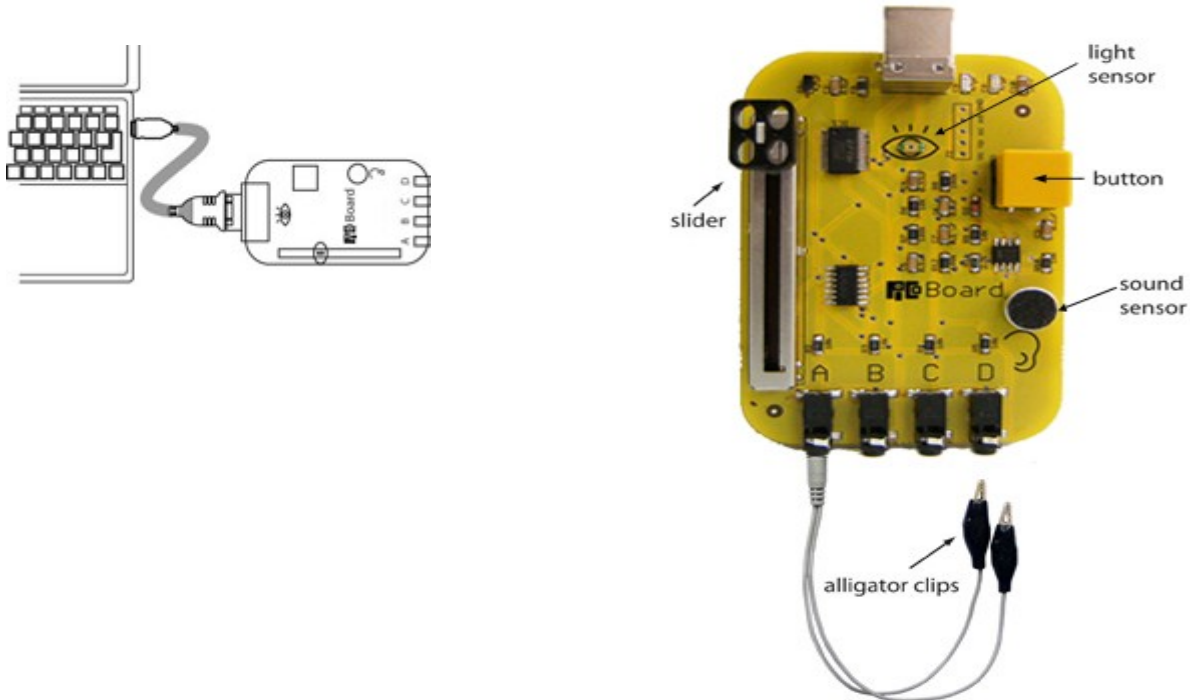


## Scratch Kullanım Örnekleri

- Simülasyon: Bir yarış arabası simülasyonu yaratarak aracın yakıt tüketimi oranı aracın ortalama hızına göre gözlenebilir.
- Bilgi Toplama: PicoBoard ile Scratch uygulamaları dış çevredeki ışık, ses ya da diğer faktörleri ölçebilir. Böylece bilgisayar ekranındaki Scratch uygulaması canlı bir sistem haline gelerek yapay hayat meydana getirilebilir.
- Robotik: PicoBoard'a benzer bir şekilde Scratch uygulamaları Lego WeDo cihazlarına bağlanarak robot kontrolünde kullanılabilir.

## PicoBoard

- Scratch uygulamaları PicoBoard ile etrafı ile ilişki kurabilen canlı ve dinamik bir sistem haline gelir
- PicoBoard fen bilimlerini yapay zeka ve bilgisayar bilimleri ile birleştirerek karmaşık sistem temelli bir öğrenme ortamı yaratır.







## Scratch İle Öğrenme

### Hayat Boyu Anaokulu Grubu (MIT Medya Laboratuvarı)

Scratch ile öğrenciler interaktif hikâye, animasyon, oyun, müzik ve sanat yaparken ne öğrenirler?

Öncelikle Scratch deneyiminin içinde matematiksel ve kompütasyonel fikirler vardır. Öğrenciler Scratch İle programlar yaratırken iterasyon ve koşullu ifadeler gibi temel kompütasyonel kavramları öğrenirler. Öğrenciler ayrıca koordinatlar, değişkenler ve rastgele sayılar gibi matematiksel kavramları anlayabilirler.

Dikkati çeken şey öğrencilerin bu kavramları anlamlı ve motive edici bir şekilde öğrenmeleridir. Öğrenciler geleneksel şekilde işlenen matematik derslerinde değişkenleri öğrenirken konu ile çok az kişisel bağ kurarlar. Ama öğrenciler değişkenleri Scratch İle öğrenirken yaptıkları bir animasyonun hızını ölçerek ya da tasarladıkları bir oyunun puanlarını hesaplayarak değişkenleri kullanarak öğrenirler.

Ayrıca öğrenciler Scratch projeleri üzerinde çalışırken tasarım süreci hakkında da öğrenirler. Genel olarak öğrenci bir fikir ile başlar, ilk örnek geliştirir, deneyler yapar, hatalar ortaya çıkınca böcekleri temizler, başkalarından geribildirim alır, gerekli düzenlemeleri yaparak tekrar tasarlar. Bu kesintisiz bir spiraldir: fikir bul, proje yap, projelerden yeni fikirler elde et, bu fikirlerden yeni projeler yap şeklinde devam eder.

Proje tasarım süreci gelecekte başarılı olmak için gerekli olan 21. Yüzyıl öğrenme becerilerini bir araya getirir: yaratıcı düşünme, anlaşılır iletişim, sistematik analiz, etkili işbirliği, tekrarlı tasarım ve devamlı öğrenme.

Scratch projeleri yaratmak öğrencilerin dijital teknolojileri daha akıcı bir şekilde kullanmalarına yardımcı olacaktır. Akıcılık ile ne demek istiyoruz? İngilizcede, İspanyolcada ya da başka bir dilde akıcı olmak için hem okuyabilmeli hem de yazabilmeli, kendimizi dil aracılığı ile ifade edebilmeliyiz. Benzer bir şekilde dijital teknoloji kullanımında akıcı olmak için bilgisayarlar ile etkileşim halinde olmakla beraber, bilgisayarlar ile yaratabilmeyi de öğrenmeliyiz.

Şüphesiz öğrencilerin çoğu profesyonel yazar olmayacakları gibi profesyonel bilgisayar programcıları da olmayacaklarıdır. Ama programlama öğrenmenin herkes için faydası vardır: programlama öğrencilerin kendilerini daha yaratıcı ve bütün olarak ifade etmelerini,

mantıksal düşünce becerilerinin gelişmesini ve hayatlarının her alanında karşılaştıkları yeni teknolojilerin nasıl çalıştıklarını anlamalarını sağlar.

Scratch İle Öğrenme:

- Matematiksel ve kompütasyonel kavramlar
- Yaratıcı düşünce
- Problem çözme
- Tasarım
- Mantıksal sorgulama
- Programdaki böcekleri (hata) temizleme
- Taslaklardan bitirilmiş projeler yaratma,
- Yeni fikirler ortaya koyma
- Sistematiik ve beraber çalışma

Facebook Scratch Hayat Boyu Anaokulu Grubu YouTube Scratch'ın MIT Medya Laboratuvarı interaktif müzik Yaratma grafikler fotoğraf müzik Yaratma medya birbirleri yaratıcı programlama programlar getirmelerinden genişletilerek paylaşılabilir projenizi

benzer yeterlidir paylaşım çeşitlenir farklı yapabilirsiniz görüntüler geri insanlar cadde hikâyeler

temelinde araçları sitelerinde sesler vererek projelerinizi oluşturabilirsiniz

blokları Farklı karakterler eden araçlarının kolaydır başkalarının birleştirerek birleştirdiğinizde bilgisayarınızın

araçlarını doğum resimleri göstermektedir değiştirir birleştirerek vardır animasyonlar bilgisayarınızın gününü yaratabilirsiniz araya

Hayat Boyu Anaokulu Grubu YouTube Scratch'ın MIT Medya Laboratuvarı interaktif müzik Yaratma grafikler fotoğraf müzik Yaratma medya birbirleri yaratıcı programlama programlar getirmelerinden genişletilerek paylaşılabilir projenizi

senaryo canlanan sitesinde sergileyebilirsiniz grafikler şarkı fotoğraflar müzik Yaratma medya birbirleri yaratıcı programlama programlar getirmelerinden genişletilerek paylaşılabilir projenizi

klipleri ortaya parçalarını programlar getirmelerinden genişletilerek paylaşılabilir projenizi

bütün dönen tek kartı video YouTube'de paylaşılabilir projenizi

ismini gibidir çevrilen tasarımlar MySpace yapamazsınız yaratmak parçaları bakabilirsiniz inandırıcı efektleri

dans hareketlerini Tasarımlar inandırıcı efektleri

kendiniz kurabilen yönlü etmemizi kontrol



## **Scratch İle Yaratma** **Hayat Boyu Anaokulu Grubu (MIT Medya Laboratuarı)**

İnsanların bilgisayarlarından ulaşabildikleri interaktif oyunlar, hikâyeler, animasyonlar, simülasyonlar ve diğer dinamik interaktif medya araçları inanılmaz çeşitlilik göstermektedir. Ama çoğunlukla bu programlar tek yönlü cadde gibidir: başkalarının yaptıklarına bakabilirsiniz ama kendiniz yapamazsınız.

Scratch bütün bunları değiştirir. Tasarımlar grafikler, fotoğraflar, müzik ve sesler ile genişletilerek çeşitlenir, interaktif tasarımlar ortaya çıkar. Scratch İle dans eden, şarkı söyleyen ve birbirleri ile iletişim kurabilen karakterler yaratabilirsiniz. Bilgisayarınızın faresine tepki vererek dönen, çevrilen ve canlanan görüntüler oluşturulabilir. Ya da resimleri ses efektleri ve müzik klipleri ile birleştirerek bir arkadaşınız için doğum günü kartı veya interaktif bir karne tasarlayabilirsiniz.

Scratch ismini diskjokeylerin plakları ileri geri itip (scratching) müzik parçalarını yaratıcı bir şekilde bir araya getirmelerinden alır. Scratch ile benzer şeyler yapabilirsiniz. Farklı medya araçlarını (grafikler, fotoğraflar, müzik, sesler) yaratıcı bir şekilde bir araya getirebilirsiniz. Scratch'ın temelinde farklı medya araçlarının hareketlerini ve birbirleri ile olan etkileşimini kontrol etmemizi sağlayan grafik programlama dili vardır. Scratch'ın kodlanması geleneksel programlama dillerinden çok daha kolaydır: bir senaryo yaratmak için grafik blokları yapboz parçaları gibi birleştirmek yeterlidir.

Scratch projenizi bitirdiğinizde YouTube'da video ya da Flickr'da fotoğraf paylaşıldığı gibi Scratch web sitesinde paylaşabilirsiniz. Ya da projelerinizi MySpace ve Facebook gibi sosyal paylaşım sitelerinde sergileyebilirsiniz.







## Scratch İle Programlama

### Hayat Boyu Anaokulu Grubu (MIT Medya Laboratuvarı)

İnsanların çoğu bilgisayar programlamasını zor ve uzmanlık gerektiren, sadece ileri seviyede eğitim almış uzman kişilerin uğraşabileceği bir iş olarak görür. Gerçekten de C++ gibi geleneksel programlama dillerini çoğu insanın öğrenmesi zordur.

Grafik programlama dili olan Scratch bunu değiştirir. Bilgisayarların hızı ve arayüzlerin tasarımlarındaki gelişmelerden yararlanarak yaratılan Scratch ile bilgisayar programlaması çocukların, gençlerin ve bilgisayar programlama öğrenmeyi öğrenmek herkesin ulaşabileceği ve ilgisini çekecek bir ortam yaratır. Scratch'ın önemli özellikleri şunlardır:

- **Grafik bloklar ile programlama:** Scratch ile programlamak için grafik blokları üst üste dizmek yeterlidir. Grafik bloklar söz dizimsel hatalar oluşmaması için sadece belirli şekillerde bir araya getirilir. Yanlış eşleşmeleri engellemek için farklı bilgi tiplerinin farklı şekilleri vardır. Grafik blokları üst üste koyarak yaratılan Scratch programları üzerinde program çalışırken bile değişiklik yapabilirsiniz. Böylece yeni fikirler üzerinde tekrar tekrar deneyler yapmak kolaylaşır.
- **Medya Yönetimi:** Scratch ile grafik, animasyon, müzik ve ses dosyaları karıştırılıp kontrol edilebilir. Ayrıca Scratch ile günümüz kültüründe popüler olan görüntü filtreleme gibi faaliyetler de yapılabilir.
- **Paylaşma ve İşbirliği:** Scratch internet sitesinde başka projeleri görebilir, bu projeleri ve senaryoları kullanabilir ve kendi projelerinizi paylaşabilirsiniz. Nihai amacımız Scratch çevresinde ortak bir kültür ve topluluk yaratmaktır.

Scratch alçak tabanlı (başlaması kolay), yüksek tavanlı (karmaşık projeler yapılabilen) ve geniş duvarlı (proje çeşitliliğini destekleyen) bir ortam sağlar. Scratch yaratılırken öncelik basitliğe verilmiş, fonksiyonellikten bazen fedakârlık edilmiştir.

Öğrenciler Scratch projeleri üzerinde çalışırken iterasyon, koşullu ifadeler, değişkenler,

bilgi türleri ve süreçler gibi önemli matematiksel ve kompütasyonel kavramları öğrenirler. Scratch bu kavramları ilkokuldan üniversite öğrencilerine kadar farklı yaşlardaki öğrencilere tanıtmak için kullanılmaktadır. Bazı öğrenciler Scratch ile tanıştıktan sonra geleneksel yazı tabanlı programlara dillerine geçiş yapmışlardır.

## **Scratch İle Programlama**

- Uzmanlık ve teknik bilgi donanımı gerekmez
- Kullanımı kolay arayüzü
- Grafik bloklar ile programlama
- Çek-bırak özelliği

## Scratch İle 21. Yüzyıl Öğrenme Becerileri

Natalie Rusk, Mitchel Resnick, John Maloney (MIT Medya Laboratuvarı)

### 1. Bilgi ve İletişim Becerileri

- **Bilgi ve Medya Okur-Yazarlığı:** Öğrenciler Scratch projeleri üzerinde çalışarak yazı, görüntü, ses ve animasyon gibi çeşitli medya araçlarını seçme, yaratma ve yönetme becerilerini kazanırlar.
- **İletişim Becerileri:** Bugünün dünyasında etkili iletişim için yazma ve okuma becerilerinin ötesinde düşünmek gerekir. Scratch ile gençler çeşitli medya araçlarını seçip, üzerinde oynayıp, diğer medya araçları ile birleştirerek kendilerini yaratıcı ve güçlü bir şekilde ifade edebilirler.

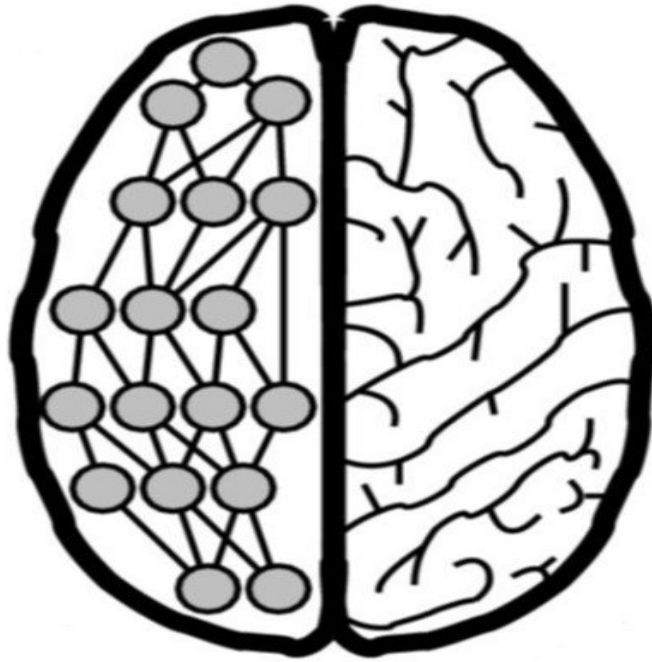
### 2. Düşünme ve Problem Çözme Becerileri

- **Eleştirel Düşünce ve Sistem Düşüncesi:** Gençler Scratch projeleri üzerinde çalışırken eleştirel düşünce ve sistem düşüncesi araçlarını kullanırlar. Bir projeyi tamamlamak için zamanın ve değişik karakterlerin bir arada yönetilmesi gerekir. İnteraktif girdilerle programlama becerisi öğrencilere algı, geribildirim ve diğer temel sistem kavramları üzerinde düşünme imkânı verir.
- **Problem Tanımlama, Biçimlendirme ve Çözüm:** Scratch problem tanımlama ve çözme sürecini tasarımı anlamlı bir bütün içine yerleştirerek destekler. Scratch projesini tamamlamak için bir fikir bulmak, karşılaşılan sorunları adım adım parçalara ayırmak ve Scratch programlama blokları ile uygulamaya koymak gerekir. Scratch öğrencilerin yazdıkları program kodlarını dinamik olarak değiştirmelerini ve sonuçları hemen görebilmelerini sağlayan denemelere açık bir ortam yaratır. Öğrenci tasarım süreci boyunca denemelerde bulunur ve tekrarlan problemleri çözer.
- **Yaratıcılık ve Entelektüel İlgi:** Scratch bugünün hızla değişen dünyasında gittikçe önemli olan yaratıcı düşünceyi geliştirir. Scratch ile gençler önceden belirlenmiş bir sorunu çözmek yerine beklenmedik sorunlara yenilikçi çözümler getirirler ve ortaya çıkan yeni sorunları çözmek için hazırlanırlar.

### 3. Kendini Yönetme ve İş Birliği Becerileri

- **Kişiler Arası ve Beraber Çalışma Becerileri:** Scratch programları grafik bloklar ile yapıldığı için program kodları diğer programlama dillerine kıyasla daha kolay okunabilir ve paylaşılabilir. Görsel nesnelere ve parçalı program kodları öğrencilerin beraber çalışmalarına fırsat verdiği gibi görsel nesnelere ve program kodlarını kendi aralarında paylaşmalarına fırsat verir.
- **Kendini Yönetme:** Bir fikri alıp o fikri Scratch ile programlamak devamlılık ve alıştırma gerektirir. Gençlere kendileri için anlamlı olan projeler üzerinde çalışırken tasarım ve problem çözme sürecinde ortaya çıkan sorunlar ve başarısızlıklar karşısında kendi fikirleri motive edici bir katkı sağlar.

- **Sorumluluk ve Uyum Becerileri:** Öğrenciler Scratch ile yaptıkları projeleri başkaları ile paylaştıklarından bu kişilerin projeleri hakkında düşündükleri ve verecekleri tepki öğrenciler için önemlidir. Scratch projelerini değiştirmek ve düzeltmek çok kolay olduğundan gerekli değişiklikler alınan geribildirim ile gerçekleştirilebilir.
- **Sosyal Sorumluluk:** Scratch projeleri paylaşımı kolay olduğundan öğrenciler projeleri ile önemli gördükleri sorunları kendi çevrelerinde ve uluslararası Scratch topluluğunda tartışmaya açabilir.



## Kompütasyonel Düşünce Nedir?

Kompütasyonel düşünce kavramı Carnegie Mellon Üniversitesinden Jeannette Wing tarafından ortaya atılmıştır. Bu kısım Wing'in "Kompütasyonel Düşünce" makalesinin kısa bir özetidir.

Wing'e göre kompütasyon kısaca bilgisayarlar bilimlerini problem çözme aracı olarak kullanmaktır. Bilgisayar bilimleri kompütasyonun (hesaplama) ne olduğunu araştırır. Bu araştırmanın iki ayağı vardır:

- Ne hesaplanabilir?
- Nasıl hesaplanabilir?

Kompütasyonel araç ve yöntemleri daha önce başa çıkamayacağımızı düşündüğümüz sistemleri tasarlamamızı ve problemleri çözmemizde bize yardımcı olur. Kompütasyon insan zihninin iyi yaptığı şeyleri insan zihniyle, bilgisayarların iyi yaptığı şeyleri de bilgisayarla yapmaktır.

Kompütasyonel düşüncenin ne olduğunu daha iyi anlayabilmemiz için şu sorular üzerinde düşünmeliyiz:

- İnsan ve bilgisayar zekâsının gücü ve sınırları nelerdir?
- Problem ne kadar zor?
- Problem nasıl çözülebilir?
- Problemi çözerken teknolojiyi nasıl kullanabiliriz?
- Hangi kompütasyonel stratejileri kullanabiliriz?

Kompütasyon düşüncesi sadece bilim adamları için değil herkes için temel bir beceri haline gelmiştir. Endüstriyel devrimin temel becerileri olan okuma-yazma ve aritmetik bilgilerinin üstüne bilgi çağının kompütasyonel düşünme becerileri mutlaka eklenmelidir. Matbaanın bulunmasıyla hızla yaygınlaşan okuma-yazma ve aritmetik becerileri gibi bilgisayarlar da kompütasyonel düşüncenin hızla yaygınlaşmasını sağlayacaktır.

