

6.SINIF 2.DÖNEM DERS NOTLARI

1.HAFTA

1- Bu Nasıl Problem?

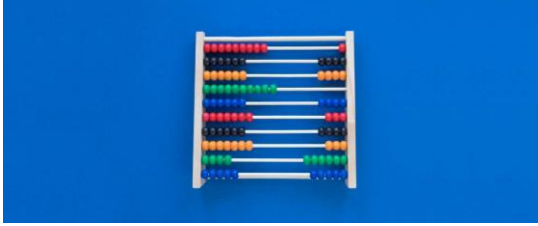
Bilişim Teknolojileri ve Yazılım Dersi'nin en önemli materyali sizin de tahmin edebileceğiniz gibi bilgisayardır. Bilgisayar için Türkiye'de önce "Kompüter" kelimesi kullanılmış. Daha sonra bu kavrama Türkçe bir karşılık bulma gereksinimi sonucu çalışmalara başlanmış ve "Bilgisayar" kavramı ortaya çıkmış.

Türkiye'de kullanılan ilk bilgisayar, 1960 yılında kullanılan "IBM-650 Veri İşleme Makinesi (Data Processing Machine)" adlı bilgisayardır. Bu bilgisayar, yol yapımında gereken hesaplamaları daha hızlı yapabilmek için Karayolları Genel Müdürlüğü tarafından alınmış ve 12 yıl kullanılmıştır.

Matematikten Bilgisayara Eğlenceli Bir Köprü

Şimdi sizinle matematik ve bilgisayar arasındaki ilişkiye bir göz atalım.

– İlk bilgisayar sizce ne olabilir? Oda büyüklüğünde bir makine mi? Bir masaüstü bilgisayar mı? Yoksa bir cep telefonu mu?



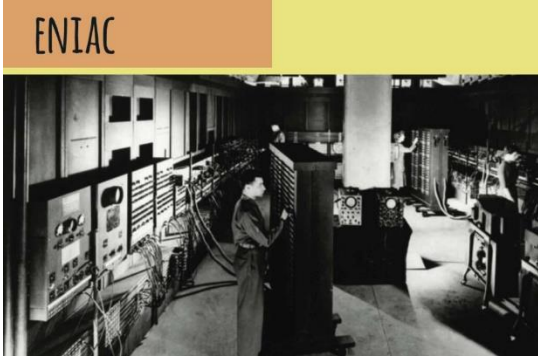
Abaküs

Aslında bilinen ilk bilgisayar **abaküstür**.

– Sizce abaküs ile bilgisayar arasında nasıl bir ilişki olabilir?

Bilgisayarlardaki programlar matematikteki formüllerden, yaklaşımlardan ve hesaplamalardan yararlanmaktadır. Bu sebeple basit matematiksel işlemler yapmaya yarayan abaküs ilk bilgisayar olarak kabul edilmektedir.

İlk **modern** bilgisayar ise elektrikle çalışan ve veri işleme kapasitesine sahip **ENIAC** adlı bilgisayardır. ENIAC bir ev büyüklüğündedir (167 metrekare) ve ağırlığı **30 tondur**.



ENIAC

Bilgisayar ve Matematikçiler

Matematikten bilgisayara uzanan süreçte pek çok bilim insanından bahsedebiliriz.

Bu bilim insanlarından biri **Ada Lovelace**'dir. Ada Lovelace kadınların çalışma hayatında bulunmadığı 1830'lu yıllarda bilgisayar programcısı olarak çalışmalar yapmış ve gelecek kuşaklara örnek olmuştur.

Blaise Pascal tarafından icat edilen **Pascalın**, 17. yüzyıl boyunca çalışan ilk ve tek mekanik hesap makinesiydi.

Bilgisayarın gelişimine katkıda bulunan iki kişiye yakından baktık ancak bu iki kişi dışında pek çok bilim insanı daha var:

- Mantık üzerine çalışmalar yapan **Gottfried Leibniz** ve **Charles Babbage**,
- Matematiksel dil yapısının öncüsü **George Boole**,
- Kuramsal makineler tasarlayan **Alan Turing**,
- Modern bilgisayarlara katkı sağlayan **John von Neumann**,
- Kuramsal programlama çalışmaları yapan **Dana Scott**.

2- Bilgisayarlar Nasıl Çalışır?

Bilgisayarların çalışma mantığını anlatan aşağıdaki videoyu izleyelim.

Aşağıdaki linkten videoyu izleyebilirsiniz.

<https://youtu.be/PVT5FsL8T10>



- İzlediğimiz videoda neler vardı?
- Videoda en çok dikkatinizi çeken nokta neresiydi?
- Videoyu izledikten sonra bilgisayar bilimleri ve matematik arasında nasıl bir ilişki olduğunu düşündünüz?

3- Haydi Veri Toplamaya

Geçen sene verinin ve bilginin ne olduğunu öğrenmiştik.

- Verinin ve bilginin ne olduğunu kim söylemek ister?

Veri: Kavram veya komutların, iletişim, yorum ve işlem için elverişli biçimli gösterimi.

Bilgi: Araştırma, gözlem ve benzeri öğrenme yolları ile elde edilen gerçektir.

Veri Tipleri

Bilgisayarda verilerin hesaplanmasında sadece sayısal bilgiler kullanılmaz. Bazı bilgiler karakter, bazıları metin, bazıları bir liste, bazıları tarih, bazıları da adres gibi bilgileri içermektedir. İşte bilgisayar verileri hesaplanırken be veri türlerine göre işlem yapmaktadır.

Aşağıdaki görselden veri türlerini inceleyelim.



46

Veri Tipleri

Ülke Veri Tablosu Etkinliği

Aşağıdaki etkinlikte yer alan açıklamaları dikkatlice okuyup

ÜLKE VERİ TABLOSU

1- Aşağıdaki metinden yararlanarak soruların yanıtlarını bularak yanlarına yazınız. Her sorunun yanıtı metin içerisinde olmayabilir. Burada araştırmacı kişiliğinizin devreye girmesi gerekiyor.

2- Üstte karışık şekilde yer alan veri tipleri ile tablodaki tanımları eşleştiriniz. Birinci soruda verdiğiniz cevapları uygun veri tipine göre tabloya yerleştiriniz.