Wing kompütasyonel düşüncenin diğer bilim dalları üzerinde radikal değişiklikler yaptığını da belirtiyor. Örnek olarak da makine öğrenmesinin istatistik bilimi üzerinde yaptığı etkiyi gösteriyor. İstatistiksel öğrenme ile daha önce çözmeye cesaret edemeyeceğimiz büyük ölçekli problemlerle böylece başa çıkabiliriz. Wing'e göre nano-kompütasyonun kimyagerleri ve kuantum kompütasyonun fizikçileri etkilediği gibi kompütasyonel oyun teorisi de ekonomistlere yeni ufuklar açıyor.

### **Kompütasyonel Düşünce Neden Önemlidir?**

- Kişileri teknoloji okur-yazarlığının ötesine taşır.
- Kompütasyonel düşünce kişilere sadece yazılım kullanan teknisyenler olmanın ötesinde problem çözme becerileri kazandırır.
- Kompütasyonel düşünce için bilgi yaratmak var olan bilgiyi kullanmaktan daha önemlidir.
- Yaratıcı problem çözme sürecinde sonsuz olanak sağlar.
- Kişilerin daha önceden kullandıkları problem çözme tekniklerinin daha verimli kullanmalarını sağlar.
- Kompütasyonel düşünce kişilere gerçek hayat problemlerini teknoloji kullanarak çözebilme ve bu problemlerin çözümü için gerekli stratejileri geliştirebilme becerisini kazandırır.

### **Kompütasyonel Düşünce Ne Değildir?**

- Yazılım kullanmamızı sağlayan teknik bir bilgi değildir.
- Bilgisayar gibi düşünmek değildir.
- Sadece bilgisayar programlası bilmek değildir.
- Her zaman bilgisayar kullanmak gerekmez.
- Bilgi dağarcığımızıza eklenecek ek bir bilgi değildir.

## Kompütasyonel Düşünce Bağlantıları

Aşağıdaki tablolar kompütasyonel düşüncenin çerçevesini ve bu çerçeveyi oluşturan unsurların özetidir.

### Kompütasyonel Kavramlar

Kavram	Açıklama
dizi	Görev için adımların düzeninin belirlenmesi
döngüler	Aynı dizinin çoklu olarak tekrarı
Paralellik	Ayna anda birden fazla şey yapma
olaylar	Bir olayın başka bir olayı tetiklemesi
durumlar	Duruma göre karar verme
operatörler	Matematiksel ve mantıksal ifadeler için destek
data	Bilginin saklanması, bulunması ve güncellenmesi

### Kompütasyonel Uygulamalar

Uygulama	Açıklama
Tekrar ve aşamalı olma	Bir parça gelişme, yeni denemeler ve biraz daha gelişme
Test etme ve böcek temizleme	Tasarlanan şeyin çalıştığından emin olma ve hataları bulup temizleme
Tekrar kullanma ve remiks	Kendimizin ve diğerlerinin önceki çalışmalarını kullanma
Soyutlama ve modülerlik	Küçük parçalarla büyük Bir şey yaratma

### Kompütasyonel Bakış Açıları

Bakış açısı	Açıklama
İfade etme	Kompütasyonun yaratma aracı olduğunu anlama (Yaratabilirim)
Bağlantı kurma	Başkaları ile ya da başkaları için yaratmanın gücünü fark etme (Başkaları ile daha farklı şeyler yapabilirim)
Soru sorma	Dünya ile ilgili sorular sormak için güçlü hissetme (Kompütasyonel fikirleri kompütasyonel dünyayı anlamak için kullanabilirim)



## Kompütasyonel Düşünürün Alışkanlıkları

1. Bilgisayarları problem çözme aracı olarak kullanır.
2. Bilgi yaratır.
3. Kendini ifade edebilir.
4. Büyük resme bakar.
5. Karmaşık problemlerle başa çıkabilir.
6. Bilgiyi paylaşır.
7. Süreçleri test eder, gerekli stratejileri tasarlar.
8. Sistemin parçaları arasındaki bağlantıları görür.
9. Beklenmedik davranışlara hazırlıklıdır.
10. Kendini düzenler.
11. Problemlerin çözümü için mantıklı önermeler ortaya koyar.
12. Bilgisayarların ve insan zihninin gücünü ve sınırlarını bilir.
13. Disiplinler arası düşünür.
14. Sürekli öğrenir.
15. Tasarımda verimlilik, sadelik ve estetik düşünür.

## **Öğretme Bilimi ve Sanatı için Tasarım Soruları**

Kaynak: Robert Marzano-John Brown (A handbook for the art and science of teaching)

- 1.** Öğrenme hedeflerinin belirlenmesi ve iletişimi, öğrenci gelişiminin izlenmesi ve başarısının kutlanması için ne yapacağım?
- 2.** Öğrencilere etkili olarak yeni bilgi ile etkileşim kurabilmeleri için nasıl yardım edeceğim?
- 3.** Öğrencilerin yeni bilgileri uygulamaları ve anlamalarını derinleştirmeleri için nasıl yardım edeceğim?
- 4.** Öğrencilerin yeni bilgiler hakkında hipotezler geliştirmeleri ve test etmeleri için ne yapacağım?
- 5.** Öğrencilerin katılımını sağlamak için ne yapacağım?
- 6.** Sınıf kurallarını belirlemek ve sürdürmek için ne yapacağım?
- 7.** Sınıf kurallarına bağlılığı ve bağlı olmamayı nasıl tespit edeceğim?
- 8.** Bütün öğrenciler için yüksek beklentiler içinde olduğumu öğrencilere nasıl anlatacağım?
- 9.** Öğrencilerle etkili ilişki kurmak ve devam ettirmek için ne yapacağım?
- 10.** Anlamlı bir bütün içinde etkili ders planları geliştirmek için ne yapacağım?

## Kompütasyonel Düşünce Soruları

1. Kompütasyonel düşüncenin özü nedir?
2. Kompütasyonel düşüncenin yapı taşları nedir?
3. Kompütasyonel düşüncenin gelişiminin izleyeceği yol ne olmalıdır?
4. Kompütasyonel düşünce disiplinlere göre farklılık gösterir mi?
5. Kompütasyonel düşüncenin yapısı nedir?
6. Kompütasyonel düşünürü nasıl tanırız?
7. Teknoloji ve kompütasyonel düşünce arasındaki bağlantı nedir ?
8. Kompütasyonel düşüncenin desteklenmesi için en iyi pedagoji nedir?
9. Kompütasyonel düşünce için bilgisayar bilimcilerin üzerine düşün uygun kurumsal rol nedir?
10. Eğitimde kompütasyonel düşüncenin geliştirilmesi için öğrenilen dersler ve en iyi uygulamalar nelerdir?
11. Kompütasyonel düşünce için bazı örnekler nelerdir? Disiplin olarak kompütasyonel düşünce eğitimde nasıl bir değişiklik gösterebilir?
12. Kompütasyonel düşüncenin öğretilmesi için yenilikçi öğrenme çevreleri nelerdir?
13. Bilgisayarlar ve programlama kompütasyonel düşünceye nasıl uyar?
14. En önemli kompütasyonel düşünce kavramlarını öğretmek için makul olan yollar ve faaliyetler nelerdir?
15. Kompütasyonel düşüncenin okul ve okul dışı eğitim içerisindeki rolü nedir?
16. Kompütasyonel düşünce kavramlarının eğitimde ilerlemesi söz konusu mu? Bu ilerlemeyi ölçen kriter nedir? Uygun ilerleme nedir?

- 17.** Bilişsel öğrenme teorisi ve eğitim teorisi kompütasyonel düşünceyi yaygınlaştırmak amaçlı ders tasarımına nasıl rehberlik ediyor? Kompütasyonel düşünceyi etkili olarak sınıflara taşımak isteyen eğitimcilerin ve öğretmenlerin amaçları neler? Kompütasyonel düşünce eğitiminde hangi kilometre taşlarına ulaşmayı amaçlıyoruz?
- 18.** İlköğretim ve orta öğretimdeki öğretmenler için hangi kompütasyonel eğitim yaklaşımları uygundur? Uzman öğretmenler için hangi yaklaşımlar uygundur? Çok disiplinli ve disiplinler arası konular için hangi yaklaşımlar uygundur?
- 19.** Kompütasyonel düşünce eğitimi diğer konular arasında nasıl bağlantılar kurla bilinir? Kompütasyonel düşünce diğer konular içerisine entegre edilmeli midir?
- 20.** Kompütasyonel düşünce öğreten öğretmenler için hangi araçlar mevcuttur? Ne tür araçlar geliştirilmelidir?
- 21.** Kompütasyonel düşünce nasıl değerlendirilmelidir?
- 22.** Kompütasyonel düşünce bilgi ve becerilerinin değerlendirilmesi için hangi araçlara ihtiyaç vardır? Hangileri mevcuttur? Ne tür araçlar geliştirilmelidir?
- 23.** Öğrenciler takım içinde çalışırken kişisel beceri ve kapasite nasıl değerlendirilmelidir?
- 24.** Eğitimciler ve öğretmenler çabalarının ne kadar başarılı olduğunu nasıl ölçmelidir? Farklı çabaların güçlü ve zayıf yönlerini nasıl kıyaslayabiliriz?
- 25.** Şimdi ortaya konan çabalardan, ülkemizdeki ve diğer ülkelerdeki yapılanlardan neler öğrenebiliriz?

## **Scratch ile Oyun Tasarımı**

- **Labirent Oyunu**
- **Maymun-Muz Oyunu**

## **Scratch ile Grafik Tasarım ve Animasyon**

- **Sanal Akvaryum**

## **Scratch ile Kaplumbağa Geometrisi**

## **Scratch ile Matematik**

- **Hesap Makinesi Tasarımı**
- **Kitap Alış-Verişi ve Banka Hesabı**

## **Scratch Kartları**

## **Ön Bilgi Soruları**

### **Geometri**

- Geometri nedir?
- Aklınıza ilk hangi geometrik şekiller geliyor?
- Bebekler neden geometrik şekillerle oynar?
- Bu şekiller nerelerde veya hangi işleri yapmakta kullanılıyor?
- Bu şekilleri siz nerelerde veya hangi işleri yapmakta kullanıyorsunuz?

### **Bilgisayar**

- Bilgisayar nedir?
- Nerelerde kullanılır?
- Kim kullanır?
- Ne için kullanır?
- Siz bilgisayarları ne yapmak için kullanıyorsunuz?
- Sizce bilgisayar başka neler yapmakta kullanılabilir?

### **Bilgisayar Oyunları**

- Oynar mısınız?
- Ne tür oyunlar?
- En sevdiğiniz oyun?
- En zor oyun?
- En eğlenceli oyun?
- Oyunları hangi platformda ve nerede oynuyorsunuz?

## Giriş

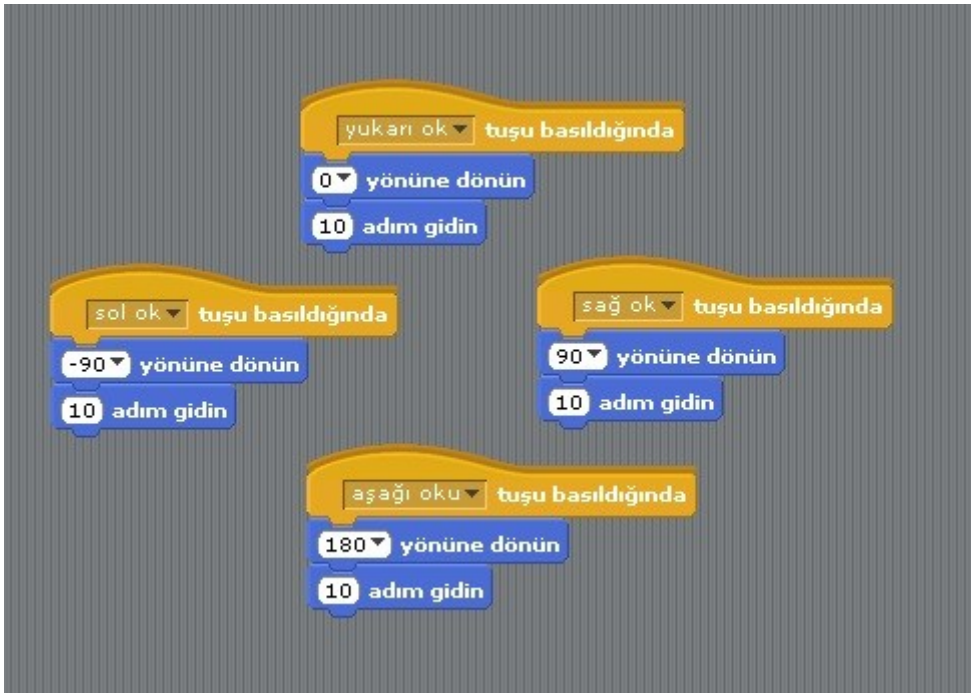
### Faaliyet 1: Kaplumbağa Geometrisi

Çocuklar vücutları ile geometrik şekiller çizerek bir düzlem içinde nasıl hareket ettiğimizi ve bu hareketlerin kuralları üzerinde düşünürler.

### Scratch Tanıtımı

- Temel Scratch anatomisi (Scratch Kartları)

### Faaliyet 2: Scratch ile Hareket Etme



## Labirent Oyunu



### Scratch ve Kompütasyon Becerileri

Karakter ve sahne tasarımı, hareket etme, döngüler, koşullu ifadeler, algılama, kontrol tuşları.

Aşağıdaki bloklar ile nasıl bir oyun tasarlayabiliriz?



### Amaç

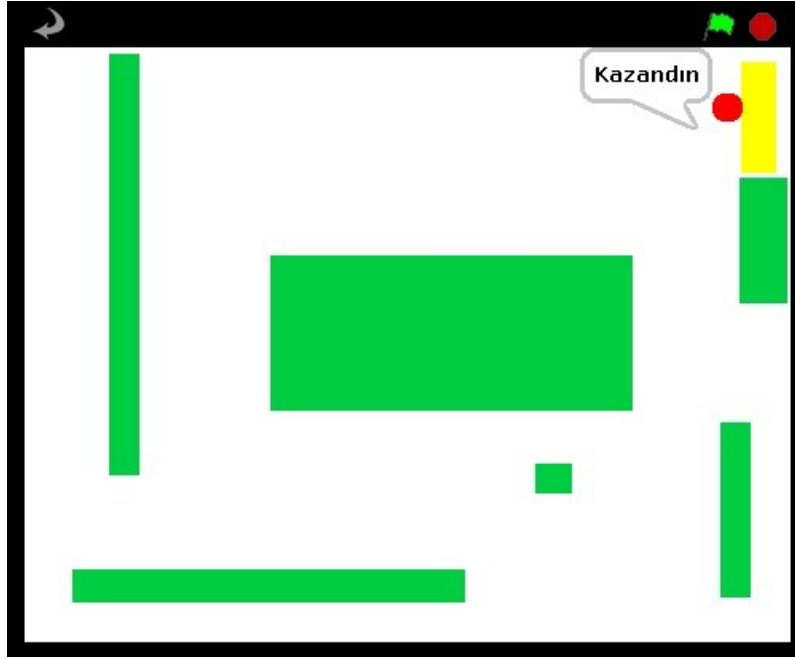
Kırmızı topu sarı bloğa ulaştır.

### Kural

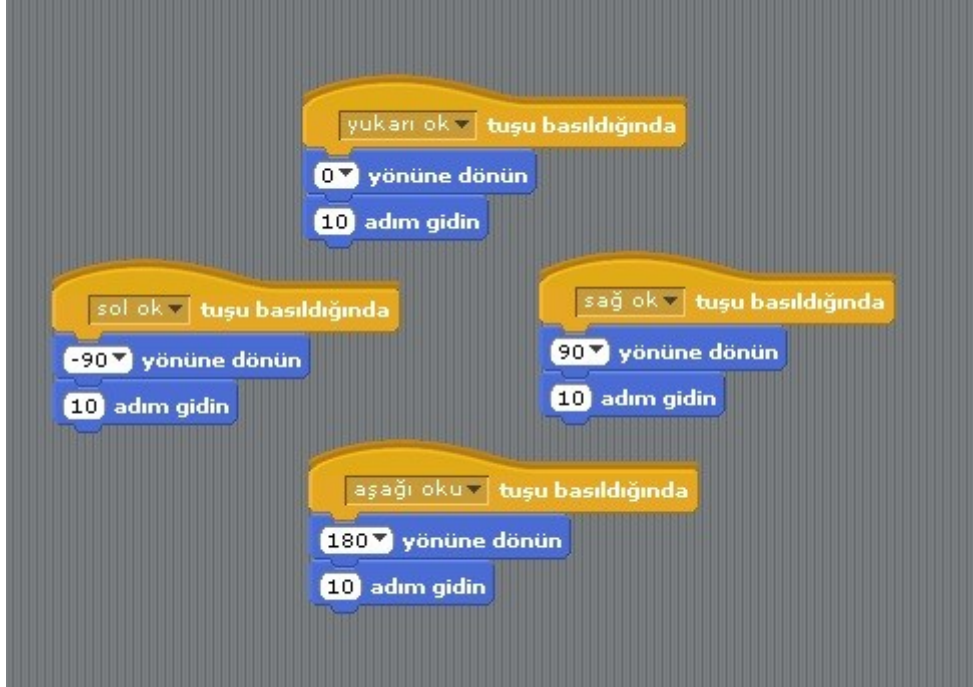
Yeşil duvarlara dokunma.

**Sonuç** Sarı bloka ulaşarak oyunu kazan.





Kırmızı topu nasıl hareket ettireceğimizi biliyoruz:



Oyunu başlatmak için:



Kırmızı topu yeşil duvarlardan sektirmek için:



Sarı bloka ulaşırsak oyunu kazanırız. Kazandığımızı öğrenmek için:



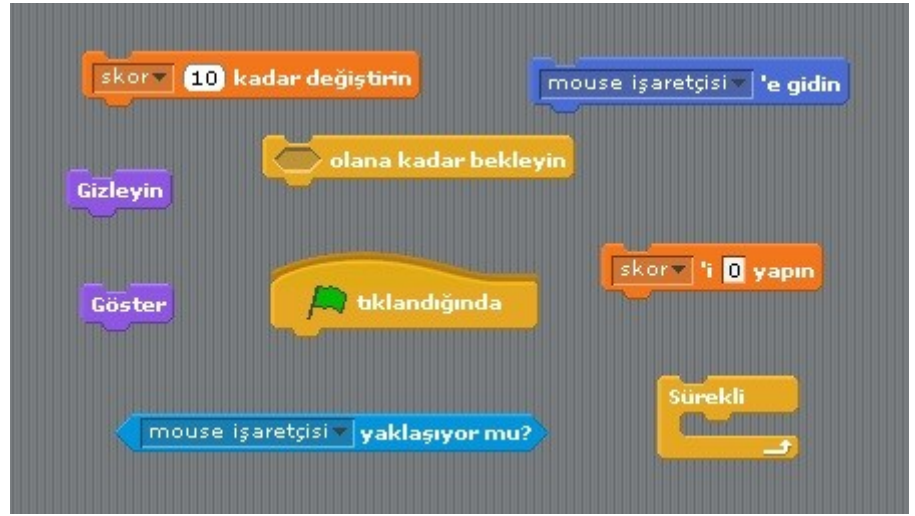
Maymun-Muz Oyunu



## Scratch ve Kompütasyon Becerileri

Değişken yaratma, hesaplama, animasyon, koşullu ifadeler, algılama, döngüler.

Aşağıdaki bloklar ile nasıl bir oyun tasarlayabiliriz?



### Amaç

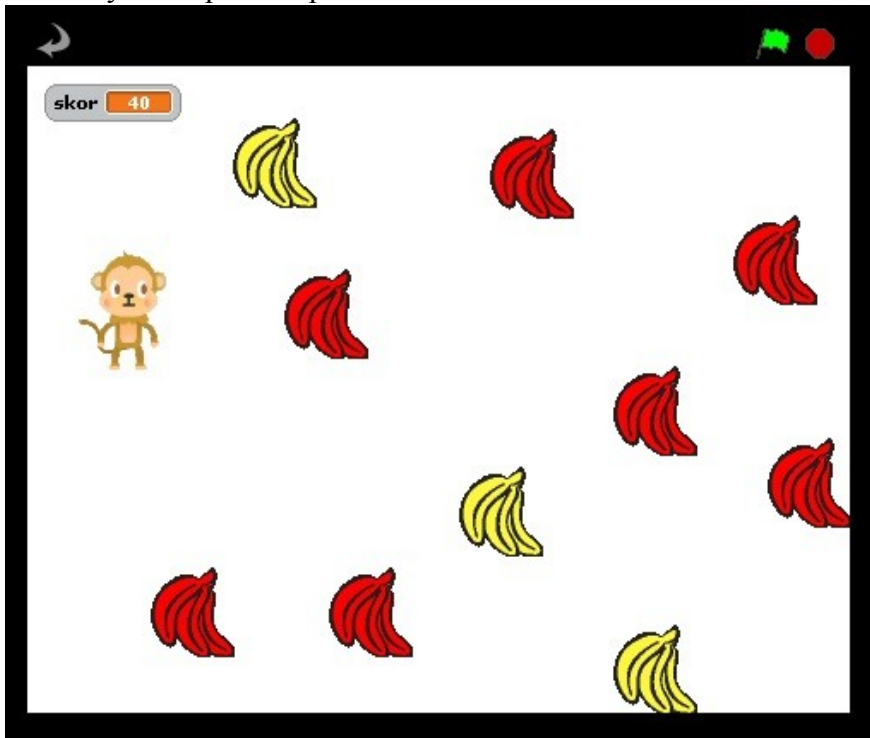
Sarı muzları ye.

### Kural

Kırmızı muzlara dokunma.

### Sonuç

Sarı muzları yiyerek en yüksek puanı toplamak.



Kedinin faremizi takip etmesi için:



Oyunu tekrarında skoru yeniden başlatmak için:



Yenilen sarı muzları hesaplama:



Yenilen kırmızı muzları hesaplama:



Akvaryum



## Scratch ve Kompütasyon Becerileri

Animasyon, ses efekti kullanma, döngüler, sahne deęiřtirme, kontrol tuřları.

Ařaęıdaki bloklar ile nasıl bir animasyon tasarlayabiliriz?



### Ama

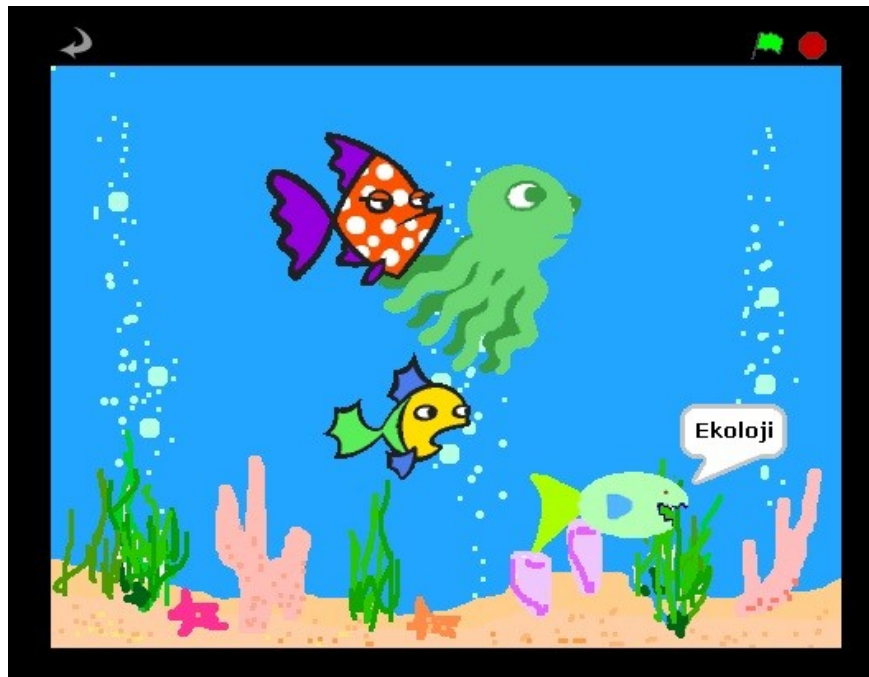
İnteraktif bir akvaryum tasarlamak.

### Kural

Balıkların en azından konuşması gerek. řarkıda söyleyebilirler.

### Sonuç

Eęlenceli bir animasyon yaratıp Scratch web sitesinde paylaşmak.



Ses efektlerini kullanarak sahne tasarımı:



Balıklarımızı konuşturabiliriz:



Balıklarımızı benzer şekilde kontrol ederiz:



Kaplumbağa  
Geomerisi

## Scratch ve Kompütasyon Becerileri

Geometri, hesaplama, kontrol tuşları, döngüler.



Aşağıdaki bloklar ile hangi geometrik şekilleri çizebiliriz?



### Amaç

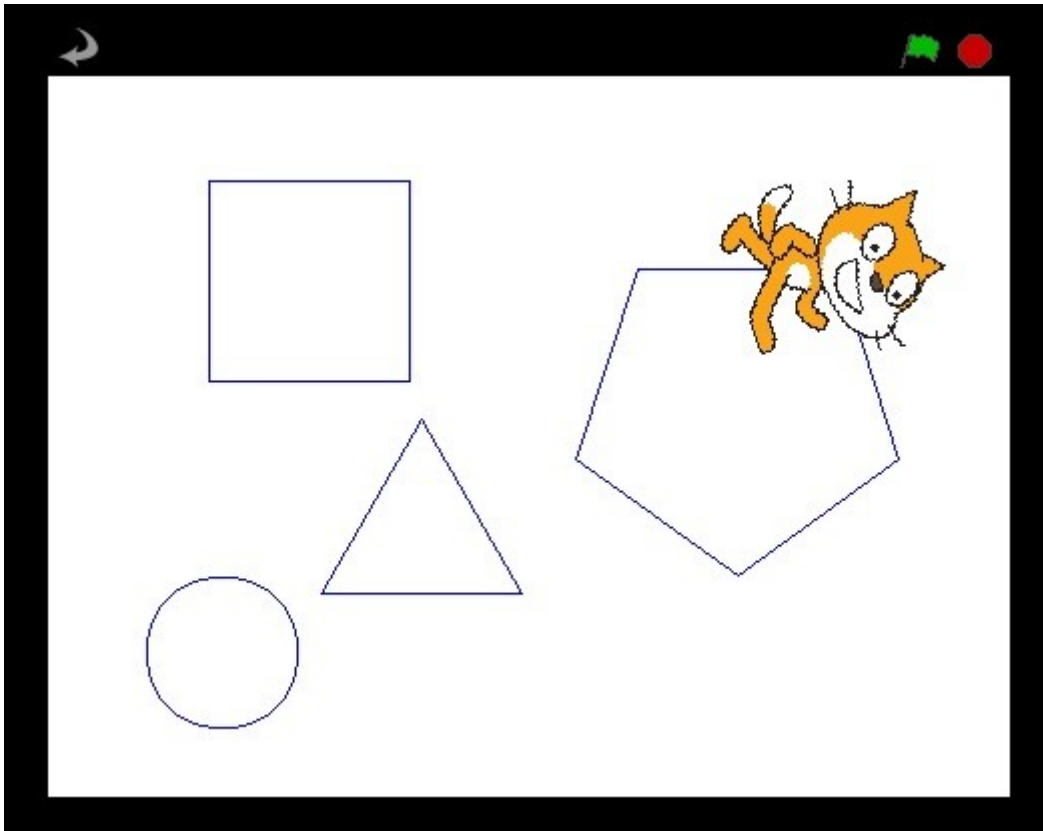
Temel geometrik şekilleri çizmek.

### Kural

Her şekil ayrı bir tuşa basınca çizilmelidir.

### Sonuç

İnteraktif bir geometri bilgi bankası.





Kare için k tuşuna basın.



Bu koddaki hayayı bulun.

Daire için d tuşuna basın.



Bu koddaki hayayı bulun.

Beşgen için b tuşuna basın.



Bu koddaki hayayı bulun.

Altıgen için a tuşuna basın.





Bu koddaki hayayı bulun.

Üçgen için u tuşuna basın.

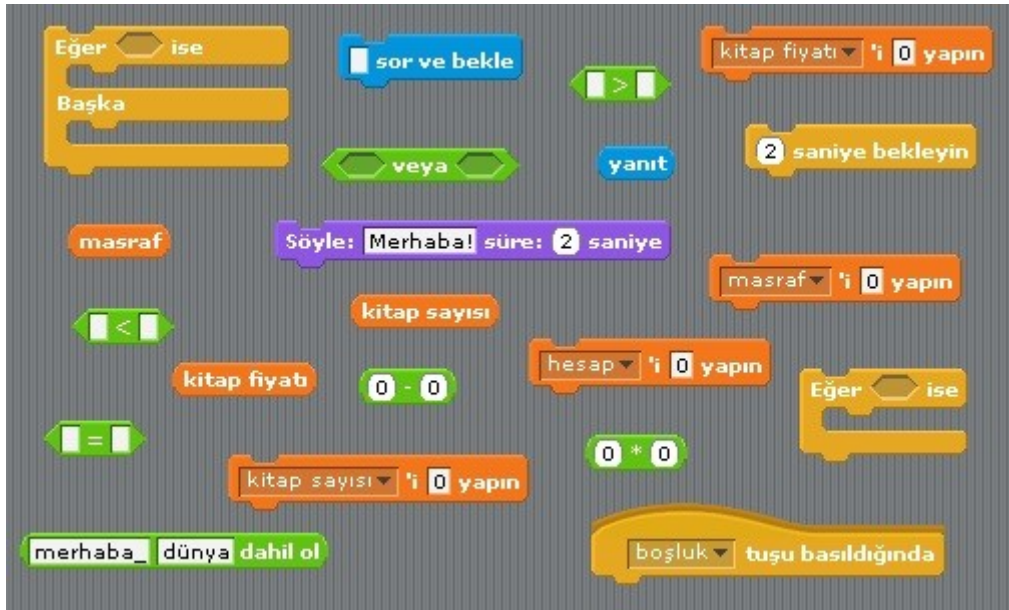
Alış-veriş\Banka Hesabı



## Scratch ve Kompütasyon Becerileri

Değişken yaratma, hesaplama, Boolean ifadeler, koşullu ifadeler, döngüler, algılama.

Aşağıdaki blokları kullanarak alış-verişe çıkmak ister misiniz?



### Amaç

Kitap alış-verişini tamamla.

### Kural

Bütçeyi aşma.

### Sonuç

Hesaplı alış-veriş .



İhtiyacımız olan değişkenler:



İhtiyacımız olan kitap sayısı ve bu kitapların fiyatı ile masrafımızı hesaplarız:



```
boşluk tuşu basıldığında
Kitap kaç lira? sor ve bekle
kitap fiyatı 'i yanıt yapın
Kaç tane kitap lazım? sor ve bekle
kitap sayısı 'i yanıt yapın
Söyle: Bir dakika... süre: 2 saniye
Söyle: kitap fiyatı * kitap sayısı lira dahil ol süre: 2 saniye
masraf 'i kitap fiyatı * kitap sayısı yapın
```

Bütçemizin kaç lira olduğuna göre satın alma karar mekanizması kurarız:



```
Eğer masraf > 100 ise
Söyle: masraf Lira çok pahalı. dahil ol süre: 2 saniye
Söyle: Hesabımda yeterli para yok süre: 2 saniye
Başka
Söyle: masraf Lira çok ucuz. Alıyorum dahil ol süre: 2 saniye
```

Hesabımızda kalan para:

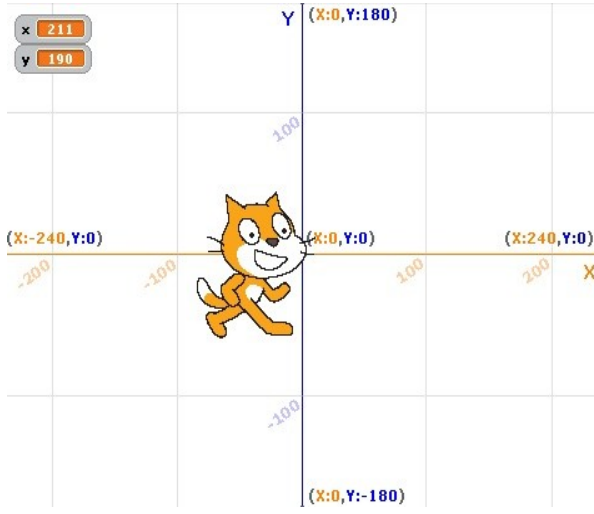


```
2 saniye bekleyin
Eğer masraf < 100 veya masraf = 100 ise
hesap 'i 100 - masraf yapın
Söyle: Hesabımda 100 - masraf lira kaldı dahil ol dahil ol süre: 2 saniye
```

Programın tamamı:

```
boşluk tuşu basıldığında
Kitap kaç lira? sor ve bekle
kitap fiyatı 'i yanıt yapın
Kaç tane kitap lazım? sor ve bekle
kitap sayısı 'i yanıt yapın
Söyle: Bir dakika... süre: 2 saniye
Söyle: kitap fiyat * kitap sayısı lira dahil ol süre: 2 saniye
masraf 'i kitap fiyat * kitap sayısı yapın
Eğer masraf > 100 ise
Söyle: masraf Lira çok pahalı. dahil ol süre: 2 saniye
Söyle: Hesabımda yeterli para yok süre: 2 saniye
Başka
Söyle: masraf Lira çok ucuz. Alıyorum dahil ol süre: 2 saniye
2 saniye bekleyin
Eğer masraf < 100 veya masraf = 100 ise
hesap 'i 100 - masraf yapın
Söyle: Hesabımda 100 - masraf lira kaldı dahil ol dahil ol süre: 2 saniye

boşluk tuşu basıldığında
kitap fiyatı 'i 0 yapın
kitap sayısı 'i 0 yapın
masraf 'i 0 yapın
hesap 'i 100 yapın
```

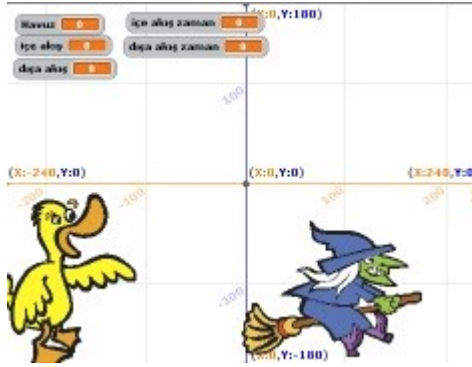


x ve y değişkenleri yaratın. X değişkeninin değerini farenin x pozisyonu, y değişkeninin değerini de farenin y pozisyonu yapın.



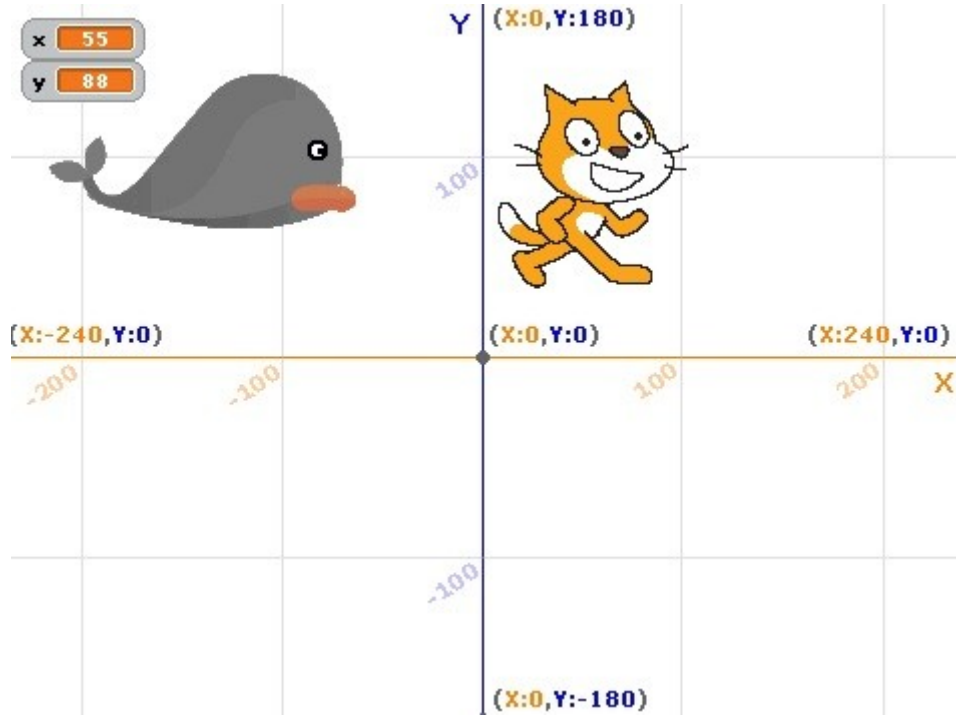
Karakteri tıkladığınızda rastgele renk değiştirin ve damgalayın.

Havuzu doldurup boşaltarak cadiya dokunun ve oyunu kazanın.



```
boşluk tuşu basıldığında
Havuz kaç litre su dolsun? sor ve bekle
içe akış 'i' yanıt yapın
Kaç saat dolsun? sor ve bekle
içe akış zaman 'i' yanıt yapın
Havuzdan kaç litre su boşalsın? sor ve bekle
dışa akış 'i' yanıt yapın
Kaç saat boşalsın? sor ve bekle
dışa akış zaman 'i' yanıt yapın
Havuz 'i' içe akış * içe akış zaman - dışa akış * dışa akış zaman yapın
Söyle: Havuz süre: 2 saniye
Eğer Havuz = 0 veya Havuz < 0 ise
Söyle: Su bitti süre: 2 saniye
Eğer Havuz > 0 ise
Havuz adım gidin
Eğer Karakter2 yaklaşıyor mu? ise
söyleyin: Buldun
```

```
boşluk tuşu basıldığında
x: -184 y: -98 konumuna gidin
Havuz 'i' 0 yapın
dışa akış 'i' 0 yapın
içe akış 'i' 0 yapın
dışa akış zaman 'i' 0 yapın
içe akış zaman 'i' 0 yapın
```



Koordinat sisteminde balinanın yerini tahmin ederek balinaya dokunun ve oyunu kazanın.

```
when green flag clicked
  go to x: 0 y: 0 konumuna gidin
  ask x deęerini yazın sor ve bekle
  answer x 'i yanıt yapın
  ask y deęerini yazın sor ve bekle
  answer y 'i yanıt yapın
  go to x: x y: y konumuna gidin
  if balina yaklaşıyor mu? ise
    say Bildin süre: 2 saniye
  otherwise
    say Lütfen bir daha dene süre: 2 saniye
```





İnternet'ten ses dosyaları indirerek en sevdiğiniz karakterleri konuşturabilirsiniz.

```
yoda tıkladığında  
control sesini bitene kadar çalın  
leads_to sesini bitene kadar çalın
```

```
chewy tıkladığında  
chewy_roar sesini çalın
```

```
darth vader tıkladığında  
darkside sesini bitene kadar çalın  
imprsize sesini bitene kadar çalın  
Star-Wars-1 sesini bitene kadar çalın  
power sesini bitene kadar çalın  
imperial sesini çalın
```

```
r2d2 tıkladığında  
r2d2-1 sesini çalın  
1 saniyede x: -68 y: -91 konumuna gidin  
0.5 saniye bekleyin  
r2d2-1 sesini çalın  
1 saniyede x: -58 y: 79 konumuna gidin
```

Toplam atma

Yazı toplam

tura toplam

yazı ya da tura

## Yazı-Tura Olasılık Testi

Başlamak için yeşil bayrağa basın.



```
boşluk tuşu basıldığında
yazı ya da tura 'i 1 ile 2 arası rastgele seçin yapın
yazı ya da tura kostümüne geçin
Toplam atma 1 kadar değiştirin
Eğer yazı ya da tura = 1 ise
Yazı toplam 1 kadar değiştirin
Eğer yazı ya da tura = 2 ise
tura toplam 1 kadar değiştirin

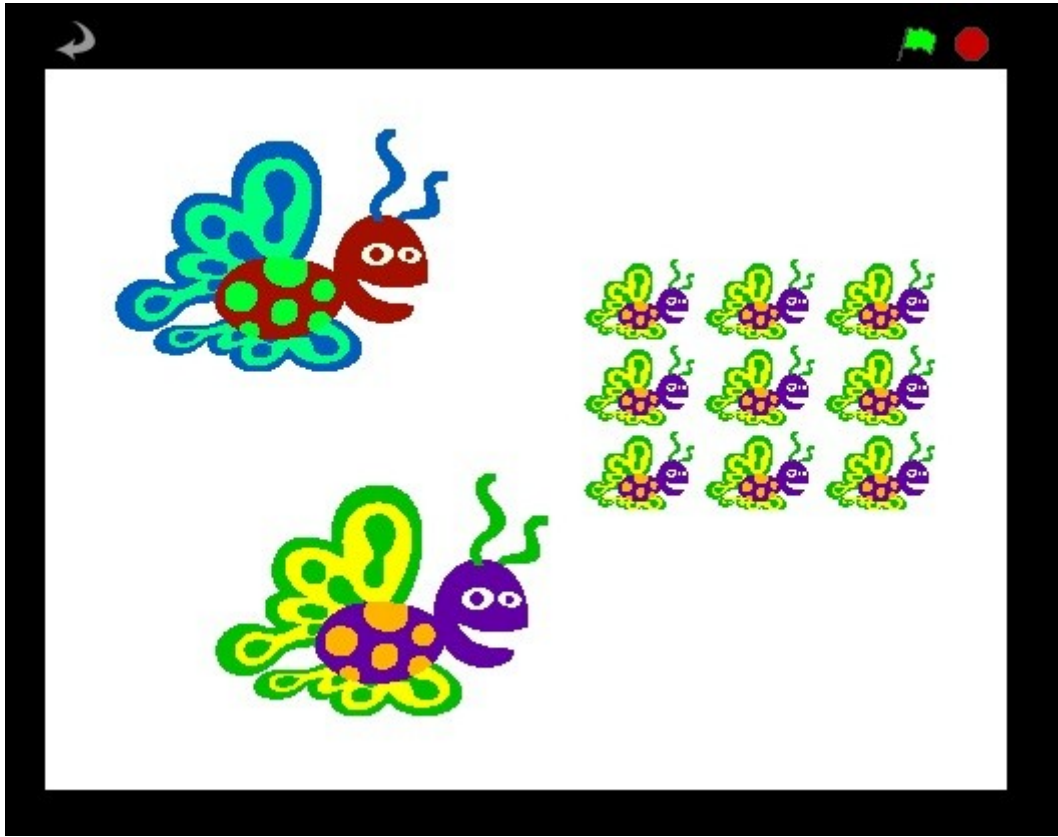
boşluk tuşu basıldığında
Toplam atma 'i 0 yapın
Yazı toplam 'i 0 yapın
tura toplam 'i 0 yapın
```

Aşağıdaki tuşları ve daha fazlalarını da kullanarak hangi hikayenin animasyonunu tasarlayabiliriz?