TÜRKİYE
Asya ve Avrupa kıtalarında toprakları bulunan Türkiye Cumhuriyeti Devleti, 29 Ekim 1923 yılında kurulmuştur. 80 milyon 810 bin 525 nüfusu olan ülkemizde toplam 81 il bulunmaktadır. Ülkenizin yüz ölçümü 783.562 km²'dir. Birleşmiş Milletler (BM) üyesi olan Türkiye, aynı zamanda 1954'ten bu yana UNICEF'in (Birleşmiş Milletler Uluslararası Çocuklara Yardım Fonu) de üyesidir. Dünyada çocukların bayramının olduğu tek ülkedir.

Sorular:

- 1- Yüz ölçümü nedir?
- 2- Nüfusu nedir?
- 3- Nüfus yoğunluğu nedir?
- 4- En çok toprağı hangi kıtada yer alıyor?
- 5- Cumhuriyet hangi tarihte kuruldu?
- 6- Başkenti neresidir?
- 7- Birleşmiş Milletler Uluslararası Çocuklara Yardım Fonu üyesi mi?
- 8- Ülkenizin en batı ucundaki yerleşim biriminin ilk harfi nedir?

Veri Tipi Başlığı	Veri Tipi Tanımı	Yanıtlar
Karakter Dizisi Veri Tipi	Hesaplama işlemlerinde kullanılır. Tüm sayı çeşitlerini içerir. Pozitif ya da negatif tam sayılar kullanılabilir. Örneğin; açılar, uzaklık, nüfus, ücret, yarıçap...	
Karakter Veri Tipi	Tüm tek haneli sayıları, harfleri ve özel karakterleri kapsar. Tırnak içinde belirtilir. Bilgisayar sözcüğündeki "B" harfi bir karakteri ifade eder.	
Sayısal Veri Tipi	Birden fazla karakter bir araya geldiğinde bu veri tipi ortaya çıkar. Örneğin "Bilgisayar" sözcüğündeki karakterlerin tümü bu veri tipindedir.	
Özel Veri Tipi	Yalnızca iki kelime barındırır; doğru ve yanlış. Evet ya da hayır şeklindeki karar verme süreçlerinde kullanılır. Örneğin; arabası var mı, lise mezunu mu?	
Mantıksal Veri Tipi	Tarih, saat, adres, banka hesap numarası gibi verileri temsil eder.	

Ülke veri tablosu etkinliği

Veri Türleri Belirleme Etkinliđi

Ařađıdaki cümlelerde yer alan verilerin türlerini belirleyin.

1 Yorum

Ařađıdaki cümlelerde bahsedilen verilerin veri tipini seđiniz.

Benim adım Kemal. Ad bilgisi veri tipidir.

Mart ayında 12 yařıma gireceđim. Yař bilgisi veri tipidir.

Öđretmenim boyumu 160 cm olarak ölçtü. Boy bilgisi veri tipidir.

Zonguldak'a ilk defa 1 Haziran 2022'de gittim. Tarih bilgisi veri tipidir.

En sevdiđim meyveler çilek, muz ve kavundur. Meyveler bilgisi veri tipidir.

Soruya sınıfın yarısı Evet, yarısı Hayır cevabını vermiř. Cevap bilgisi veri tipidir.

Üç tane "P" harfi ile bařlayan gün vardır. Harf bilgisi veri tipidir.

Okulumuz Yenimahalle 186. sokaktadır. Adres bilgisi veri tipidir.

2.HAFTA

1- Sabit ve Deđişken Karmařası

– Hatırlıyor musunuz, “Sabit” ne demektir?

– Peki, “Deđişken”in “Sabit”ten farkını açıklayabilir misiniz?

Büyüyorum Şarkısı

- » İsmim sabittir benim
- » Yaşım deđişken
- » Ve deđiřtikçe yaşım
- » Deđiřirim bedenem
- » Deđiřiyorsa demek ki
- » O da bir deđişken
- » Deđiřtikçe deđişkenler
- » Büyüyorum her yıl ben...



Büyüyorum şarkısı

Sabit: İlk biçimiyle kalan, deđişmeyen ifade ya da nesnelere.

Deđişken: İlk biçimiyle kalmayıp yeni deđerler ya da biçimler alabilen ifade ya da nesnelere.

– Su 100 °C’de kaynar. Suyun kaynama derecesi sabittir.

– Bugün hava sıcaklığı kaç derece? Ölçülen hava sıcaklığı düne göre ya da gün içindeki farklı zaman dilimlerine göre farklılık gösterecektir. Bu sebeple hava sıcaklığı deđişkendir.

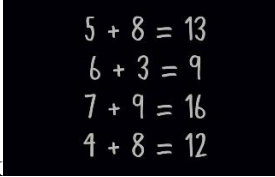
Yani özetle 100°C gibi sayısal değerler sabittir. Sıcaklık gibi farklı değerler alabilen ifadeler ise değişken olarak adlandırılır.



Sınıf



Oyun Parkı



Matematik İşlemi



Trafik



Yemek

StadyumGünlük hayattan değişkenler ve sabitler

Yukarıdaki görselleri inceleyelim ve nelerin sabit nelerin değişken olduklarına bakalım.

1. Sınıf görselinde öğrenciler, öğrencilerin çantaları, öğrencilerin montları **değişken**, Sıralar, tahta, panolar **sabit**.
2. Oyun parkındaki çocuklar **değişken**, oyuncaklar **sabit**.
3. Yemek görselinde yemek **değişken**, tabak çatal ve kaşık **sabit**.
4. Matematik işleminde sayılar **değişken**, + ve = gibi semboller **sabit**.
5. Trafikteki arabalar, içindeki yolcular **değişken**, yollar, trafik işaretleri **sabit**.
6. Stadyum görselinde, taraftarlar, futbolcular **değişken**, stadyum koltukları, kale, çizgiler **sabit**.

2- Kimler Burada Kalıyor?

Son zil çalıp da bizler okuldan gidince okulda neler kalıyor biliyor musunuz? Peki burada kalan sabitler biz yokken neler yapıyor olabilir acaba? Onlar da kendi aralarında sohbet edip gün boyunca olanlar hakkında konuşuyorsa... Neler konuşabilirler?

Bu soruları aşağıdaki etkinliği yaparak cevaplamaya çalışalım. Kalın yazı ile yazılan nesnelere sabit ve değişken olarak ayıralım.

KİMLER BURADA KALİYOR?

Son zil çalıp da bizler okuldan gidince okulda neler kalıyor biliyor musunuz? Peki burada kalan sabitler biz yokken neler yapıyor olabilir acaba? Onlar da kendi aralarında sohbet edip gün boyunca olanlar hakkında konuşuyorsa... Neler konuşabilirler?

ZİLİN ARDINDAN
Bir akşam okulum son zili çalıp da öğrenciler eve gitmek için heyecanla okuldan çıktıktan sonra bir sessizlik baktılar okulda. Ayardan geçen bir süre sonra okuldan bazı sesler duyulmaya başlar.
Diff, yine ne çok yoruldukları **(Sıralar)**
Bugün ne kadar heyecanlıydılar çocuklar değil mi? **(Kapı)**
Evet, Ayşe bugün ödevini öğretmene teslim etmek için koşarken bacasını benim köşeme çarptı. Camı yandı diye çok üzeldim. **(Sıralardan birisi)**
O da bir şey mi çocuklar sabah sınıfa giderken beni bana çarptılar. Bir an düşünürken kalemimi sandım ama Yusuf beni havada yakalayıp yerine yerleştirdi. Bu beni çok mutlu etti. **(Bilgisayar ekranı)**
Sarıya mı sarıya. Ben bugün biraz yiprandım. Birkaç çocuk beni alıp bir yere götürdüler. Beni götürürken atacaklarını sandım önce korktum. Dışta çocuklar beni aramaları için idareye götürdüler. Müdür yardımcısının odasına girince nasıl sevindim anlatamam. Sımda daha iyiyim. **(Çöp kutusu)**
O sırada okulun boş koridorlarında hızlı hızlı ayak sesleri duyulmaya başlayınca sınıftaki sesler kesilir. Emir hızla sınıfa girip masasına gider ve masasının üzerinden kitabını alıp koşarak sınıftan çıkar.
Emir'in ayak sesleri kesilince sınıfa yeniden sesler başlar.
Emir gelipormuş. Ben de kitabını unuttu diye endişeleniyordum. Neyse ki çok geç olmadan hatırladı.
(Emir'in kitabı)
Semra da geç olmadan hatırlasaydı keşke unuttuğu kalemliğini. **(Semra'nın kalemliği)**

Konuşmada yer almış olabilecek sabitleri ve değişkenleri aşağıya yazın.

SABİT	DEĞİŞKEN

Kimler Burada Kalıyor çalışma kağıdı

3- Bir Problem mi Var?

Aşağıdaki akış şemalarını inceleyelim. Hangi akış şeması yapıları üzerinde sabit ve hangileri üzerinde değişken kullanıldığını bulalım.



Toplama



Karşılaştırma



Yazdırma

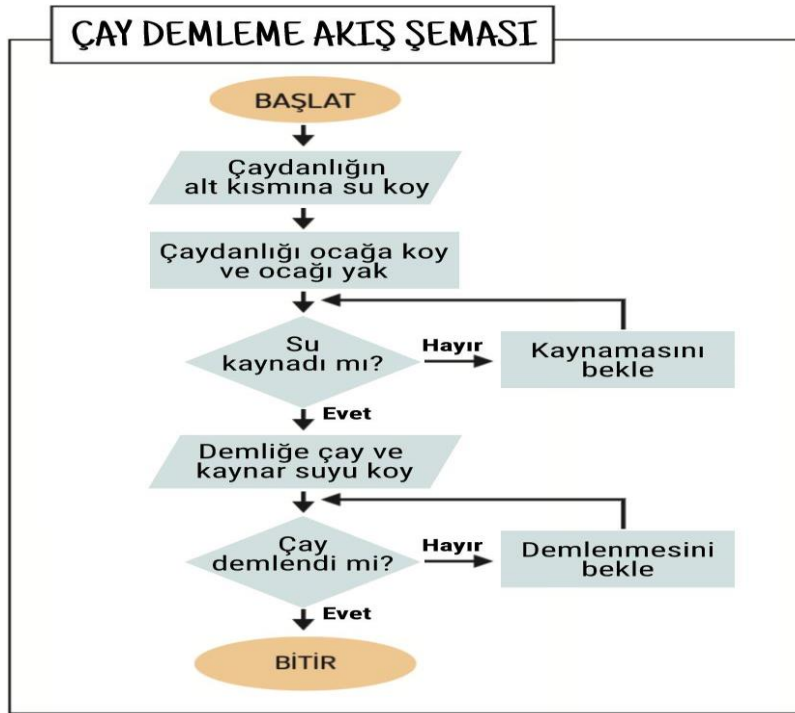


SıralamaAkış Şemaları

4- Ne Çok Problem Varmış!

İster bireysel ister grup halinde çalışabileceğiniz bir etkinlik yapacağız. Günlük hayattan bazı örneklerle hem sabitlerin hem değişkenlerin bulunduğu akış şemaları hazırlayalım. Akış şemanızı üstteki çalışmada gördüğümüz örneklerden faydalanarak yapabilirsiniz.

Örneğin basitçe çay demleme işlemi için akış şemasını oluşturalım.



Basit düzeyde çay demleme

akış şeması

Bu akış şemasında çaydanlık ve ocak **sabit**, koyulan çay, su ve bunların miktarı **değişkendir**. Sıra sizde! Günlük hayattan herhangi bir olayı basitçe akış diyagramına çevirin ve bu akış diyagramındaki sabitleri ve değişkenleri belirleyin.

3.HAFTA

1- Bu Nasıl Problem?

- Sizce problem nedir?
- Problemlerin çeşitleri olabilir mi?
- Karmaşık problem kavramını duydunuz mu?

Çevre kirliliği, evde ekmek kalmaması, yarına yapılacak ödevlerin olması, kuraklık tehlikesi, suyun iki saatliğine kesilmesi gibi olaylar gerçek yaşamda karşılaştığımız problemlere örnek olarak verilebilir.

Basit Problem: Basit adımlardan oluşan ve her koşulda aynı yönde ilerleyerek çözülebilen problemlerdir.

Örnek: Araba yıkamak, kek yapmak, evden okula gelmek vb.

Karmaşık Problem: Duruma özgü ve şartlara göre değişebilen çözüm adımlarından oluşan ve alt problemlere ayrılabilen problem türleridir. Çözüm için takım çalışması gerekebilir.

Örnek: Araba lastiği değiştirmek, pazar alışverişi yapmak, okulda başarılı olmak vb.