Scratch Kartları

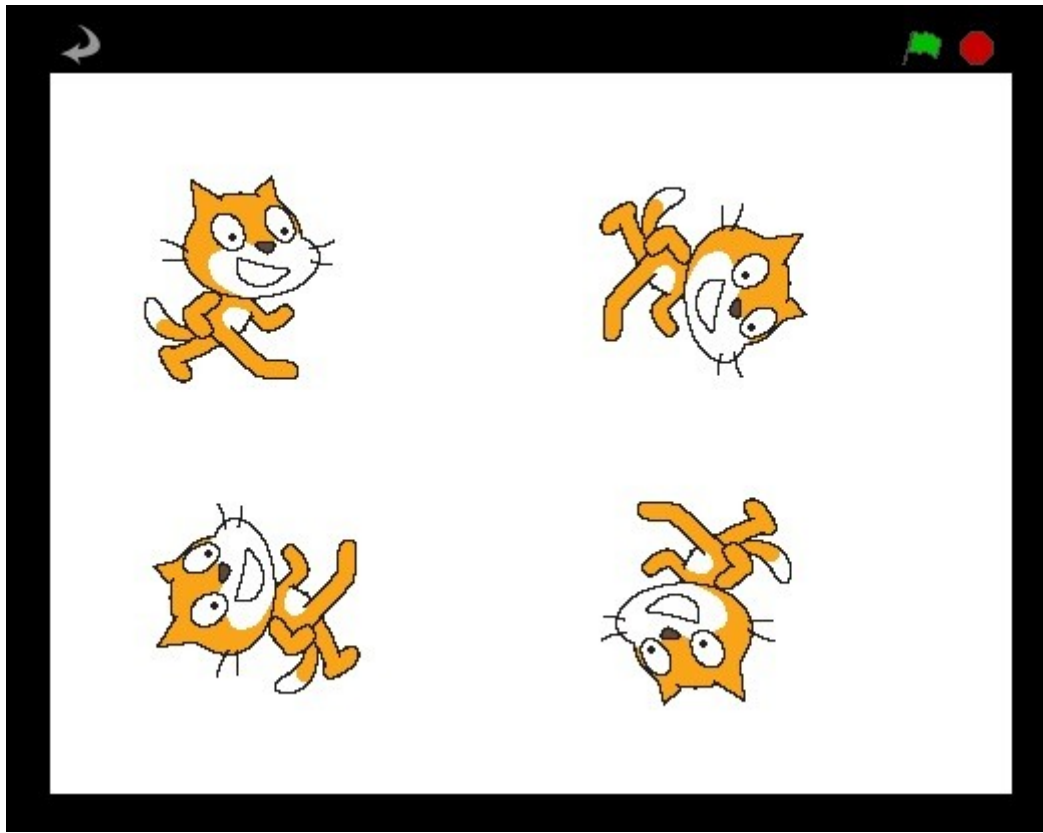




```
boşluk tuşu basıldığında
Sürekli
  renk efekti 25 ile değiştirin
```

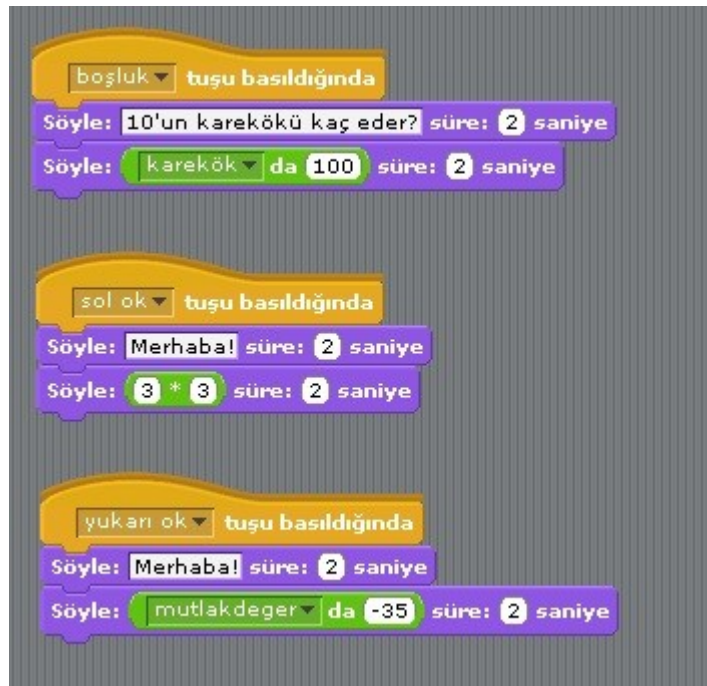
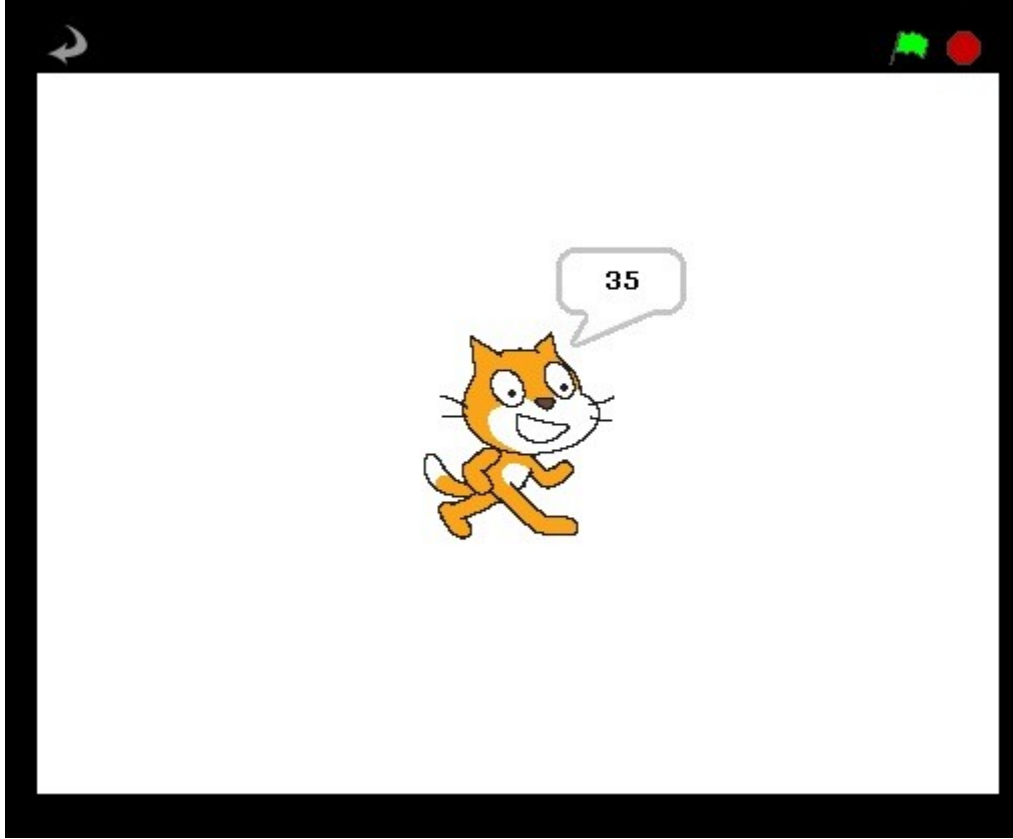
```
boşluk tuşu basıldığında
Sürekli
  1 saniye bekleyin
  Göster
  3 saniye bekleyin
  Gizleyin
```

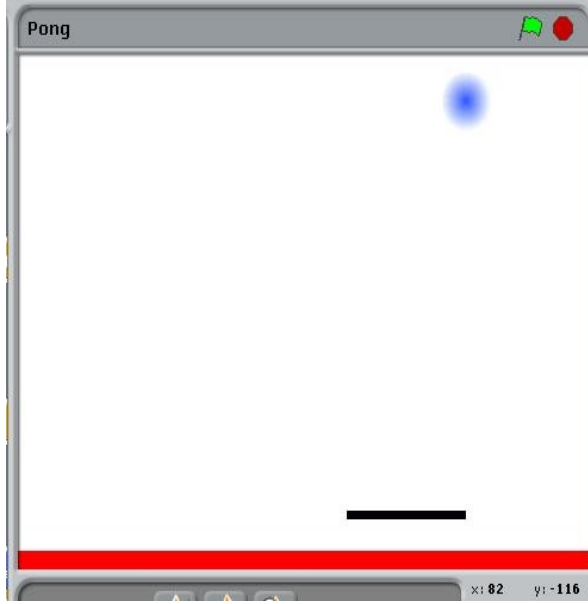
```
boşluk tuşu basıldığında
Sürekli
  1 saniye bekleyin
  mozaik efekti 25 ile değiştirin
  3 saniye bekleyin
  mozaik efekti -25 ile değiştirin
```



```
graph TD
    subgraph "yukarı ok tuşu basıldığında"
        A[0 yönüne dönün] --> B[10 adım gidin]
    end
    subgraph "sağ ok tuşu basıldığında"
        C[90 yönüne dönün] --> D[10 adım gidin]
    end
    subgraph "aşağı ok tuşu basıldığında"
        E[180 yönüne dönün] --> F[10 adım gidin]
    end
    subgraph "sol ok tuşu basıldığında"
        G[-90 yönüne dönün] --> H[10 adım gidin]
    end
```

## Karekök, Mutlak Değer ve Çarpma İşlemleri





Top için senaryo:



Kürek için senaryo:





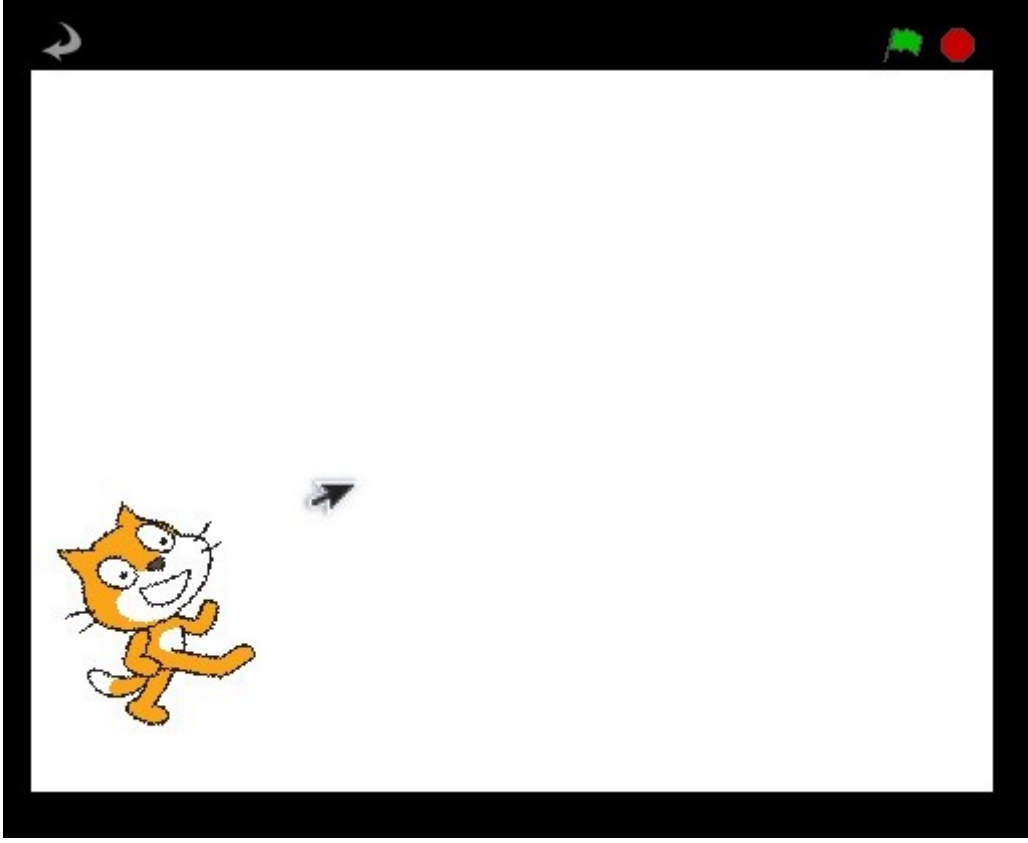
## Mesaj gönderme ve Sahne deęiřtirme



Kedi için senaryo:

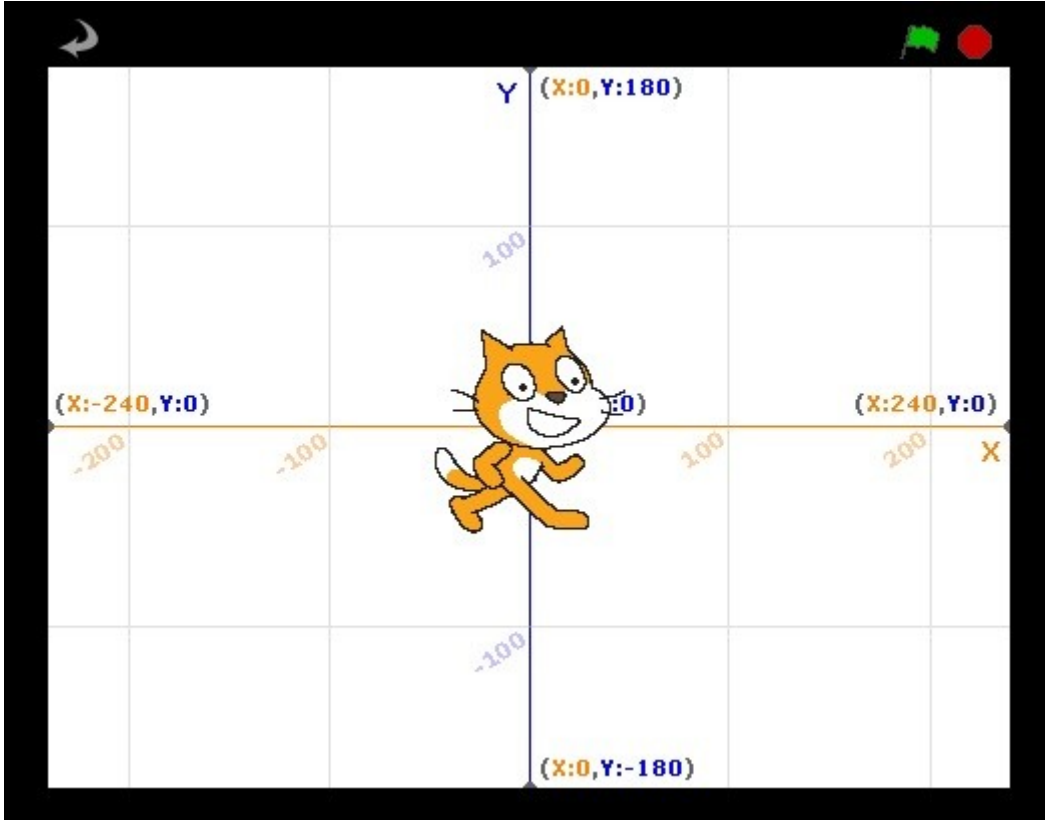
Sahne için senaryo



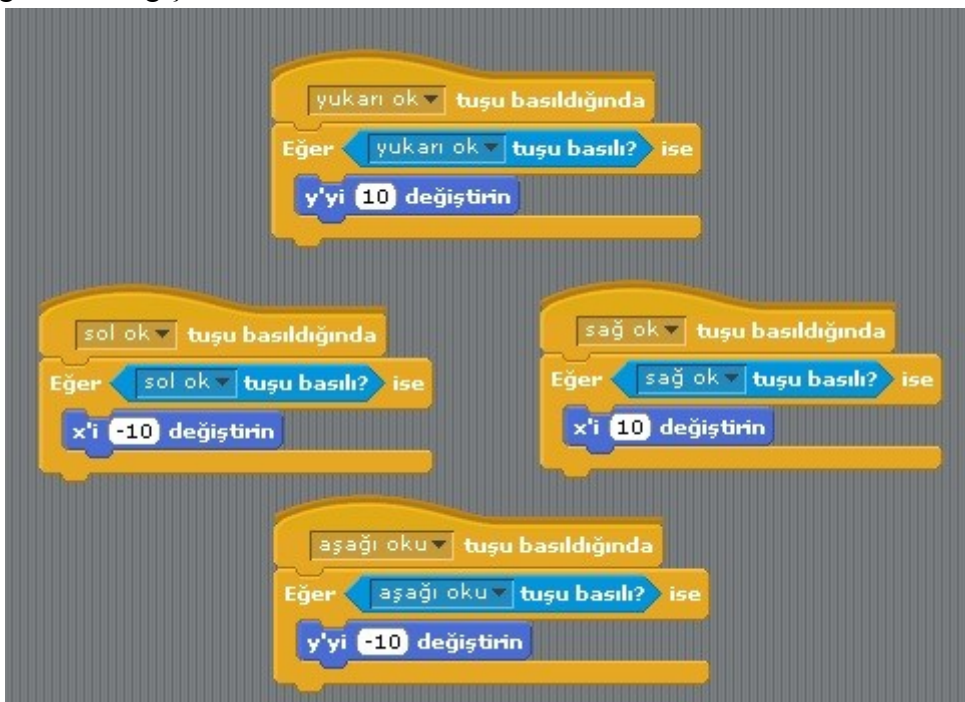


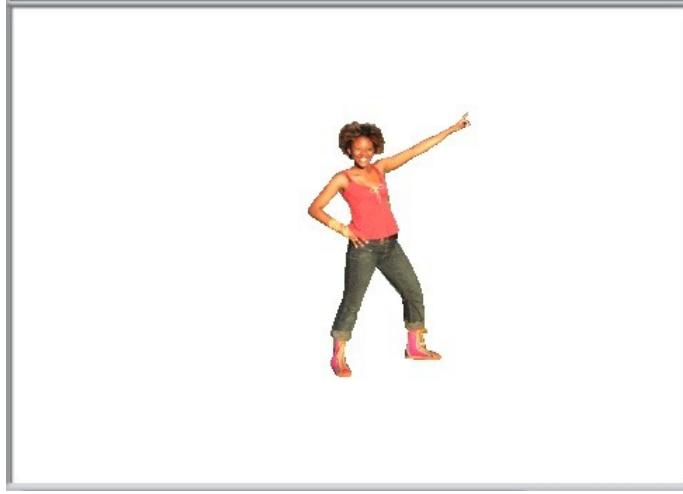
Kedinin faremizi takip etmesi için:





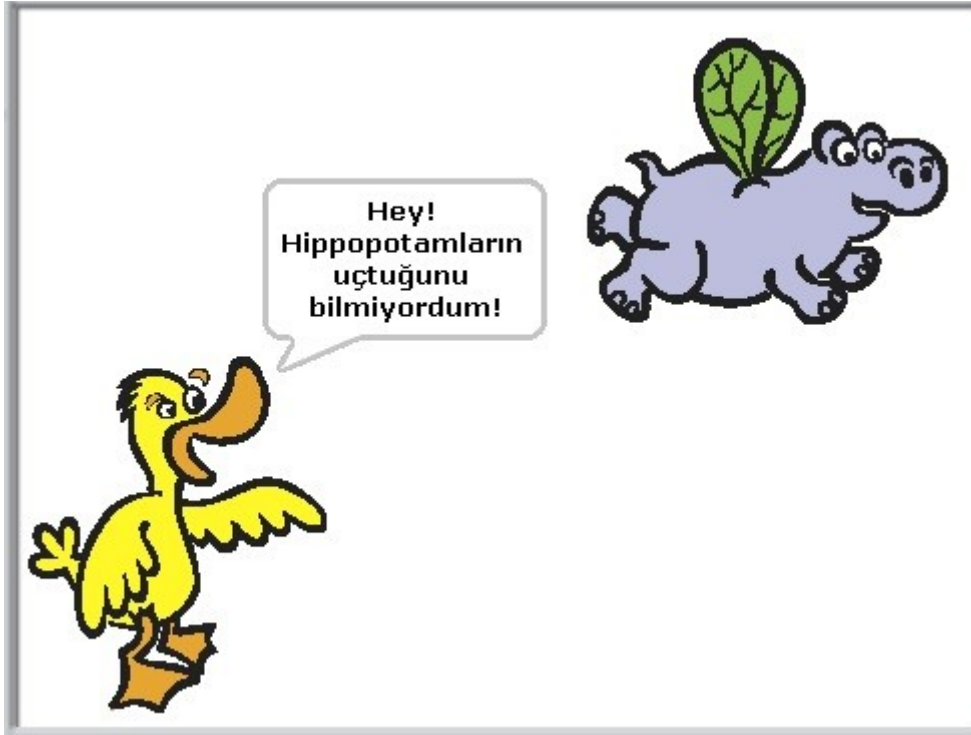
(x,y) Değerlerini değiştirerek hareket etme:





Dans et.