Algoritma

Bir problemi çözmek ya da belirli bir amaca ulaşmak, bir işi gerçekleştirmek için tasarladığımız yola **algoritma** denir. Algoritma yardımıyla bir işi adım adım gerçekleştirebiliriz. Aslında algoritmalar yaşamımızın bir parçasıdır.

Pek çok işimizi farkında olalım ya da olmayalım algoritma yardımıyla yaparız. Bu işlerin tümünde, algoritmalar gibi belirli bir sıra bulunur.

Aşağıdaki çalışma kâğıdındaki problemleri olası çözüm önerileri ile eşleştiriniz ve çözüm önerisine göre bir algoritma yazınız.

BASİT VE KARMAŞIK PROBLEMLER

Aşağıdaki tabloda bazı basit ve karmaşık problem cümleleri yer almaktadır:

- 1) Bu problem cümlelerini okuyunuz ve sizce basit (B) mi yoksa karmaşık (K) mi olduğunu tespit ederek (B) ya da (K) harflerinden birini yuvarlak içine alınız.
- 2) Problem sütununda yer alan her bir problem için uygun olduğuna düşündüğünüz olası çözüm önerilerini Olası Çözüm Önerileri sütunundan tespit ediniz. Problem cümlelerinin başında yer alan numarayı (1, 2, 3, ...) olası çözüm önerilerinin başında yer alan harfler (A, B, C, ...) ile eşleştiriniz.
- 3) Her bir problemin çözümü için ortadaki sütüne olası çözüm önerilerinden yararlanarak kendi algoritmanızı oluşturunuz.

PROBLEM	OLASI ÇÖZÜM ÖNERİLERİ
<p>1) Arabayla yolda gidiyordunuz ve aniden lastik patladı. B – K</p> <p>2) Tam akşam yemeğinden kalkıyordunuz ki davanız arayıp size geleceklerini söyledi ve misafirler için bir hazırlığınız yok. B – K</p> <p>3) Yarın Fen Bilimleri projesini teslim etmeniz gerekiyor ama henüz projenizi tamamlamadınız. B – K</p> <p>4) Okuldan mezun olacaksınız. Ama mezuniyet sürecinde neler yapman gerektiğini bilmiyorsunuz. B – K</p> <p>5) Büyüdüncü Bilgi Teknolojileri ve Yazılım dersi öğretmeni olmak istiyorsun ama nasıl bir süreç izlemen gerektiğini bilmiyorsun. B – K</p>	<p>A) Fen Bilimleri projesinin konusuna tekrar bakarım ve evdeki malzemelerle yarına kadar ne yapabileceğimi düşünerek, yapabileceğim en iyi proje tasarımı hazırlamaya başlarım.</p> <p>B) Bilgi alanına olan ilgimi, bilgiye dönüştürmek için araştırma ve çalışmalar yaparım. Bilgi Teknolojileri ve Yazılım dersi öğretmeni olabilmek için hangi okul ve bölümleri tamamlamam gerektiğini araştırırım. Araştırmamdan elde ettiğim veriler doğrultusunda bu derslerime daha çok çalışırım.</p> <p>C) Mezuniyetle ilgili işlemleri belirlerim ve sıralarım. Sırasıyla işlemleri tamamlarken arkadaşlarımla vedalaşır ve görüşmeyi sürdürmek için çeşitli yollar belirlerim.</p> <p>D) Aracı en güvenli şekilde yolun kenarına çekerek uyarı için gerekli işaretlemeleri yaparım. Bagajı açarak ihtiyacım olan malzemelerle birlikte stepneyi çıkarırım. Patlayan lastiği stepne ile değiştiririm.</p> <p>E) Akşam yemeğini hemen toplar ve misafir ağırlamaya dair evimde neler olduğunu gözden geçiririm. Evdeki malzemelerle ne yapabileceğimi belirler ve hazırlığa başlarım.</p>

ALGORİTMALAR

1	2	3	4	5
<p>1) Arabayı güvenli bir şekilde durdur.</p> <p>2) Dörtüleri yak.</p> <p>3) Aracıtan inerek reflektörlerini bagajdan çıkar.</p> <p>4) Aracın ortunda ve arkasında uygun mesafede reflektörleri yerleştir.</p> <p>5) Yedek lastik var mı ve sağlam mı kontrol et.</p> <p>6) Lastik değiştirmeyi bilmiyorsan servisi ara.</p> <p>7) Lastik değiştirmeyi biliyorsan başlıyorsun.</p> <p>8) Krikoyu tak ve aracı yükselt.</p> <p>9) Patlayan lastiği sök.</p> <p>10) Yedek lastiği tak.</p> <p>11) Patlayan lastiği araca yerleştir.</p> <p>12) Aracı indir.</p>				

Basit ve Karmaşık Problemler Çalışma Kağıdı

2- Çizim Makinesi

– *Fonksiyonun ne olduğunu hatırlıyor musunuz?*

Fonksiyon: Bağımsız değişkenler ile bağımlı değişken arasındaki ilişkinin matematiksel ifadesidir.



Hesap Makinesi

Hesap makinesi üzerinde yer alan tuşların her birisi birer fonksiyon ifade eder.

Bizler de bu dönem öğreneceğimiz Scratch veya mBlock gibi programlar aracılığıyla bir hesap makinesi yapabiliriz.

Biliyor musunuz hesap makinesinde de **sabitler** ve **değişkenler** yer alır. Sabit ve değişken kavramlarını hatırlıyor musunuz?

Hesap makinesi karmaşık bir problemdir. Sabit olarak tanımladığımız her bir işlem de karmaşık problemimizin alt problemleridir.

– *Mesela standart bir hesap makinesinde sabit olarak tanımlamamız gerekenler neler olabilir?*



Çizim Makinesi

Yanda görmüş olduğunuz şekilleri çizen bir makine olduğunu düşünün. Örneğin **üçgen** tuşuna basınca ekrana üçgen çizen, **dikdörtgen** tuşuna basınca ekrana dikdörtgen çizen bir makinemiz olsun. **Eşittir** şekil tuşlarına basınca çizimi onaylayan, **yuvarlak ok** ise ekranı temizleyen tuş.

Bu tuşlara ne görev verildiyse hepsi kendinen atanan belli bir görevi yerine getirmekte. İşte bunlara **fonksiyon** da diyebiliriz.

3- Gülse'nin Hikayesi -Şimdi aşağıdaki hikayeyi dikkatlice okuyalım ve hikayedeki alt problemleri belirlemeye çalışalım. Belirlediğimiz bu alt problemler için çözüm algoritması yazalım.

GÜLSE'NİN HİKAYESİ

Gölse bu sene 6. sınıfa devam ediyor. Kitap okumayı çok sever ve küçük yaşlarından beri eline geçen her kitabı okumaya çalışır. Birkaç seneden beri de yazmaya merak sađı. Kendi hikayelerini, yaşadıklarından kesitleri... Yazmaya başladığından beri katılmak istediđi okul gazetesine de bu sene dahil olabildi. Esra Öğretmen bu sene Gölse'yi okul gazetesine almıştı. Bu Gölse'yi çok sevindirdi. O kadar sevindi ki bir ara yapması gerekenleri bile görmedi. Heyecanı biraz durulunca okul gazetesinde yayın yapılabilmesi için gerekenleri listelemeye başladı.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____

Listeye bakınca bir an gözü korkar gibi oldu Gölse'nin. Sonra düşününce bunları halihazırda zaten yapıyordu, iyi bir planlamayla hepsini halledebileceğini düşündü. Şimdi tek iş yapılacaklar listesini detaylandırıp algoritmalarını hazırlamaktı...

Şimdi grubunuza denk gelen alt problem başlığını yazınız ve ardından çözüm algoritmasını oluşturun.

ALT PROBLEM BAŞLIĞI:

ÇÖZÜM ALGORİTMASI:

<p>ALT PROBLEM BAŞLIĞI:</p> <p>ÇÖZÜM ALGORİTMASI:</p>



4. HAFTA

1- Hayatım Problem, Çözümüm Algoritma

– Sizler yaşamınızda hangi işleri, hangi problemleri algoritma ile çözüyorsunuz?

– Problem çözümünde algoritma oluşturmak önemli midir? Neden?

Yaşamımızda bazı işleri farkında olmasak da alışkanlık gereği algoritma ile çözeriz. İşte bazı somut örnekler:

- Çay demlemek (Öğrencilere çay demlerken hangi adımları gerçekleştirdiklerini sorunuz ve çay demlemek amacıyla tahtaya öğrencilerle birlikte basit bir algoritma yazınız.),
- Yemek pişirmek,
- Okul bahçesinde uygun bir yer bulup oyun oynamak,
- Açılmayan bir bilgisayarın sorununu tespit edip açılır hâle getirmek,
- Proje çalışmasını yapmak.

Algoritma kullanarak;

- Problemleri daha hızlı ve sistematik olarak çözeriz.
- Problem çözme sürecini takip ederiz ve nerede hata yapıldığını görebiliriz.
- Tüm olasılıkları gözden geçirebiliriz.
- Hatalı işlem yapma olasılığımızı azaltırız.
- Olası hatalarımızı düzeltebiliriz.
- Çözüme ulaşmak için farklı yolları deneyebiliriz.
- Problemin çözümü için harcayacağımız süreyi kısaltırız.

2- Algoritmayla Sorunları Çözüyorum

Gündelik hayatta karşılaştığımız bazı durumları algoritma ile çözelim.

6.2.4.B1 - Algoritmayla Sorunları Çözüyorum
1. Hava çok sıcak ve iyice bunaldınız serinlemek için ne yaparsınız? Adım adım yazınız.
2. Taşınabilir belleği bilgisayara taktınız ancak çalışmadığını fark ettiniz. Çalışır hâle getirmek için ne yaparsınız? Adım adım yazınız.
3. Yarın önemli bir sınavınız var, çalışma masanıza oturdunuz. Birden masa lambanızın çalışmadığını fark ettiniz. Bu durumu çözmek için ne yaparsınız? Adım adım yazınız.
4. Bugün akşam yemeğine en sevdiğiniz arkadaşınız ve ailesi geliyor. Siz de arkadaşınıza ayran yapma sözü verdiğiniz. Ayrarı nasıl yapacağınızı adım adım yazınız.
5. Bilgisayarın başına oturdunuz ve klavyenin çalışmadığını fark ettiniz. Çalışması için neler yaparsınız? Adım adım yazınız.
6. Bir hafta sonra teslim etmeniz gereken bir projeniz var ve bu projenin en önemli kısmını oluşturacak kitap elinizde yok. Kitabı edinmek için neler yaparsınız?
7. Okula gitmek için evden çıktınız. Gökyüzüne baktınız ve kısa bir süre sonra yağmur yağacağını tahmin ettiniz. Okula ıslanmadan gidebilmek için neler yaparsınız? Adım adım yazınız.
8. Okulda yapılacak yıl sonu kermesi için kek yapacaksınız. Malzemeleri hazırlarken evde şeker kalmadığını fark ettiniz. Bu sorunu çözmek için izleyeceğiniz yolu adım adım yazınız.
9. Arkadaşınızdan aldığınız kitabı kaybettiniz. Kitabın son teslim tarihi yarın. Nasıl bir çözüm bulacağınızı adım adım yazınız.
10. Babanız sizi okuldan almaya gelecekti. Okul bittikten sonra uzun süre beklediniz ama gelmedi ve telaşlanmaya başladınız. Ne yapacağınızı adım adım yazınız.
11. Çok önemli bir ödevinizin son teslim tarihi bugüncü ancak evde unuttunuz. Nasıl çözüme ulaşacağınızı adım adım yazınız.

Algoritma ile sorunları çözüyorum çalışma kağıdı

3- Mayın Tarlası Oyunu

Mayın tarlası oyunu ile problem çözümü için algoritma geliştirmeyi daha da pekiştirmiş olacağız.

Şimdi birlikte bir oyun oynayacağız. Bu oyunun ismi Mayın Tarlası ve oyundaki amacımız mayın tarlasını başarılı bir biçimde geçecek algoritmayı adım adım yazmak. Bu süreçte grup arkadaşlarınızla birlikte hareket etmelisiniz. Her ayrıntıyı dikkatle inceleyip algoritmayı yazmalısınız.