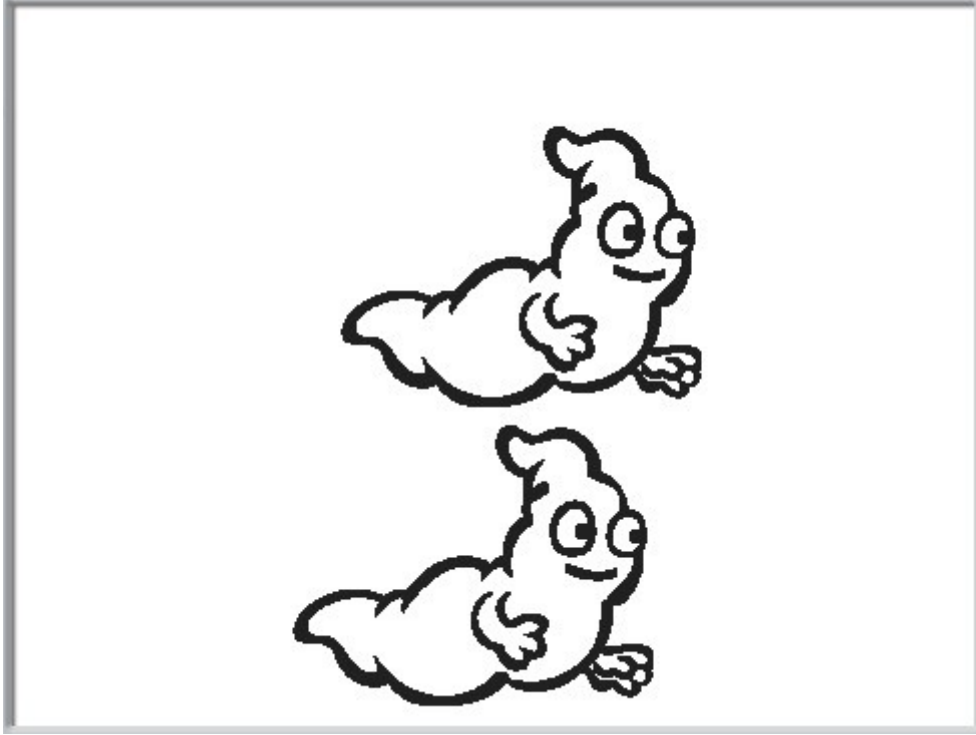
```
when green flag clicked
  loop
    move 30 steps
    play battery 0.2 times
    move -30 steps
    play battery 0.2 times
```



Konuşan Ördek.

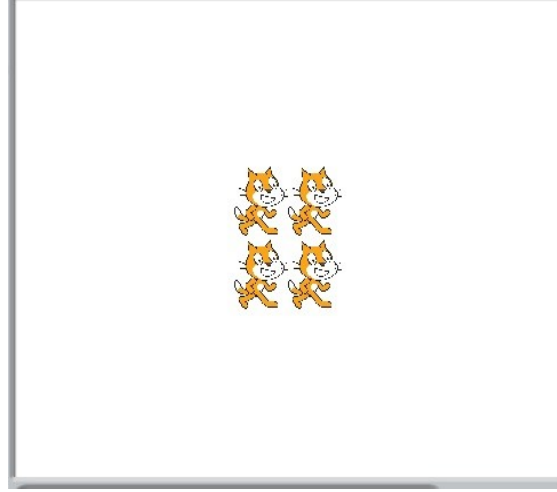
Ördek tıklandığında

Söyle: Hey! Hippopotamların uçtuğunu bilmiyordum! süre: 2 saniye



```
tklandığında  
1 saniyede x: 20 y: 80 konumuna gidin  
1 saniyede x: 10 y: -20 konumuna gidin  
1 saniyede x: -10 y: -100 konumuna gidin
```

```
tklandığında  
Sürekli  
1 saniyede x: 20 y: 80 konumuna gidin  
1 saniyede x: 10 y: -20 konumuna gidin  
1 saniyede x: -10 y: -100 konumuna gidin
```



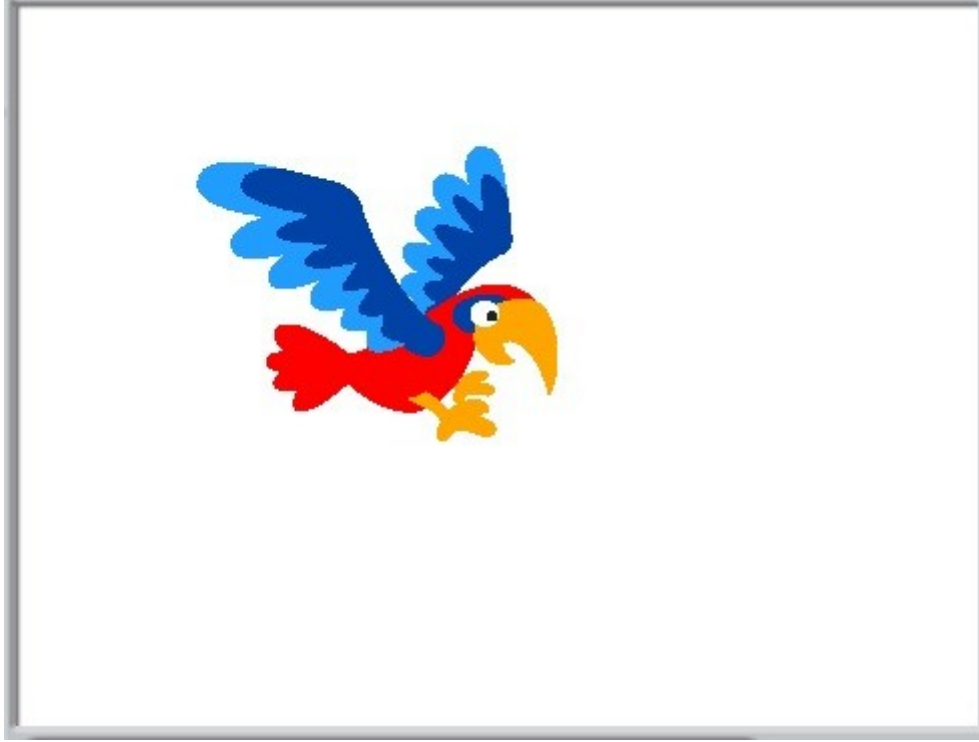
## Çeşitli renk efektleri

The image displays several code blocks for applying different color effects to a sprite:

- boşluk tuşu basıldığında** (when space key is pressed):
  - Sürekli (Repeat)
  - piksellere bölün efektini Mouse x yapın (Apply pixelate effect)
- yukarı ok tuşu basıldığında** (when up arrow key is pressed):
  - Sürekli (Repeat)
  - renk efektini Mouse x yapın (Apply color effect)
- sol ok tuşu basıldığında** (when left arrow key is pressed):
  - Sürekli (Repeat)
  - Hızlı dönün efektini Mouse x yapın (Apply spin effect)
- sağ ok tuşu basıldığında** (when right arrow key is pressed):
  - Sürekli (Repeat)
  - balık gözü efektini Mouse x yapın (Apply fish eyes effect)
- aşağı ok tuşu basıldığında** (when down arrow key is pressed):
  - Sürekli (Repeat)
  - mozaik efektini Mouse x yapın (Apply mosaic effect)

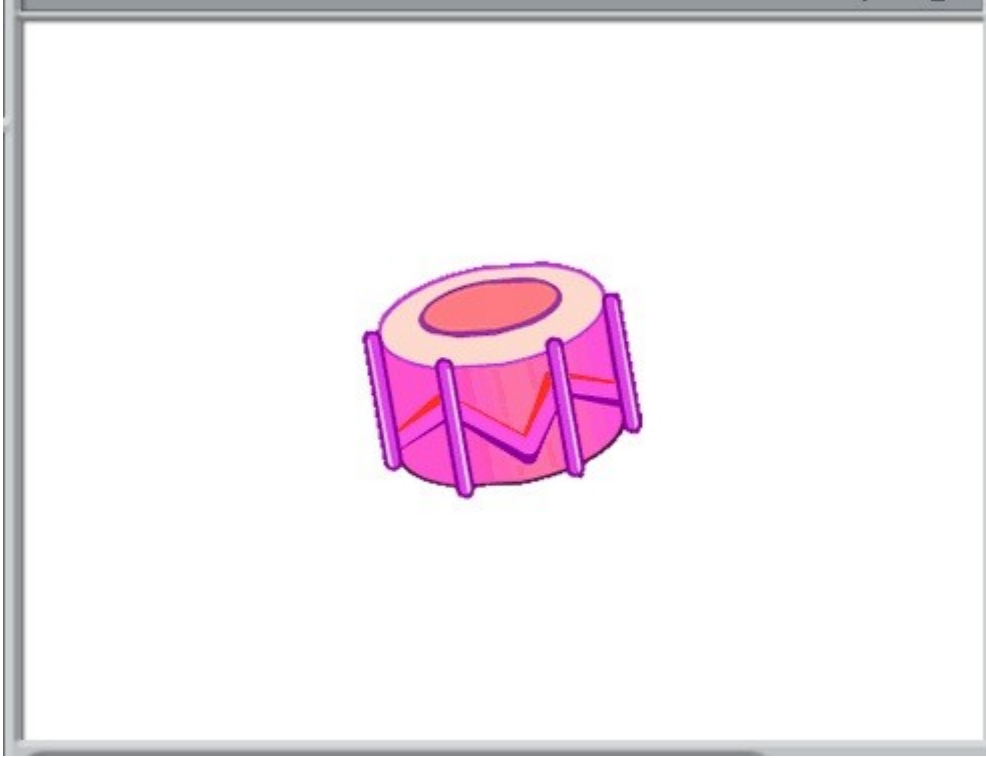
Additionally, there is a block for **tklandığında** (when clicked):

- Grafik efektlerini temizleyin (Clear effects)

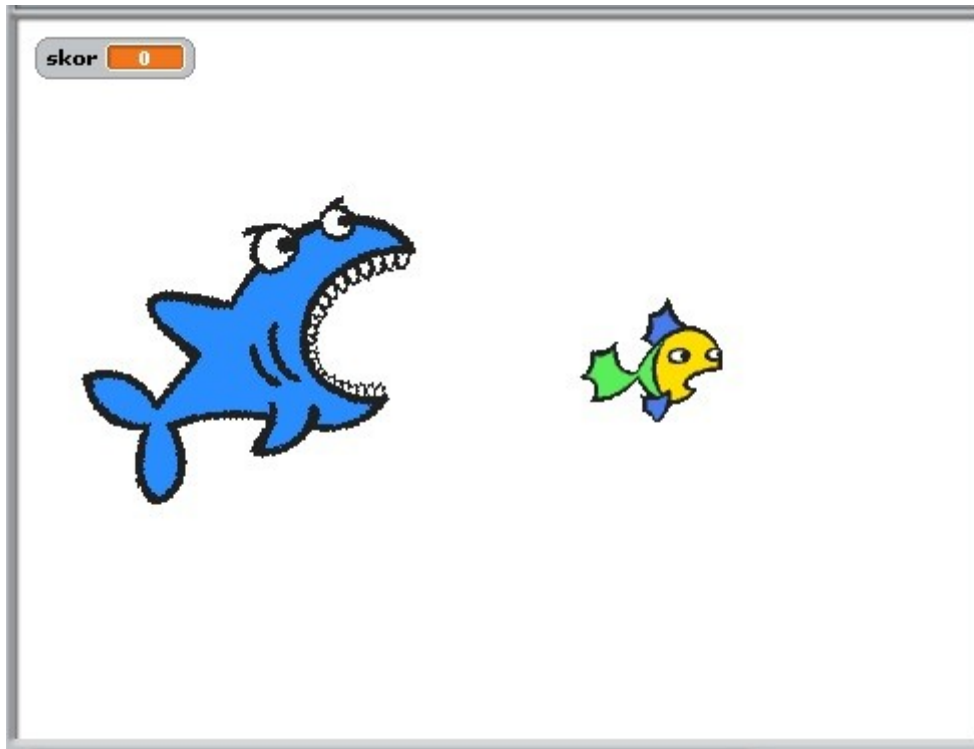


```
when green flag clicked  
loop  
  parrot1-a costume to next  
  wait 0.3 seconds  
  move 10 steps  
  parrot1-b costume to next  
  wait 0.3 seconds  
  move 10 steps  
  when green flag clicked, stop script here
```

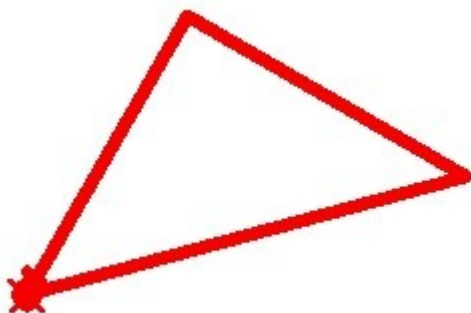
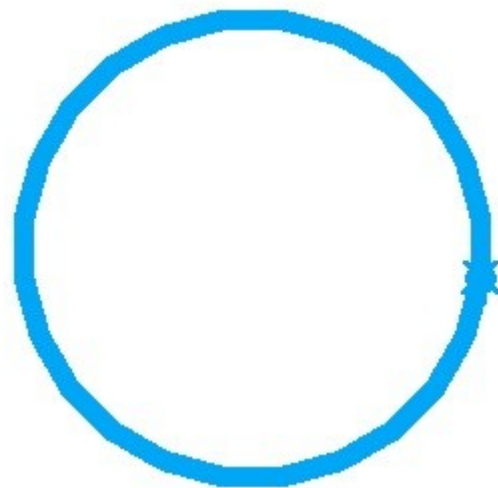
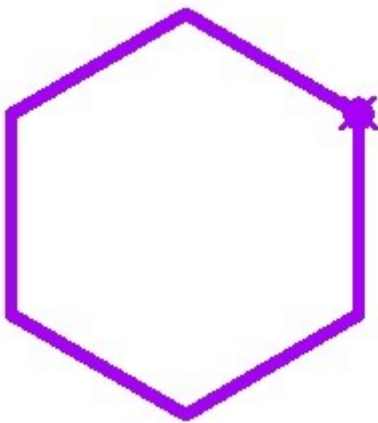
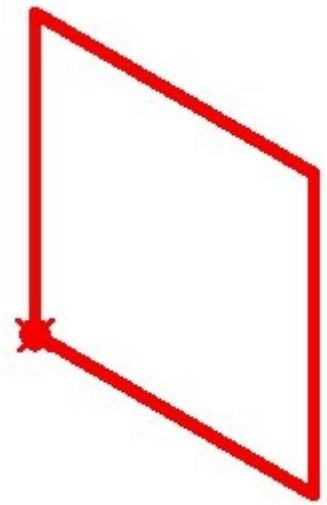
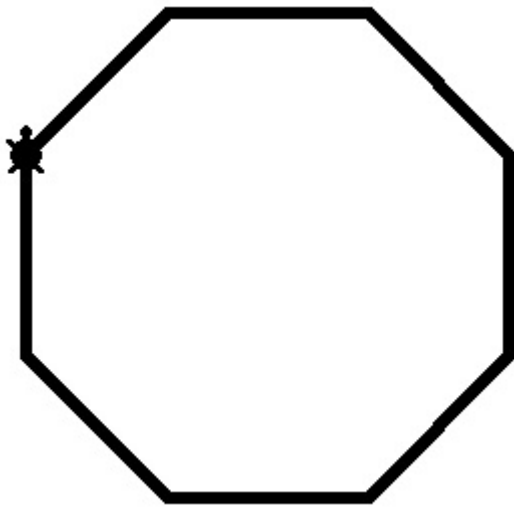


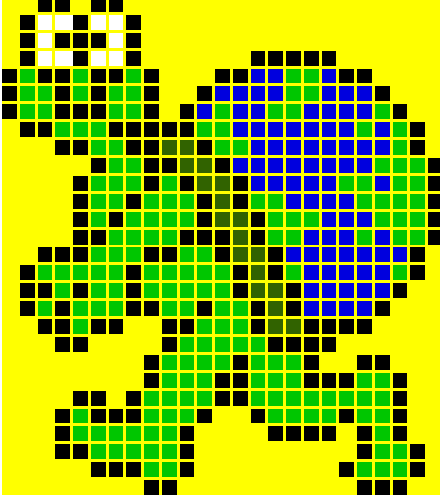


```
when drum is hit  
  change effect color by 25  
  play battery 0.2 times  
  change effect color by -25
```



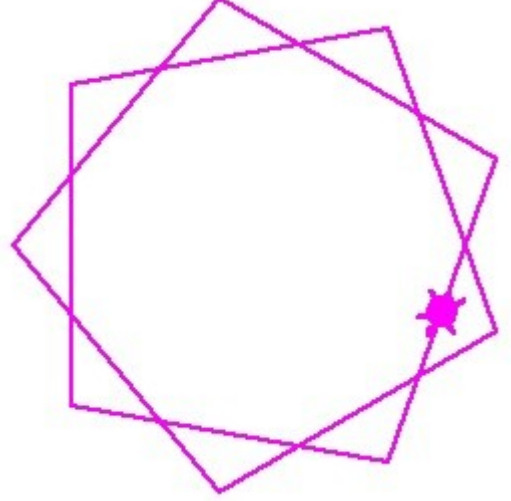
```
when green flag clicked
  set skor to 0
  loop
    choose random from -30 to 30 degrees
    move 5 steps
    if small fish is approaching then
      change skor by 1
      play pop sound
      move -100 steps
```





## MikroDünyalar

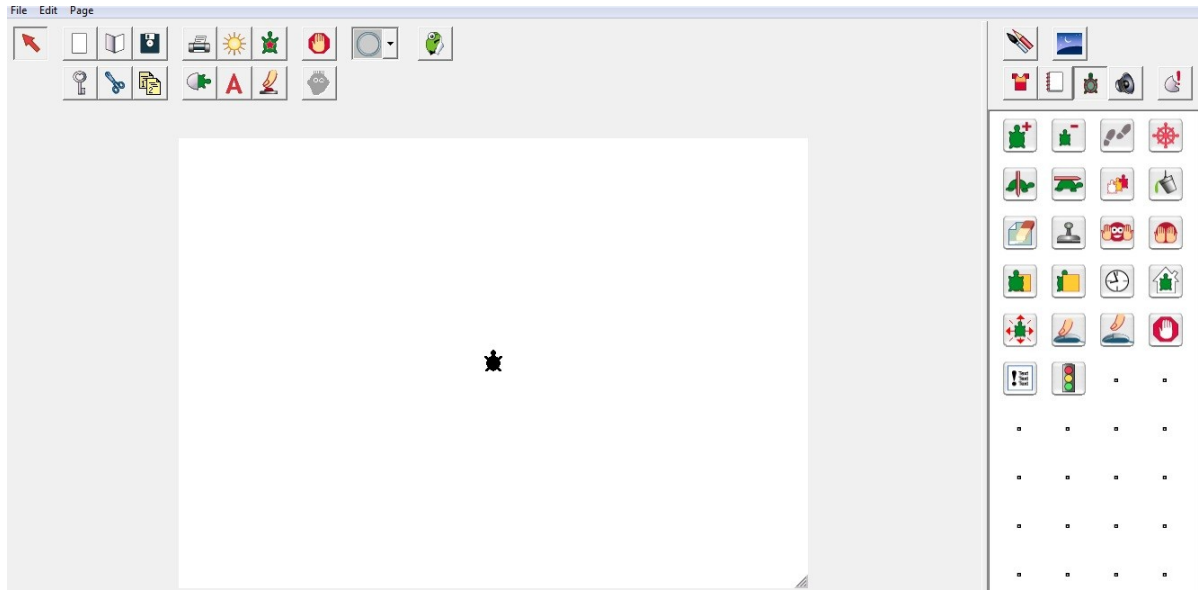
MikroDünyalar okul öncesi eğitimi için tasarlanmıştır interaktif bir öğrenme ortamıdır. MikroDünyalar çocuk dostu sembolleri sayesinde kullanması ve öğrenmesi çok kolaydır.



Çocuklar bu semboller ile okuma-yazma bilgisine gerek duymadan yazdıkları basit bilgisayar programları ile interaktif projeler yaratarak geometri, düzen, organizasyon, hareket gibi önemli matematiksel ve bilimsel kavramları keşfedebilirler.

Çocuklar MikroDünyaların baş kahramanı Kaplumbağayı hareket ettirerek (programlayarak) bilgisayarı bir yaratma ve düşünme aracı olarak kullanmasını öğrenirler.

MikroDünyalar kullanması çok kolay bir arayüze sahiptir.



Yan taraftaki resim Kaplumbağa Komuta Merkezidir. Çocuklar:

**Kalemi bastır**



**Yürü**

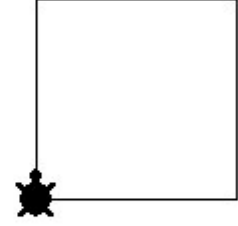
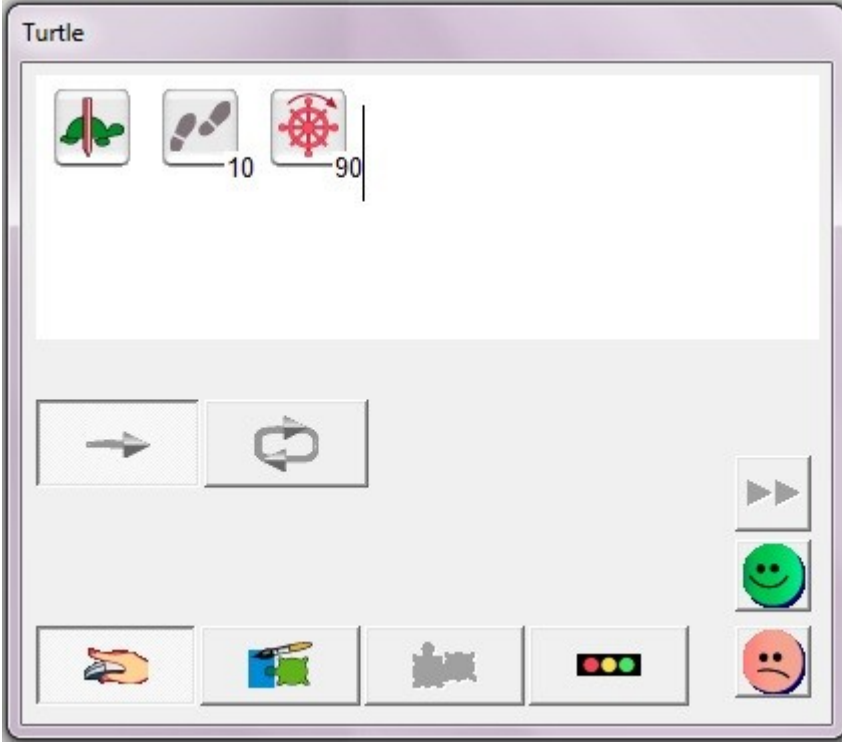


**Dön**

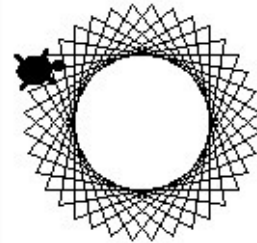
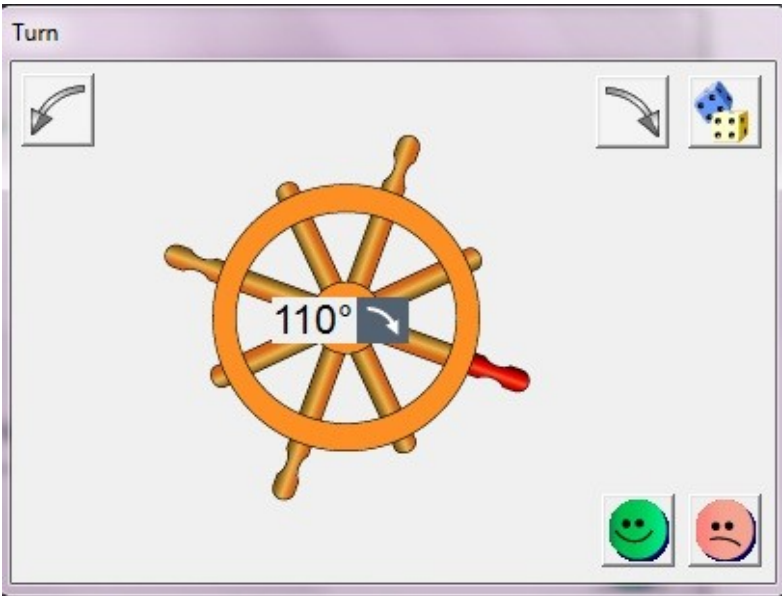


komutlarını tıklayıp Kaplumbağanın çantasına göndererek (programlama) Kaplumbağayı canlandırıp, bir kare ya da çok daha değişik ve karmaşık geometrik şekiller çizdirebilir, animasyonlar ve basit bilgisayar oyunları tasarlayabilirler.

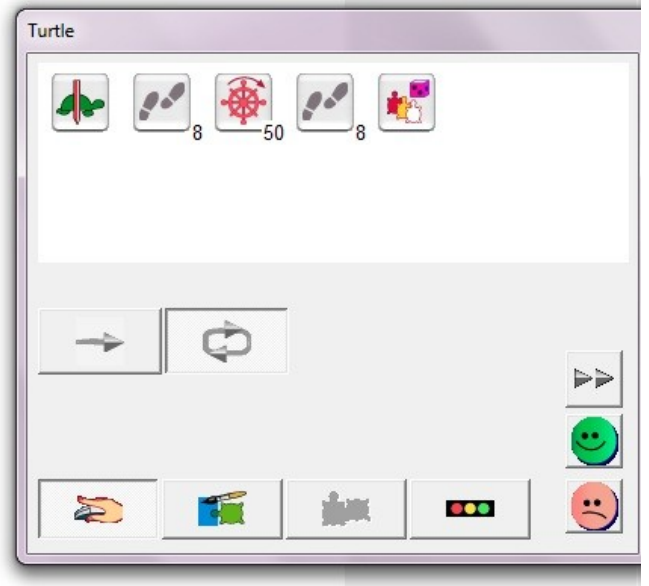
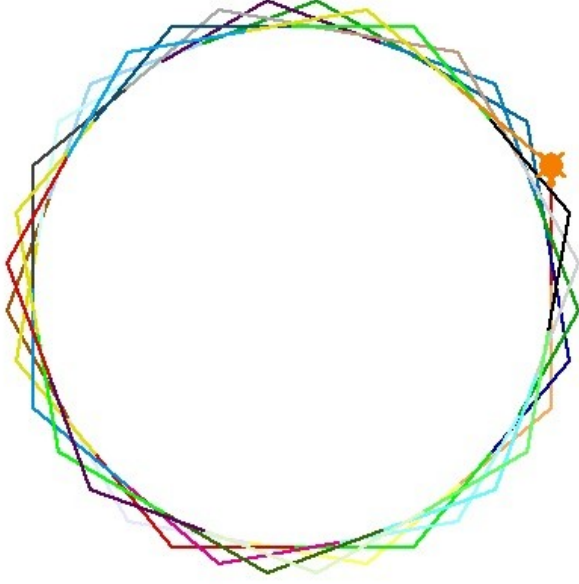




MikroDünyalar çocuk dostu düğmeleri ve ekranı sayesinde çocuklara yaptıkları projeler üzerinde kolaylıkla değişiklikler yapabilmeye ve yeni fikirleri deneme fırsatı verir. Eğer kaplumbağa kalemini bastırıp, 10 adım gidip, 90 derece yerine 110 derece dönerse nasıl bir şekil ortaya çıkar.



Çocuklar MikroDünyalar ile kısa zamanda daha karmaşık fikirleri hayata geçirebilirler.

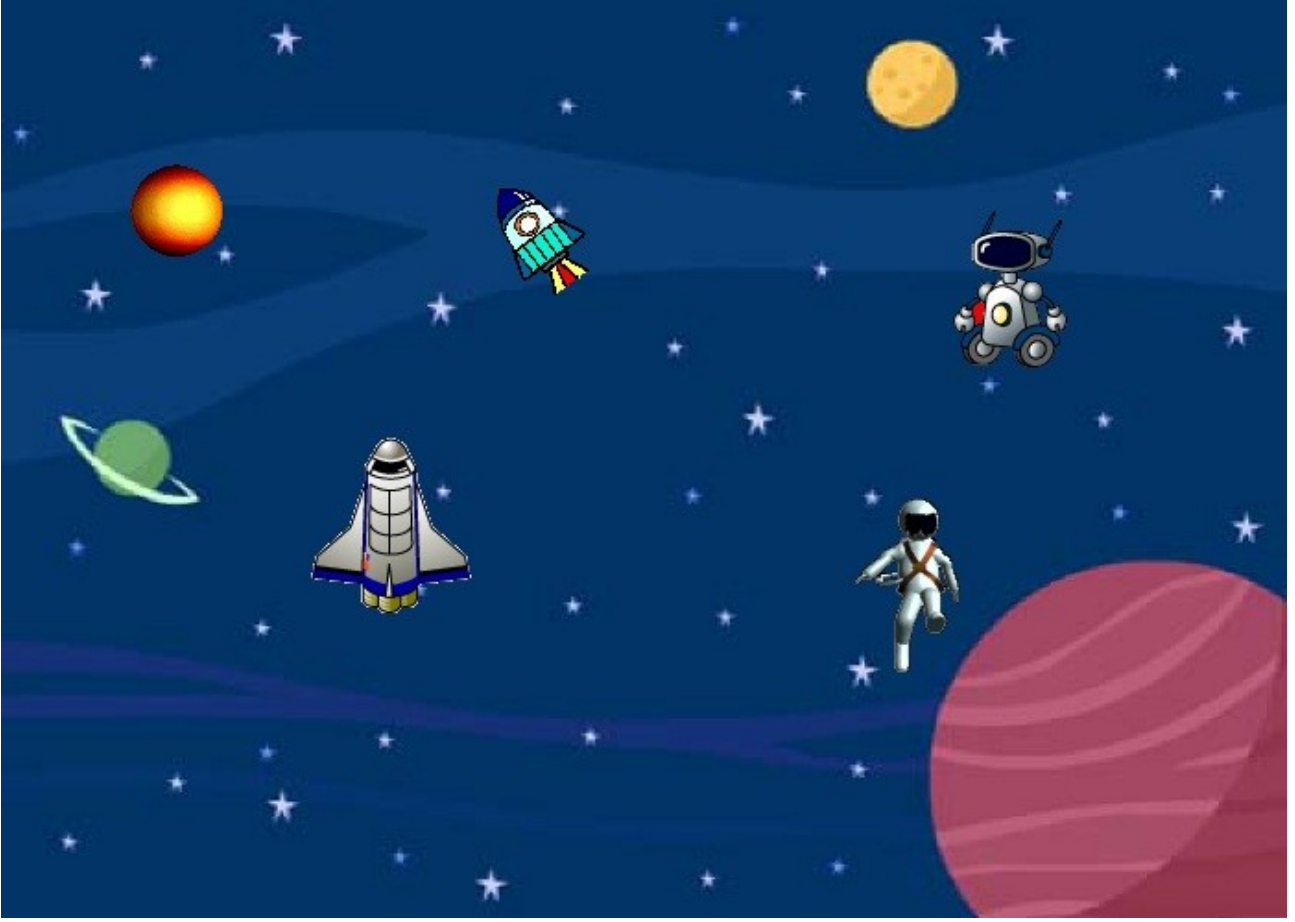


Bilgisayar programlamasını öğrenmenin faydaları nelerdir?

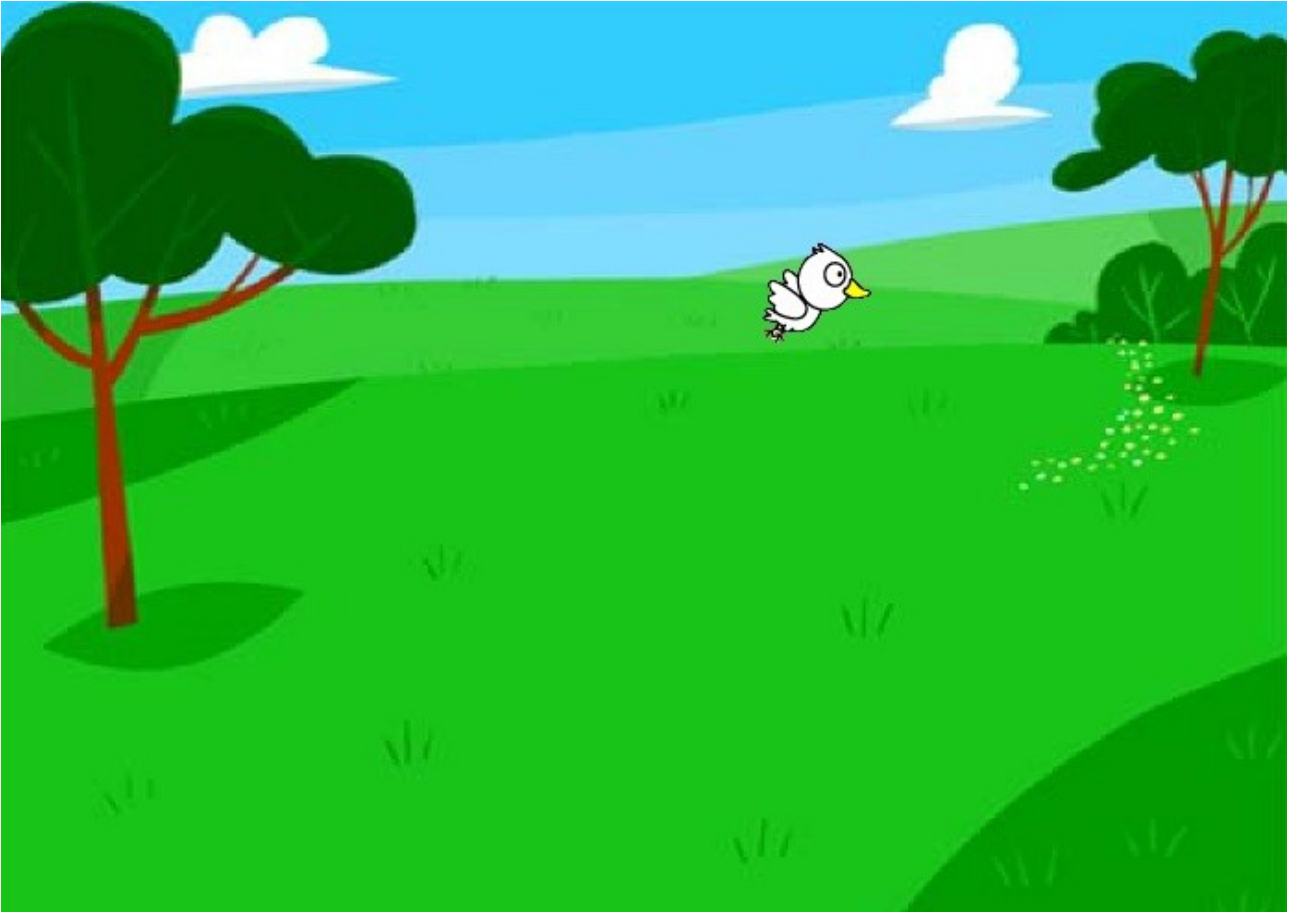
- Yaratıcılık
- Problem çözme
- Eleştirel düşünce
- Mantıksal düşünme

Ekrandaki kontrol edilebilinen Kaplumbağa çocukların gerçek ve somut dünyaları ile bilgisayardaki soyut dünya arasında bağlantı kurar. Yaratma ve öğrenme süreçlerini kendi kontrolleri eline alan çocuklar kendi ilgilerini çeken projeler yaratarak kendi bilgilerini aktif olarak yaratırlar.

Çocuklar isterlerse uzayın derinliklerinde bir astronot olarak maceralar yaşayabilirler...



Ya da tıklandığı zaman uçan bir kuş animasyonu yapabilirler.





MikroDünyalar çocukların projeye başlama ve bitirme beceri ve disiplinine sahip olmalarını sağlar. Kişisel disiplin, gerektiği zaman takım çalışması ve zaman yönetimi önemli kazanımlar arasındadır.

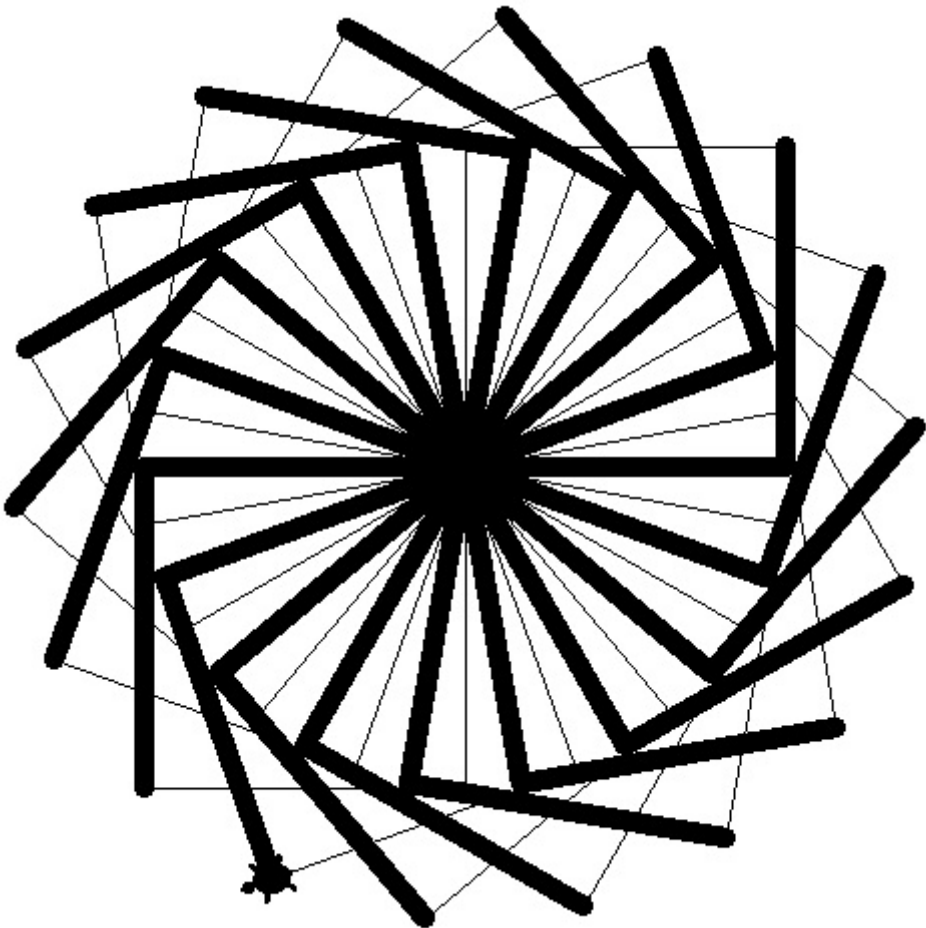
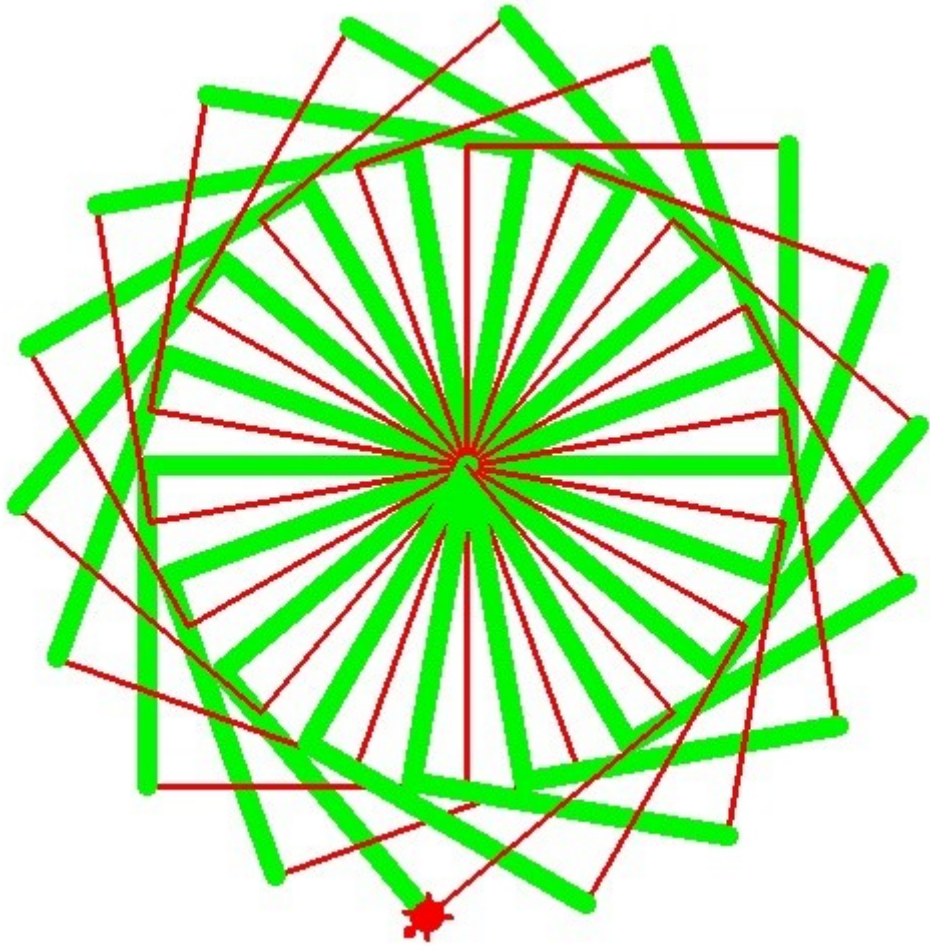
Bununla beraber çocuklar kendilerini özgür ve yaratıcı bir şekilde ifade ederler. Yapılan kişisel projeler çocukların iletişim becerilerini geliştirmekte önemli bir rol oynamaktadır. Çocuklar boyama araçları ile kendi hikayelerini anlatacakları arka planlar yaratabilir, değişik hikayeler için değişik karakterler kullanabilir ya da hikayelerini daha heyecanlı ve hareketli bir hale getirmek için kendi şarkılarını yazabilir, ses ve müzik dosyalarını kullanabilirler.

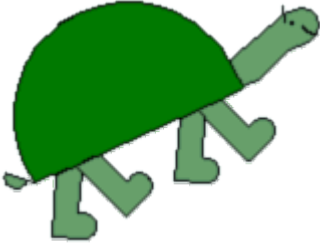
### **Neden MikroDünyalar?**

MikroDünyalar yapısalıcı eğitim teknolojileri için çok güzel bir örnektir. MikroDünyaların temel felsefesi çocuğun kendi bilgisini aktif olarak yaratması ve yaratılan bu bilgiyi aktif olarak kullanmasıdır.