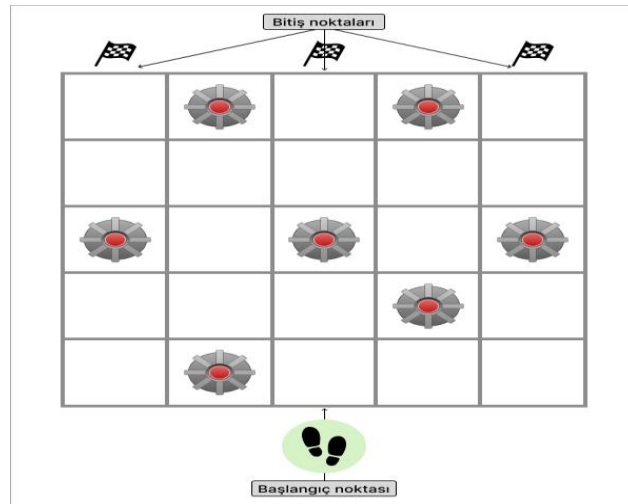
Kurallar:

1. Karşılıklı 2 grup oluşturulur. Gruplar en az 2 kişilik olmalıdır.
2. İlk grup mayın tarlasının yerleşimini dikkatlice inceleyerek, mayınlara basmadan karşıya geçilebilecek algoritmayı yazar.
3. Bu algoritma ikinci gruba verilir.
4. İlk gruptan bir kişi mayın tarlasının başlangıç noktasına getirilir ve gözleri bağlanır.
5. İlk gruptan başka bir kişi de algoritma okunurken doğru okunup okunmadığını kontrol etmek için ikinci grubun yanına geçer.
6. İkinci grup yazılan algoritmayı sırayla okur. Gruptaki her kişi bir sonraki maddeyi okuyacaktır.
7. Eğer okunan algoritma sonunda gözleri bağlı olan kişi mayına basmadan bitiş noktasına varırsa algoritmayı doğru yazdığı için oyunu kazanan ilk grup olacaktır. Eğer mayına basarsa ikinci grup “Bommm!” diye bağırarak, algoritma yanlış olduğu için algoritma yazma sırası ikinci gruba geçecek.
8. İkinci grup için yeni bir mayın tarlası düzeni hazırlanacak ve yeni düzene göre ikinci grup bir algoritma yazacak.

Aşağıdaki görsele göre mayın tarlasından başarılı bir çıkış için uygulanabilecek muhtemel algoritmalarından biri şu şekildedir:

- 1- Başla
- 2- İki adım ilerle
- 3- Olduğun yerde sola dön
- 4- Bir adım ilerle
- 5- Olduğun yerde sağa dön
- 6- İki adım ilerle
- 7- Olduğun yerde sola dön
- 8- Bir adım ilerle
- 9- Olduğun yerde sağa dön
- 10- İki adım ilerle
- 11- Bitir

Mayın tarlası oyunu



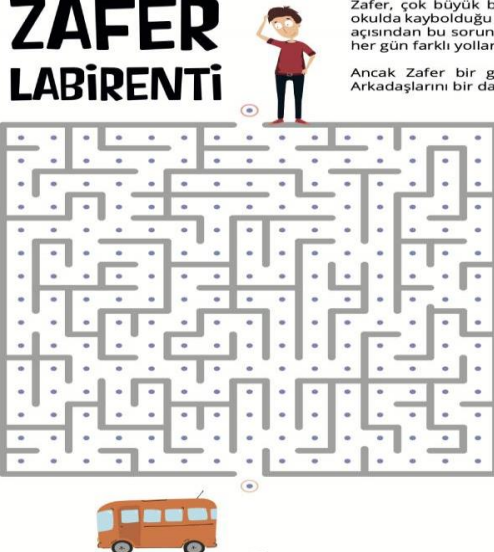
5.HAFTA

1- Farklı Yollarla Aynı Çözümü

Gündelik hayatta karşılaştığımız sorunlardan bazılarını tek bir yolla değil, farklı yol ve yöntemler kullanarak da çözebiliriz. Şimdi bununla ilgili bir örnek yapacağız.

Aşağıdaki çalışma kağıdında yazan açıklamaları oku ve etkinliği tamamla.

ZAFER LABİRENTİ



Zafer, çok büyük bir okulda öğrenciydi. Okul öyle büyüktü ki yeni başlayan öğrencilerin ilk günlerde okulda kaybolduğu bile oluyordu. Okul aynı zamanda oldukça karmaşık çıkış yollarına sahipti. Ancak Zafer açısından bu sorun her okul çıkışında bir eğlenceye dönüşüyordu. Zafer, okulun bu özelliğini bildiği için her gün farklı yollardan çıkış kapısına giderek eğlenmekteydi.

Ancak Zafer bir gün uzun bir yolu denemişti ve servisteki arkadaşlarını az da olsa bekletmişti. Arkadaşlarını bir daha bekletmek istemeyen Zafer, çıkış kapısına en kısa yoldan gitmeye karar verdi.

Zafer'i en kısa yoldan servise ulaştırmak için yardım etmeye ne dersin?

Yandaki labirentte Zafer'i okulun çıkış kapısında bulunan servise ulaştıran üç yol bulunuyor. Senin ilk görevin bu yolları hızlı bir biçimde bulmak. İkinci görevin ise bu yolları uzunluklarına göre sıralamak. En kısa yol Zafer'i servise en çabuk ulaştıracak olan yol olacaktır.

Bu yolları çizerken ya da uzunluklarına göre sıralarken birim noktalarını kullanabilirsin.

Haydi iş başına! Zafer'i servise ulaştıran en kısa yolun algoritmasını net ifadelerle aşağıya yaz ve servise en kısa zamanda ulaşmasına yardım et!

.....

.....

.....

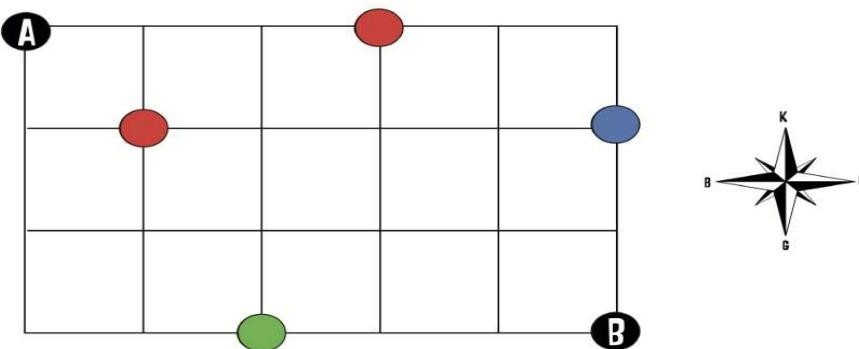
.....