MikroDünyalar çocukların tasarlama becerilerini geliştirerek aktif olarak keşifler yapmalarını, fikirlerini test etmelerini ve kendilerini ifade edebilmelerini sağlar. Çocuklar bunları MikroDünyalar ile interaktif hikayeler, animasyonlar ya da bilgisayar oyunları tasarlayarak gerçekleştirir.

Tasarlama becerisi aynı zamanda çocukların problem çözme, eleştirel düşünme ve hayat boyu öğrenme becerilerini de geliştirir. Yeni tasarımlar yeni fikirleri ortaya çıkarır, bu sırada ortaya çıkan hatalardan geliştirilen yeni fikirler öğrenmeyi devamlı bir döngü haline getirir.





## Logo Programlama Dili

Logo programlama dili yapısalci eğitim felsefesine göre tasarlanmış, yapısalci öğrenmeyi destekleyen bir öğrenme ortamıdır. Yapısalci bilgi teorisinde öğrenme çevre ile kurulan ilişkilerin öğrenenin zihinde öğrenenin de katılımı ile aktif olarak kurulması ile gerçekleşir.

Logo Programlama dili MIT Medya Laboratuvarı kurucularından olan Seymour Papert öncülüğünde yaratılmıştır. Matematikçi olan Seymour Papert yapısalci teorisinin kurucularından Piaget ile 5 yıl beraber çalışmış ve doğal olarak fikirlerinden derinden etkilenmiştir. Papert daha sonra MIT Medya Laboratuvarında Marvin Minsky ile beraber yapay zeka çalışmalarına öncülük etmiştir.



Papert'ın 40 yıl önce 100.000 dolarlık bilgisayarlar ve Kaplumbağa robotları kullanarak yarattığı Logo dilini bugün Lego robot setleri, Scratch, StarLogo ve sayıları yüzlerce olan diğer Logo programları olarak değişik şekillerde karşımıza çıkmaktadır. 100.000 dolarlık bilgisayar ve robot kaplumbağanın fiyatı Birkaç yüz dolara kadar inmiştir.

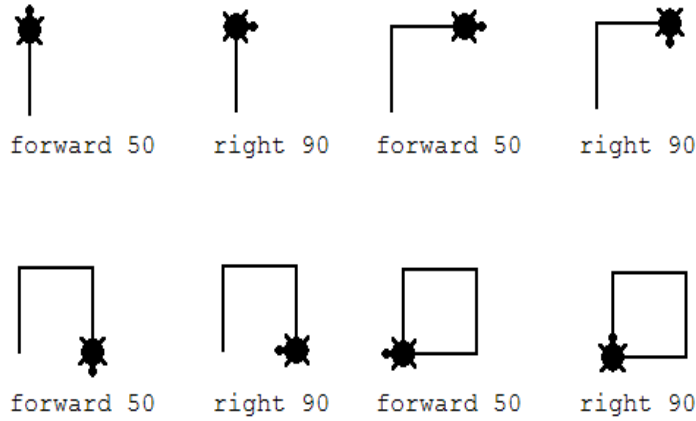
1980 yılında çıkan Mindstorms (beyin fırtınası) kitabında Papert çocukların Logo dili kullanarak bilgisayarlar ile kurdukları mikrodünyaları anlatır. Logo ile çocuklar kendi mikrodünyalarını tasarlayarak bilgisayarı interaktif ve kişisel bir öğrenme aracı haline getirirler. Papert'a göre Logo "bilgisayarların çocukları değil", "çocukların bilgisayarları programladığı" bir öğrenme ortamı hazırlar.

Kaplumbağa Geometrisi

Seymour Papert'a göre Kaplumbağa Geometrisi değişik bir geometri çalışma biçimidir.

Seymour Papert'in yaptığı şey çok basittir. Öklid geometrisinde pozisyonu olan ama eni, boyu, yüksekliği ve hacmi olmayan "nokta", Papert'in Logo dili ile bir Kaplumbağa haline gelir.

Öklid geometrisinde noktanın sadece pozisyonu vardır. Kaplumbağa geometrisinde nokta yerine Kaplumbağanın sadece pozisyonu değil ayrıca bir yönü vardır ve hareket edebilir. Çocuk bilgisayardaki ya da robot kaplumbağayı hareket ettirerek Kaplumbağayı canlandırabilir ve soyut matematik dili çocuğun kendi düşünce ve gerçekleri ile örtüşen interaktif bir dil haline gelir. Çocuk kendisini Kaplumbağanın yerine koyabilir.



© 2000 Logo Foundation

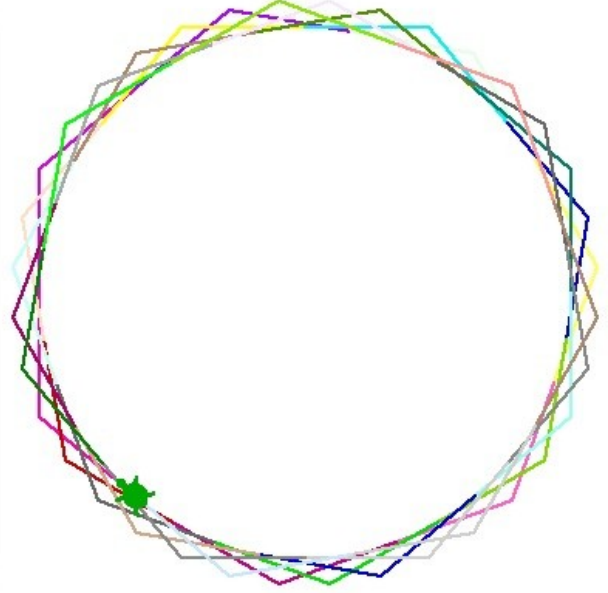
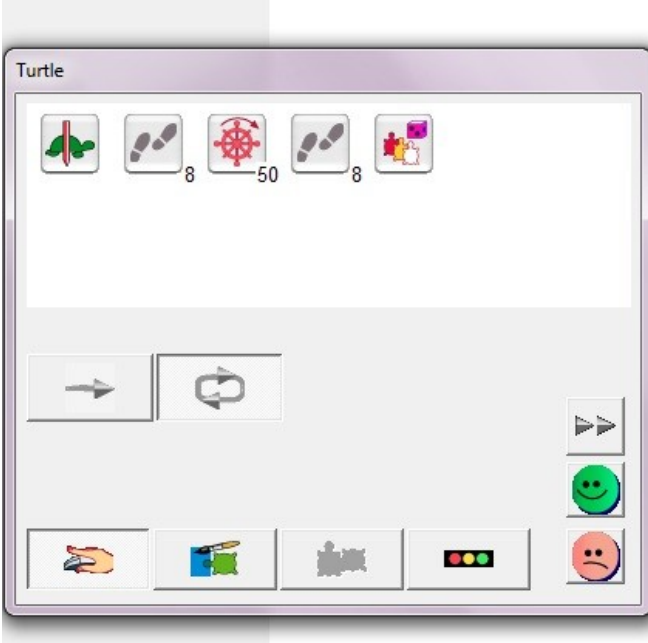
Papert'in "Kaplumbağa dili" adını verdiği bu dili öğrenmesi ve kullanması çok kolaydır. Bilgisayar arayüzlerindeki gelişmelerle bu dil daha da kolay ve ulaşılabilir hale gelmiştir. Klasik Logo dilinde Kaplumbağaya bir kare çizdirmek için şu komutlar gerekir:

İLERİ 50  
SAĞ 90  
TEKRAR 4

Scratch ya da MikroDünyalar gibi yeni ve gelişmiş Logo dilleri ile aynı kareyi çizmek



ya da daha karmaşık projeler yapmak daha kolaydır.



## Nasıl?

Çocuklar bu dili kullanarak yarattıkları ve kendileri için anlamlı olan mikrodünyalar içinde hikayeler anlatabilir, animasyonlar ya da bilgisayar oyunları tasarlayabilir, geometrik şekiller çizebilir ve bütün bunlarla beraber önemli matematiksel kavramları ve ilişkileri kendi kendilerine keşfederek öğrenirler. Çocukların kontrolü altında olan kaplumbağa aynı zamanda çocuklara kendi öğrenme süreçlerini de kontrol altına almalarını sağlar.



Matematik (Yun. "Matematika") Pisagor Okulu üyeleri tarafından, "öğrenilmesi gereken her şey anlamında kullanılmıştır. Papert'e göre "matetik" öğrenme sanatı ya da öğrenme hakkındaki bilgidir. "Matefobi" matematik ve öğrenme korkusu, "matofil" ise matematik ve öğrenme sevgisidir. Yapay olan okul matematiğinin beşeri bilimlerden ayrılması ile Papert'e göre öğrencilerin çoğu matefobik olmuşlardır.

Logo dili açık uçlu bir öğrenme ortamı sağlar. Okul matematiği çocuklara bilgiyi



basitleştirerek parçalanmış zaman ve konular içinde verirken Logo dili ile çocuklar problemlerin bütünlüğünü ve çeşitliliğini fark ederek çok daha karmaşık problemler üzerinde çalışabilirler. Logo yeni fikirlerin ve problem çözme yöntemlerinin kolaylıkla denenmesini sağlar. Öğrenme sadece yeni bilgi edinmeyle değil aynı zamanda var olan bilginin yeniden organize edilmesi ile gerçekleşir.

### PAPERT İLKESİ

**Öğrenme sadece yeni bilgi edinmeyle değil aynı zamanda varolan bilginin yeniden organize edilmesi ile gerçekleşir.**

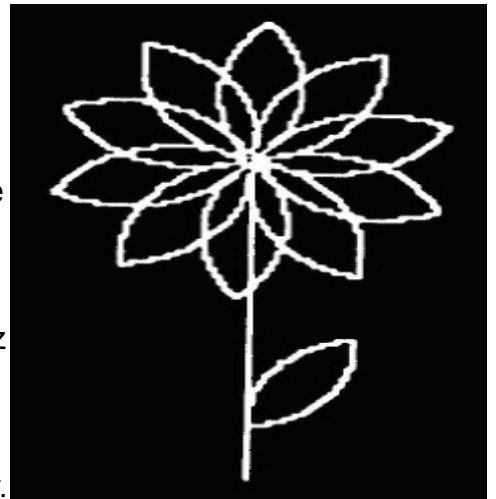
Arie De Geus "Yaşayan Şirket" kitabında Logo dilinin şöyle anlatır:

Papert o sıralarda henüz yeni yeni yaygınlaşmakta olan kişisel bilgisayarları çocuklar için öğrenme amaçlı bir geçici nesne olarak kullanıyordu. Papert'in geliştirilmesinde katkıda bulunduğu bir bilgisayar dili olan "Logo"yu kullanarak, çocuklar mekanik bir kaplumbağayı (veya ekrandaki bir kaplumbağayı) programlayıp çeşitli yönlere doğru hareket ettiriyor, ya da onu kullanarak kalıplar çizdiriyorlardı. Papert'in verdiği isimle, kaplumbağalar dünyası bir mikrodünya olmuş ve çocukların gerçek dünya hakkında bilgiler öğrenmelerine yardımcı dokunmuştu. Logo programlama deneyimi, insanın düşleyebileceği en saf oyuna yakın bir şeydi. Örneğin, kaplumbağanın hareketini, konum ve hızı belirleyen komutlar kullanmak yoluyla kontrol ederek, devinim yasaları gibi Newton fiziğinin çeşitli kavramlarını derinlemesine, -Piaget'nin kullandığı terimlerle söyleyecek olursak- "uzlaştırılmış" bir anlayış geliştiriyorlardı. Seymour Papert'i okumak ve onun mikrodünyalarını bir başka çeşit geçici nesne olduğunu anlamak son derece kolaydı.

### Kompütasyonel Geometri

Kaplumbağa geometrisi kompütasyonel geometridir. Kaplumbağa geometrisinin değişik bir geometri çalışma biçimi olmasının temel sebebi budur. Kompütasyon kısaca bilgisayarları problem çözme aracı olarak kullanmaktır. Kompütasyon insan zihninin iyi yaptığı şeyleri insan zihniyle, bilgisayarların iyi yaptığı şeyleri de bilgisayarla yapmaktır.

Kompütasyonel düşünme becerileri ile çocuklar daha önce daha önce başa çıkamayacaklarını düşündüğümüz sistemleri tasarlayabilir ve problemleri çözebilirler. Endüstriyel devrimin temel becerileri olan okuma-yazma ve aritmetik bilgilerinin üstüne bilgi çağının kompütasyonel düşünme becerileri mutlaka eklenmelidir. Matbaanın bulunmasıyla hızla yaygınlaşan okuma-



yazma ve aritmetik becerileri gibi bilgisayarlar da kompütasyonel düşüncenin hızla yaygınlaşmasını sağlayacaktır.

## 21. Yüzyıl Becerileri

1. **Karmaşık Hikâye Anlatımı:** Öğrenci sistematik olarak ilişkiler, düzen ve süreçler üzerinden düşünebilmeli ve sebep ve sonuç ilişkilerini bir ağ (network) içindeki bağlantılar üzerinden kurabilmelidir. Örneğin öğrenci sosyal bilimler çalışırken yerel bir ekonomiyi ya da biyoloji çalışırken bir hücreyi senaryo haline getirip bu sistemler içindeki ilişkileri karmaşık bir hikâye haline getirip anlatabilmelidir.
2. **Çok Disiplinli ve Disiplinler Arası Düşünme:** 21. Yüzyılda uzmanlık tek bir disiplin üzerinden değil değişik disiplinlerin bir araya gelmesinden oluşur. Disiplinler arasındaki bağlantılar görülerek daha önce çözümü bulunamayan ya da keşfedilmeyi bekleyen yeni çalışma alanları bulunabilir.
3. **Dijital Okur-Yazarlık:** Dijital okur-yazarlık teknolojinin sadece pasif olarak kullanılmasının ötesinde öğrencinin teknoloji tasarlamasını ve pratik uygulamalarla günlük hayatla ilgili sorunlara teknolojik çözümler getirmesini öngörür. Öğrenci ileri teknoloji araçlarını bilimsel araştırma süreci içerisinde gerektiği şekilde tararlar ve verimli bir şekilde kullanabilir.
4. **Sistem Okur-Yazarlığı:** Sistem okur-yazarlığı parçaların yerine bütün olanı görebilmek, bu bütün içindeki karmaşık ve dinamik ilişki ağlarını uzun vadeli sebep-sonuç ilişkileri ile açıklayabilmek becerisidir. Sistem okur-yazarlığı sistemlerin davranışları hakkındaki öngörülerimizi (zihin modelleri) daha sağlam temeller üzerine oturtarak daha güvenilir teknolojik ve sosyal sistemler kurulmasına yardımcı olur.
5. **Dijital Pasaport:** Karmaşık küresel ilişki ağları ve “kişisel kitle iletişim araçları” ile küresel mesafeler kısalmış ve öğrencilere herhangi bir konu ya da ilgilerini çeken olaylar hakkında dünyanın başka bir ucundaki başka bir kişi ile görüş alış verişini yapma imkânı ortaya çıkmıştır.

## Anahtar Kelimeler

**Matematik:** Matematik bilimsel, ekonomik ve sosyal gelişmenin motorudur. Pisagor okulu matematiği “öğrenilmesi gereken her şey” anlamında kullanır.

**Matetik:** Öğrenme sanatıdır.

**Matefobi:** Matematik ya da öğrenme korkusudur. Endüstriyel eğitim modelinin çocukların canlı ve hayal gücü geniş dünyalarına uymayan soyut matematik dili çocuklarımızı matematikten ve öğrenmekten korkar hale getirmiştir.

**Matofil:** Matematik ya da öğrenme sevgisidir. Öğrenci merkezli eğitim ve çağdaş eğitim teknolojileri ile çocuklarımız öğrenme sürecinin aktif katılımcıları haline gelerek matematik ve öğrenme sevgisini hayat boyu içlerinde taşıyabilirler.

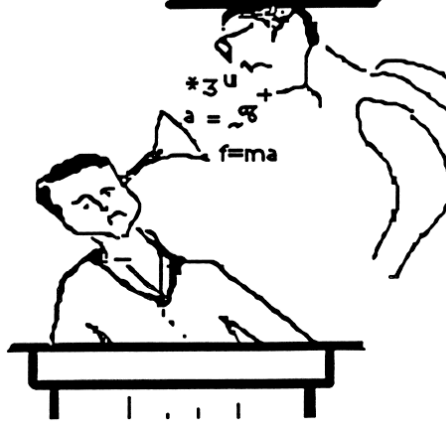
aktif öğrenme projeleri tarafından yapılarak yeni fikirler insanların sadece kendilerini eğitim için kullanılmaları değil, öğrenme sürecinde yaratıcı fikirler üretmelerini sağlar. Seymour Papert'in 'Logo' programı, çocukların kendi fikirlerini ifade etmelerini sağlar. MIT Medya Laboratuvarı tarafından geliştirilen 'Scratch' programı, çocukların kendi fikirlerini ifade etmelerini sağlar. Seymour Papert'in 'Logo' programı, çocukların kendi fikirlerini ifade etmelerini sağlar. MIT Medya Laboratuvarı tarafından geliştirilen 'Scratch' programı, çocukların kendi fikirlerini ifade etmelerini sağlar.



## Tasarım İle Öğrenme

### Mitchel Resnick (MIT Medya Laboratuvarı)

İnsanlar öğrenme ve eğitim hakkında düşündükleri zaman bir kişinin başka bir kişiye bilgi iletmesini düşünürler.



Ama eğitimciler giderek “iletme yaklaşımının” işe yaramadığının farkına varıyor. İnsanların en iyi şekilde öğrenebilecekleri ortam sadece pasif olarak bilgi alımıyla değil, aktif olarak keşfetme, deney yapma ve kendilerini ifade edebilmeleri ile sağlanır.

Daha fazla okul “yaparak öğrenme” üzerine odaklanırken, uygulamalı faaliyetleri ile de öğrencilerin ilgisini çekmeye çalışıyor. Amacımız öğrencilerin tasarım ile öğrenmeleri, yaratmaları ve bir şeyler icat etmeleri. Amaç sadece yaparak öğrenme değil; tasarım ile öğrenme.

### Neden Tasarım?

- Tasarım projeleri çocukların aktif katılımcılar olmalarını sağlarken, öğrenme sürecini kendi kontrol ve sorumlulukları altına almalarını sağlar.
- Tasarım projeleri yaratıcı problem çözme teşvik eder.
- Tasarım projeleri genellikle sanatsal, teknolojik, matematiksel ve bilimsel düşünceyi kapsadığı için disiplinler arasıdır.
- Tasarlanan projelerin başkaları tarafından da kullanılabileceğini düşünen çocuklar kendilerini başka insanların yerine koymayı öğrenirler.
- Tasarım projeleri derin düşünme ve işbirliği fırsatları sunar.
- Tasarım projeleri öğrenmede pozitif geribildirim döngüsü yaratır: çocuklar yeni şeyler tasarladıklarında yeni fikirler bulurlar, bu fikirler yeni tasarımları ortaya çıkarır, bu tasarımlardan da tekrar yeni fikirler ve tasarımlar ortaya çıkar.

## Yapısalcılık ve Yapıcılık

Öğrenme ve eğitimdeki "tasarım ile öğrenme" yaklaşımı iki önemli teoriden esinlenmiştir. İsviçreli psikolog Jean Piaget tarafından geliştirilen *yapısalcı* teori öğrenmeyi bilginin yeni tecrübelerle aktif olarak kurulması olarak görür. Bu teoriye göre insanların aklına fikirler gelmez, insanlar fikirleri yapar. Yapısalcı teori birçok eğitim reformu girişiminin temelini oluşturur.

Massachusetts Institute of Technology (MIT) profesörlerinden Seymour Papert tarafından geliştirilen *yapıcı* yaklaşıma göre insanlar yeni bilgiyi, özellikle de yeni şeyler yaparken, dünya ile kurdukları ilişkinin sonucunda elde ederler. Bilgisayar programlayabilirler, kumdan kaleler ya da LEGO robotlar yapabilirler. Önemli olan insanların kendileri ya da çevreleri için anlamlı olan bir şeyler yaratmak için aktif olarak çalışmalarıdır.

## Öğrenme Devrimi Yaratmak

Nicholas Negroponte, Mitchel Resnick, Justine Cassell  
(MIT Medya Laboratuvarı)

Çocuklar gelecektir. Eğer dünyanın önemli problemlerini çözmeyi umuyorsak- dünya barışı, sağlıklı hayatlar, ekonomik gelişme, küresel sürdürülebilirlik- dünya çocuklarına daha zengin öğrenme fırsatları sağlamalıyız. Şüphesiz eğitilmiş ve yaratıcı bir nüfus küresel sağlık, zenginlik ve barışa giden en iyi yoldur.

Ama bugünün dünyasında eğitim uygulamaları yaygın olarak çağdışı kalmıştır. Bilimsel ve teknolojik gelişmelerin tarım, tıp ve endüstriyel sektörleri köklü değişikliklere uğratmalarına rağmen, çocukların öğrenme şekilleri geçmiş yüzyılların fikirlerine bağlı kalarak değişmemiştir.

Yeni dijital teknolojiler çocukların öğrenmesi ve eğitimi için radikal ve küresel ölçekte tarihsel fırsatlar ortaya koymaktadır. Biyoteknolojik gelişmelerin tarımda "yeşil devrimi" mümkün kıldığı gibi, şimdi de dijital teknolojiler "öğrenme devrimini" gerçekleştiriyor. Bize göre bu yeni dijital teknolojiler sadece çocukların nasıl öğrendiklerini değil, ne öğrendiklerini ve kiminle öğrendiklerini de değiştiriyor.

**Çocuklar Nasıl Öğrenir:** Dijital teknolojiler çocukların daha aktif ve bağımsız olmasını sağlar. Çocuklar doğrudan keşif, ifade ve deneme yoluyla öğrenmeyi kendi sorumlulukları altına alırlar. Odak "öğretilmeden", "öğrenmeye" kaymıştır.

**Çocuklar Ne Öğrenir:** Çocukların okulda öğrendiklerinin çoğu kâğıt-kalem dönemi için tasarlanmıştır. Yeni dijital teknolojiler ile çocuklar dijital çağ öncesinde çok karmaşık olduğu düşünülen projeler ve kavramlar üzerinde çalışabilirler.

**Çocuklar Kiminle Öğrenir:** Küresel bağlantılar yeni "bilgi-inşa toplulukları" oluşturularak çocukların (ve yetişkinlerin) projeler üzerinde beraber çalışmalarını ile birbirlerinden öğrenerek küresel işbirliği sağlanabilir. Bunun için öğrenime çok kültürlü, çok disiplinli ve çok kanallı yeni bakış açıları getirilmelidir.

Bu değişimler kendiliğinden olmayacaktır. Her ne kadar dijital teknolojilerin ucuzlaması ile çocukların bu teknolojilere ulaşması daha kolay hale gelse bile, internet ve bilgisayar ulaşımı istenen seviyede değildir. Çocuklar için üretilen çoğu yazılım ürünü çocukların geniş ufuklarını genişletmek yerine daraltmaktadır. Gerçek bir eğitim devrimi yaratmak için yeni öğrenme ve çocuk vizyonunu destekleyen teknolojiler üretmeliyiz.

## Ana İlkeler

**Doğrudan Keşif:** Geleneksel görüş çocukların anaokuluna kadar dünyayı doğrudan (emekleyerek, dokunarak, çiğneyerek) öğrendikleri ama daha sonra ileri kavramların onlara "öğretilmesi" gerektiği şeklindeydi. Amacımız çocukların daha karmaşık fikirleri keşfetmeleri ve deneyim kazanmalarını devam ettirecek dijital teknolojiler geliştirmektir. Örneğin uzak köylerde yaşayan çocuklar yaşadıkları yerdeki zirai faaliyetlere İnternet'te bağlanarak ya da yerel sensörler ile toprak, hava ve bitki örtüsünün kalitesini ölçüp deneyler yaparak katkı sağlayabilirler.

**Doğrudan İfade:** Yeni kitle iletişim araçları büyüklerin çocuklar yerine konuşmaları yerine çocukların kendi hikâye ve fikirlerini daha iyi anlamalarını ve daha geniş bir izleyici kitlesine anlatabilmelerine yardımcı olacaktır. Klasik bakış açısı kendi fikirlerini ifade etmeleri yerine fikirleri büyüklerden “almaları” yönündeydi. Hatta çocukların kendileri ve kültürleri hakkındaki bilgiler bile büyüklerden gelmekte. Amacımız geleneksel bakış açısının ötesine geçip çocukların kendilerini hikâyelerle, iletişimle, tasarımla ve yeni yollar olarak kendi sesleri ile kendilerini ifade edebilecekleri dijital teknolojiler geliştirmektir.

**Doğrudan Deneyim:** Gelecekte çocuklar yaşadığımız dünya hakkında bilgilenmek için sadece anne ve babalarına bağımlı olmayacaklardır. Bunun yerine bütün dünya ile kendi kişisel bağlantıları ile doğrudan ilişki kuracaklardır. Elektronik gözler ve kulaklar sayesinde dünyanın öteki tarafının nasıl görüldüğünü ve seslerini duyacaklardır. Bu tecrübe ulusal sınırların önemini azaltacak ama yerel kültürler, çocukların okul bahçelerinde ve evlerinde yaşadıkları önemini koruyacaktır. Belki de en önemlisi çocukların kendilerini yaşadıkları gerçek ve sanal toplumlar için önem taşıyan entelektüel temsilciler olarak hissetmeleri olacaktır. Çocuklar kendilerini coğrafyaları, kültürleri, dilleri ve yaşlarının ötesinde ifade etmeye alışacaklardır.

**Çok Kültürlülük:** Geliştirilen çoğu teknoloji sınırlı tarz ve yaklaşımları destekler. Küresel bağlantılar bir ihtiyaç ve fırsat olarak daha kapsayıcı yaklaşımları ve bütün kültürlerden çocukların katılımını gerektirir. Amacımız farklı katılım ve kullanim ihtiyaçlarını karşılarken bütün dünya çocuklarının kendi kültürlerini paylaşmaları ve diğer kültürleri öğrenmelerini sağlayan dijital teknolojiler geliştirmektir.

**Çok Dillilik:** Bugüne kadar dünyada konuşulan dillerin çeşitliliği küresel bir toplumun oluşturulması karşısında büyük bir engel olarak algılandı. Küresel bağlantıların oluşması ile çocukların “ortak bir dil” konuşmalarına büyük bir ihtiyaç varken, daha büyük fırsat çocukların birbirlerinin dillerini ve genel olarak dil öğrenmeleridir. Amacımız çocukların dilsel sınırlar ötesinde kendi aralarında iletişim kurmalarını sağlayacak, yabancı dilleri öğrenirken kendi dillerine değer katacak yeni araçlar geliştirmektir.

**Çok Kanallılık:** Çocuklar ve bilgisayarlar arasındaki iletişim kanalları çok sınırlıdır: klavye ve fare bir tarafta, ekrandaki yazılar ve grafikler de diğer taraftadır. Bilgisayarların sözsüz ve işaretler ile iletişimi anlamalarını ve kullanmalarını sağlayarak çocuklar ve bilgisayarlar arasındaki iletişimin doğasını zenginleştirebiliriz. Ayrıca sözlü ve sözsüz iletişim kurabilen bilgisayarlar ile bilgisayar kullanımını daha geniş bir yaş yelpazesine ve değişik kültürlere tanıtabiliriz. Henüz klavye kullanamayan ama konuşup işaretlerle anlaşabilen çocuk bilgisayarla iletişim kurabilir ve bilgisayarın kendisiyle kurduğu sözlü ya da sözsüz iletişimi anlayabilir.

**Robotiđi Yeniden Düşünmek: Yaratıcı Mühendislik İle Öğrenme**  
**Margaret Pezalla-Granlund, Science Museum of MN**  
**Natalie Rusk, MIT Media Lab**  
**Mitchel Resnick, MIT Media Lab**  
**Robbie Berg, Wellesley College**

Robotların son yıllarda eğitim faaliyeti olarak kullanılması artmıştır. Öğrencilerine bilgisayar kontrollü robot tasarlama imkânı veren üniversiteler, üniversite öncesi eğitim kurumları ve diğer organizasyonların sayıları çoğalmaktadır. Robot tasarlama ve programlama sürecinde öğrenciler önemli mühendislik, matematik ve bilgisayar mühendisliği kavramlarını öğrenirler.

Özellikle okul sonrası programlar ya da seçime bağlı öğrenme ortamlarında robotik faaliyetleri kız öğrencilere oranla erkekler tarafından daha yüksek bir oranda tercih edilmekte ve cinsiyetler arasında belirgin bir fark oluşmaktadır.

Kız öğrencilerin robotik faaliyetlerine daha fazla katılmaları için bayan rol modeller bulunmalı ve daha fazla kız öğrenciye robot tasarımının önemi belirtilerek ilgilerin çekilmelidir. İlgı çekerek kız öğrencilerin sayılarının arttırılması ve bayan rol modellerin bulunmasının önemli olmasına rağmen gençlerin ilgilerinin çeşitliliğini arttırmak için alternatif yöntemler de gerekmektedir. Robotik faaliyetlerinin tanıtımı ve sunumu hakkında yeniden düşünmemiz gerekmektedir.

Bu kısa makalede özellikle genç kızları teşvik eden başarılı uygulamaları paylaşıyoruz. Bu uygulamalar olamadan bu gençler öğrenme fırsatlarına ilgi göstermeyebilirlerdi.

### **Yarışma Yerine Sergi**

Robotik faaliyetlerinin çođu yarışma olarak düşünölmüştür. Örneđin, First Lego Ligi her yıl yeni bir parkur ile gençlerden oluşan binlerce takımı yerel, ulusal ve uluslararası yarışmalara çağırılmaktadır. Yarışmalar birçok öğrenciyi motive etmekte, ama bazı öğrencileri de robotik faaliyetlerinden sođutmaktadır. Gençlerin çalışmalarını göstermeleri için sergiler yarışmalara alternatif bir yaklaşımdır.

Örneđin, Wellesley Üniversitesindeki "Robotik Tasarım Stüdyosu" aileleri ve ilgili herkesi bir araya getirerek robotik projelerini ve yaratıcılarını bir sanat galerisini açılışı gibi bir araya toplayan bir sergi düzenlemektedir. Serginin açık uçlu formatı çeşitli becerilerin ve yaratıcı ifade şekillerinin ortaya çıkmasını sağlarken, projelerin halka açık bir şekilde sergilenmesi tasarımcı öğrencilerin motivasyonu arttırılır.

### **Kişisel İfade**

Robotik atölye çalışmaları genellikle "robotun parkur üzerindeki engeller arasında manevra yapması" gibi görevler üzerine yoğunlaşmıştır. Tek bir görev üzerine odaklanmak yerine, öğrencilerin kişisel ilgi duydukları projelerle çeşitlilik yaratmanın daha iyi bir yol olduğunu keşfettik. Örneđin bir seferinde öğrencilerden bir oyun parkı tasarımlarını istedik. Günün sonunda önümüze cıvıldaayan kuşlar, yuvarlanan kay kaycılar, interaktif oyun parkı aletleri ve otomatik sulama sistemleri gibi birçok projeye geldi.

## **Sanat ve Mühendisliđi Birleřtirmek**

Tipik bir mühendislik projesi kutudaki yumurtayı kırmadan yere bırakmak gibi bir sorunun çözümlenmesinden oluşur. Biz gençlerin boya makinesi, müzik okuyup yazan bir makine ya da programlanabilir çeşme gibi disiplinler arası projeler ile mühendislik kavramlarını öğrenmede daha istekli olduklarını gördük.

## **Eliři Malzemeleri (Mekanik Parçalarla Birlikte)**

Eliři malzemelerini, mekanik parçaları ve programlanabilir cihazları kız ve erkek öğrencilerin neler tasarlayabilecekleri ve neler tasarlamak istedikleri konusunda daha yaratıcı bir şekilde düşünmelerine yardımcı olabilir. Öğrencilere mekanik parçaların (diřli çarklar, kiriřler, dingiller) yanında boru ya da kumař gibi daha geniş yelpazeli eliři malzemeleri verebiliriz. Verilecek malzemeleri de çalışma konusuna uygun olarak seçeriz- oyun parkı için yapraklar, dallar ve başka doğal malzemeler; interaktif ışık çalışması için buzlu bardaklar, parıltılı ve yansıtıcı kâğıtlar gibi. Ayrıca tanıdık nesnelere yeni fikirlerin ortaya çıkmasına sebep olabilir: modaın geleceđi hakkındaki bir çalışma için gençler eski kemerlerini, eldivenlerini ve botlarını interaktif kıyafetlere dönüřtürmüřtü.

## **Hikâye Anlatımı**

Bazı insanlar şekiller ya da yapı ile ilgiliyken bazı insanlarda hikâye anlatmak ya da drama ile ilgilidir. Robotik faaliyetleri genellikle şekil ve yapı ile ilgili olanları tatmin eder. Ama bazı robotik faaliyetleri her iki tarzı da tatmin eder. Örneđi bilim merkezlerinde bir düzeneđin başka bir düzeneđi tetiklediđi Rube Goldberg mekanizmaları çok popülerdir. Buna benzer bir şekilde bir zincirleme tepki çalışmasında katılımcılar bir hikâye ile başladılar ve hikâyeye uygun giriři, gelişmesi ve sonucu olan bir mekanizma geliřtirdiler.

## **Sanatsal Buluşlar İçin Yeni Teknolojiler**

Robotik teknoloji tasarımı teknolojinin nasıl algılandığının ve kullanıldığını önemli bir şekilde etkileyebilir. Örneđi Lego Mindstorms robot seti robotun labirent içinde yolunu bulması gibi geleneksel projeler için çok uygundur. Biz özellikle sanatsal buluşları desteklemek için Cricket adlı verdiđimiz teknolojiyi geliřtirdik. Cricket seti çok renkli ışıklar ve ses cihazlarından oluşur. Cricket seti beste yapmak, hayvan sesi ya da ritim yaratmak için kullanılabilir. Cricketin amacı tasarlanacak, yaratılacak ve öğrenilecek şeylerin alanını genişletmektir.



## **Sistem Nedir?**

Sistem bir amacı gerçekleştirmek için bir araya gelmiş ve beraber çalışan parçalar bütünüdür.

Birbirleri ile ilişkili, bağlı ve birbirlerini etkileyen elementlerin oluşturduğu karmaşık bütün.

Donella Meadows'a göre:

- Bir sistem parçalarının toplamından daha fazlasıdır.
- Sistem içindeki bağlantıların çoğu bilgi akışı içinde çalışır.
- Sistemin en az belirgin parçası olan amacı genellikle sistem davranışı üzerinde en etkili olanıdır.
- Sistemin yapısı sistemin davranışının kaynağıdır. Sistem davranışı kendini zaman içinde ortaya çıkan olaylar serisi ile gösterir.

## **Sistem Düşüncesi**

Sistem Düşüncesi kısaca sistemler hakkında düşündürmektir. Sistem düşüncesinin temeli sistemin parçalarını, parçalar arasındaki bağlantıları ve sistemin amacını görmektir. Sistem düşüncesi sistemlerin nasıl çalıştığı hakkında düşündürmektir.

Hayatımızın her anını sistemleri içinde yaşamamıza rağmen sistemlerin nasıl çalıştığına dair bilgimiz çok azdır. Sistemlerin genel çalışma ilkelerini bilmek bu sistemler içinde yaşamak ve öğrenmek zorunda olan bizlere büyük kolaylıklar sağlayacaktır.

Bunun için yapılması gereken sistemleri basit sistemlerin ötesinde dinamik sistemler olarak görmek ve düşündürmektir. Dinamik sistemler sürekli değişen sistemlerdir. Bu sistemlerde

sebeup ve sonu farklı zaman ve mekandadır. Bu zelliklerinden dolayı dinamik sistemler karmařık davranıřlar gsterirler. En kk bir deęiřiklięin bile sistem zerinde byk etkileri olabilir. Karmařık sistemler bu sebepten dolayı da beklenmedik davranıřlar ortaya koyar. Scratch'in yaratıcısı Mitchel Resnick ve MIT Medya Laboratuvarından Oren Zuckerman'a gre:

Dnyayı anlamak iin dinamik sistemleri anlamayı ęrenmek nemli bir adımdır. Fakat bunu niversite, lise ve hatta orta ęretimde ęrenmek ok ge olabilir. Bu yařta ocuklar dnyanın nasıl alıřtıęına dair modellerini oktan geliřtirmiř oluyorlar.

ocukların fiziksel dnya ile iliřkileri zerinden dnya ile ilgili kendi modellerini oluřturdukları uzun zamandan beri biliniyor (Piaget 1972). Buna ek olarak yapısalcı arařtırma insanların ilgi duydukları projeler aracılıęı ile dnya hakkında daha etkili olarak ğrendiklerini ortaya koyuyor (Papert 1991; Kafai and Resnick 1996).

Sistem dřncesi becerilerini erken yařlardan itibaren ocuklarımıza kazandırmak ve onları 21. Yzyılın karmařık ve etrefilli problemleri ile bařa ıkabilecek hale getirmeliyiz. řu anki eęitim sistemimiz endstriyel dnyanın basit sistemleri zerine kuruludur. Ama dnya basit ve karmařık sistemlerin bir arada bulunduęu bir sistemdir.

Fiziki Murray Gell-Mann bu konuyu řyle aıklıyor:

"Bu konu iin bulduęum isim Yunanca "bklmř" ya da "rlmř" anlamına gelen plektos kelimesinden tremiř plectics'dir. İngilizcedeki complexity (karmařık) kelimesinin kk de Latince orijinal anlamı "beraber rlmř" olan complexus kelimesinden tremiřtir. Plektos kelimesi aynı zamanda Latince orijinal anlamı "bir kere katlanmış" olan simplex kelimesi ile uzaktan baęlantılıdır. Bu kelimedenden de İngilizce simple (basit) kelimesi doęmuřtur. Bylece plectics kelimesi bizim hem basitlik hem de karmařa ile uęrařtıęımızı gsterir. "

### **Scratch ile Sistem Dřncesi Becerileri**

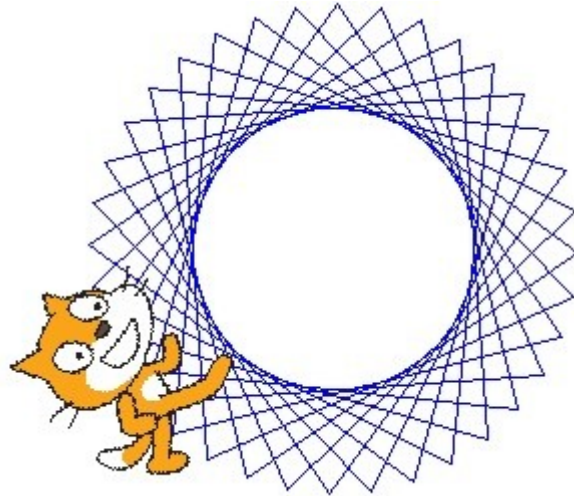
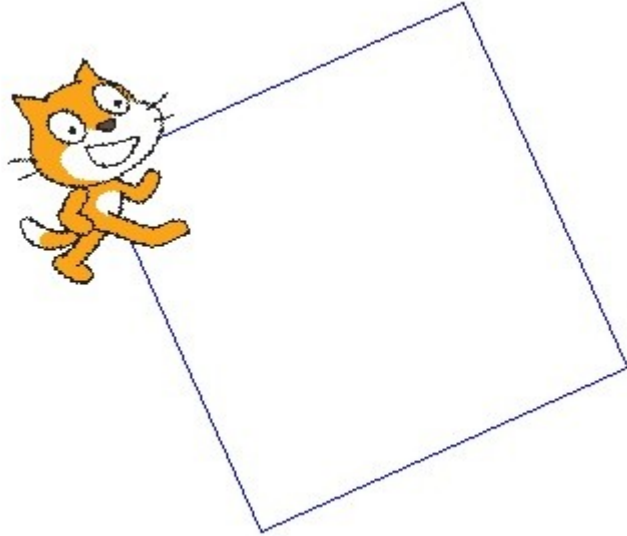
Scratch sistemin paralarının i ie geme durumunun nasıl olduęunu gsteren ok gzel bir rnektir.

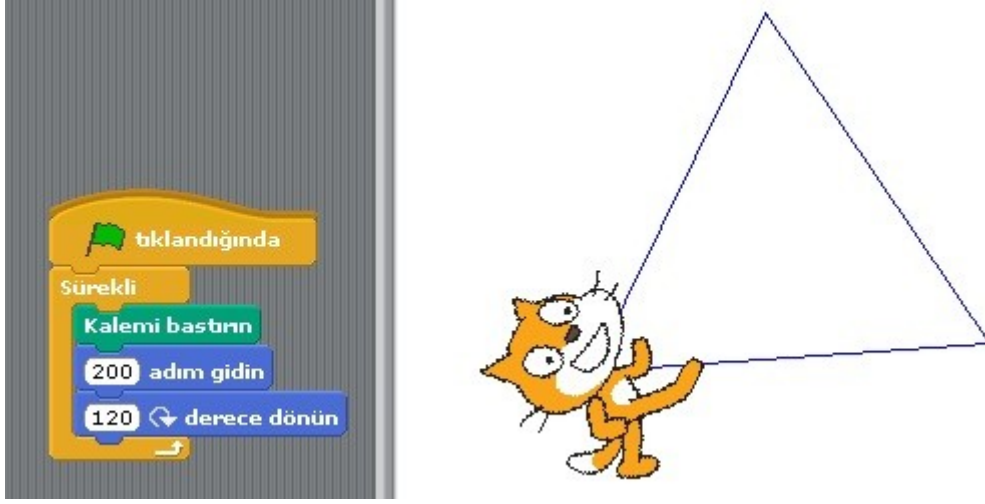


Scratch ile sistemin parçalarını görmek:



Scratch ile sistemin parçalarının birbirleri ile olan karmaşık ve iç içe geçmiş ilişkilerini, yapısını ve davranışını görmek:





Scratch ile sistemin hedefini belirleme ve kontrol edebilme:



### Donella Meadows'dan Sistemler İçinde Yaşama Rehberi

- Sistemin ritmini bulun.
- Zihin modellerinizi gün ışığına çıkarın.
- Bilgiyi onurlandırın, saygı duyun ve dağıtın.
- Dilinizi dikkatle ve sistem kavramları ile zenginleştirerek kullanın.
- Önemi olan şeylere dikkat edin, sadece sayıyla ölçülene değil.
- Geribildirim sistemleri için geribildirim politikaları üretin.
- Bütün sistemin iyiliği için hareket edin.
- Sistemin aklını dinleyin.

- Sistemdeki sorumluların yerini belirle.
- Alçak gönüllü ol, devamlı öğren.
- Karmaşayı kucakla.