.....

Zafer Labirenti çalışma kağıdı

2- En Kısa Yol

Aşağıda bir görsel ve altında yönergeler bulunmaktadır. Her bir yönergenin altında, yönergenin gerçekleşmesi için A noktasından B noktasına hangi adımlarla ilerleyeceğinizi belirten 3 farklı seçenek yazılmıştır. Sizden istenen bu seçenekleri denemek, en kısa ve en uzun yolu bulmaktır. Denedikten sonra, en uzun yolun karşısına 'U', en kısa yolun karşısına 'K' yazınız.



En kısa yol etkinliği

1. **A'dan B'ye bir yeşil bir maviye uğrayarak git.**
 - a) 2 birim doğuya, 1 birim güneye, 3 birim doğuya, 1 birim güneye, 3 birim batıya, 1 birim güneye, 3 birim doğuya git.
 - b) 3 birim güneye, 2 birim doğuya, 2 birim kuzeye, 3 birim doğuya, 2 birim güneye git.
 - c) 2 birim doğuya, 1 birim güneye, 3 birim doğuya, 3 birim batıya, 2 birim güneye, 2 birim doğuya git.
2. **A'dan B'ye iki kırmızıya uğrayarak git.**
 - a) 3 birim doğuya, 1 birim güneye, 2 birim batıya, 1 birim güneye, 4 birim doğuya, 1 birim güneye git.
 - b) 1 birim güneye, 1 birim doğuya, 1 birim kuzeye, 2 birim doğuya, 3 birim güneye, 2 birim doğuya git.
 - c) 3 birim doğuya, 3 birim batıya, 1 birim güneye, 1 birim doğuya, 1 birim güneye, 4 birim doğuya, 1 birim güneye git.
3. **B'den A'ya önce yeşile sonra maviye uğrayarak git.**
 - a) 3 birim güneye, 2 birim doğuya, 2 birim kuzeye, 3 birim doğuya, 2 birim güneye git.
 - b) 2 birim doğuya, 3 birim güneye, 2 birim doğuya, 3 birim kuzeye, 1 birim doğuya, 3 birim güneye git.
 - c) 2 birim doğuya, 3 birim güneye, 1 birim doğuya, 2 birim kuzeye, 2 birim doğuya, 2 birim batıya, 2 birim güneye, 2 birim doğuya git.

6.HAFTA

1- Haydi Sırala!

Günlük yaşamımızda yaptığımız işler gibi doğa olaylarının da kendi döngüleri ve algoritmaları vardır. Biraz düşündüğümüzde, bu döngülerin tamamlanabilmesi için de belli işlem adımlarının gerçekleşmesi gerektiğini fark edebiliriz.

Haydi Sırala çalışma kâğıdında doğadaki su döngüsüne ilişkin görseller yer alıyor. Bu tabloda görüldüğü gibi su döngüsü yağışla başlıyor. Diğer aşamaların görselleri ise tablonun altında karışık olarak verilmiş. Sizden istenen eksik olan resimleri tablodaki uygun yerlerle eşleştirmeniz. Bazı aşamalarda birden fazla görselin yer aldığına dikkat ediniz. Bu nedenle tablo üzerinde, aynı anda meydana gelen olaylar için birden fazla görselin yerleştirilebileceği boşluklar mevcut.

HAYDİ SIRALA!

Elinizdeki çalışma kâğıdında doğadaki su döngüsüne ilişkin resimler yer almaktadır. Tabloya göre su döngüsü yağışla başlar. Diğer aşamaların resimleri ise tablonun altında karışık olarak verilmiştir. Bu resimleri keserek tablodaki uygun yerlere yapıştırın.

Dikkat! Bazı aşamalarda birden fazla görsel vardır. Tablo üzerinde, aynı anda meydana gelen olaylar için birden fazla görselin yerleştirilebileceği boşluklar mevcuttur.

Haydi Sırala çalışma kâğıdı

Eğer çalışmayı bilgisayar üzerinde görsel düzenleme araçlarını kullanarak (photoshop, GIMP, photopea.com gibi) yapmak isterseniz katmanlar halinde hazırlanmış olan aşağıdaki PSD dosyasını indirebilirsiniz.

İNDİR: [haydi_sirala.psd](#)

Çalışmayı tamamladıktan sonra [buraya](#) tıklayıp çözümü görebilirsiniz.

2- Farkı Fark Edelim

Aşağıda 3 farklı senaryo ve bu senaryolara ait oluşturulmuş algoritmalar yer almaktadır. Ancak bu algoritmalar oluşturulurken bir hata yapılmıştır.

Bu hataları bulup doğru algoritmayı yazmanız gerekiyor. Kolay gelsin.

FARKI FARK EDELİM			
Senaryo 1 <p>Yağmur okulun basketbol takımında oynamaktadır. Haftada üç gün (pazartesi, çarşamba ve cuma) antrenmana gitmektedir. Antrenman günlerinde antrenman çantasını yanına alması gerekmektedir.</p> <p>Antrenmanı 15.30'da okul çıkışı yapmaktadırlar ve antrenman çantasını unutursa eve dönüp alması için zamanı kalmamaktadır. Ders saatlerinde cep telefonu kullanamadığı için antrenman çantasını antrenman günleri sabahtan yanına alması gerekmektedir.</p> <p>Yağmur süreci sağlıklı yürülebilmek için telefonunun alarm sistemini kurmuştur. Yağmur'un algoritmasında her şeye karşın aksayan bir şeyler olmuştur.</p> <p>Algoritmasını düzenleyebilmesi için Yağmur'a yardımcı olur musunuz?</p>		Senaryo 2 <p>Ercan okuldan çıkınca çalışmak ve ödevlerini yapmak için dairesinin iş yerine gitmektedir. Dairesinin iş yeri çarşıdaki Günay-Ham'ın 7. katındadır. Handa bir asansör yer almaktadır ama kat düğmelerinde sorun bulunmaktadır. Asansörde 8 katlı düğme sistemi olmasına karşın asansör yalnızca 5. kata kadar çıkmaktadır. Yani siz gitmediği katların düğmesine basarsanız bile asansör kabini sizi 5. kata götürmektedir.</p> <p>Ayrıca, çocukların yalnız başına binmelerini engelleyebilmek için de min. 30 kg. yük sınırı vardır.</p> <p>Asansörü çalıştıran algoritmayı kurarken yapılan hatayı nasıl giderebilirsiniz? Yardımcı olur musunuz?</p>	
Hatalı Algoritma: <ol style="list-style-type: none">1- Başla2- Antrenman çantasını al.3- Hayır ise 8. adıma git.4- Evet ise 6. adıma git.5- Antrenman çantasını yanına almak için saat 07.00'ye hatırlatma kur.6- Pazartesi, çarşamba veya cuma mı?7- Okula git.8- Bugün günlerden ne?9- Bitir.		Doğru Algoritma: <ol style="list-style-type: none">1- Başla2- "Tek başınıza asansöre binemezsiniz" yaz.3- Kat değeri 4'ten büyük ise 7. adıma git.4- Kullanıcı ağırlığını, ağırlık değişkenine ata.5- Kat değerine atanan kata çık ve 9. adıma git.6- Kat değişkenine, basılan kat numarasını ata.7- 5. kata çık ve 9. adıma git.8- Ağırlık 30'dan küçük ise 8. adıma git.9- Bitir	
Hatalı Algoritma: <ol style="list-style-type: none">1- Başla2- Kısa mesaj gönder3- Kod yanlış ise 7. adıma git.4- Sayaç değerine 0 ata.5- Sayaç değerini 1 arttır.6- Veli cep telefonu iste.7- "İndirme işlemi kullanımınıza açılmamaktadır" yaz.8- Yaş değerini oku.9- Yaş 12'den küçük ise 13. adıma git.10- Sayaç değeri 3'ten büyük ise 13. adıma git.11- Kısa mesaj ile gönderilen onay kodunu oku.12- Dosya indirme işlemi yap ve 14. adıma git.13- Yaş 15'den büyük ise 12. adıma git.14- Bitir.		Doğru Algoritma: <ol style="list-style-type: none">1- Başla2- Kısa mesaj gönder3- Kod yanlış ise 7. adıma git.4- Sayaç değerine 0 ata.5- Sayaç değerini 1 arttır.6- Veli cep telefonu iste.7- "İndirme işlemi kullanımınıza açılmamaktadır" yaz.8- Yaş değerini oku.9- Yaş 12'den küçük ise 13. adıma git.10- Sayaç değeri 3'ten büyük ise 13. adıma git.11- Kısa mesaj ile gönderilen onay kodunu oku.12- Dosya indirme işlemi yap ve 14. adıma git.13- Yaş 15'den büyük ise 12. adıma git.14- Bitir.	

Farkı Fark Edelim çalışma kağıdı

3- Yanlış Bunun Neresinde?

Bir bütünü oluşturan işlemlerin tamamı parçalardan ve eylem adımlarından oluşmaktadır. Tıpkı algoritmada olduğu gibi. Bir sonuca bir hedefe varmak için bir karar vermek için sürekli ardışık adımları izleriz. İzlemekte olduğumuz bu adımların tamamı o anki hedefimiz, vermek istediğimiz kararımız ya da ulaşmak istediğimiz sonuca yönelik özellikler içerir.

Mesela ilk olarak okula geldiğimizi düşünelim, güne okula gelmek için gerekli adımları atarak geliyoruz. Yine benzer adımlarla da eve dönüyoruz.

Şimdi sizinle bir hikâye paylaşacağım. Etkinlik için hazırlanmış olan "Böyle Biçim Ne Bu Hikâye!" akışında anlamsal ve kavramsal yanlışlıklar vardır. Yalnız bu hikâye biraz karışmış. Sizlerden sözcüklerin yerlerini değiştirerek metni anlamlı bir hâle getirmenizi istiyorum.

Aşağıda gördüğünüz hikâyede yolunda gitmeyen şeyler var! Anlamsal ve kavramsal hataların bulunduğu bu hikâyeyi anlaşılır hale getirmeye ne dersiniz? Hikâyeyi düzeltirken kimi zaman sözcüklerin yerlerini değiştirebilir, kimi zaman da cümle anlamını bozan sözcüklerin yerine başka sözcükler kullanabilirsiniz.

BÖYLE BİÇİM NE BU HİKÂYE!



Osman, başucundaki gözlerini çalınca saat açar. yatağından ayağa hemen kalkar. dişlerini yıkayıp yüzünü fırçaladıktan sonra kahvaltısını giyinir ve yapar. Osman giyen ayakkabılarını formasını düzeltir, evden takar ve çantasını çıkar. yoluna gitmek için okula koyulur.

arkadaşları da okula koyulurlar yola gelmek için. yolun köşelerinde yolunda karşılaşırlar okul ayrı ayrı. amca köşedeki iyi çocuklara marketteki dersler diler. kapıdaki günaydın güvenlik girerlerken der çocuklara okula. öğretmenlerini doğru giderlerken görürler merdivenlerde sınıfa. beden Osman eğitimi der öğretmenine günaydın. merdivenlerde grup doğru grup devam ederler sınıflarına öğrenciler. zil oturur çalınca herkes sıralarına. öğretmen alır sınıfa ve gelir yoklamayı.

gelmeyen der öğrenciler için sınıf yok arkadaşları, gelenlerse burada. öğrencilerin yavaş sesi ve nefesi yavaş sınıfa ısıtır. konuşmaya öğretmen başlar, bilişim dersimiz teknolojileri ve bugün yazılım. bilgisayarlarınızı şimdi açın. bilgisayarlarımızdan kedi programımızı simgesi olan çalıştırın. Ayşe'nin açılmayınca yeniden dener açmayı bilgisayarı. açamayınca öğretmeni çağırır. klavye tıklamaları fare tuşları dakikalar dakikaları kovalar karışırken. zil için çalar sınıf aniden ve tenefüs boşaltılır.

okulun kantininde sıra olmuştur. en susamış çok simiti severler. kantinin sıraya önündeki girilir, su bir alınca çıkılır şişe sıradan. elinde hızla şişesi merdivenlere su yürünür zil yetişebilmek için derse çalmadan. bir önceki materyalleri altına dersin sıranın kaldırılır ve yeni başlayacak heyecanla hazırlık derse yapılır

Böyle Biçim Ne Bu Hikaye çalışma kağıdı

Eğer hikâyeyi anlamlı hale getirdiyseniz kontrol etmek için [buraya](#) tıklayabilirsiniz.

4- Ayıkla Pirincin Taşını

Aşağıda her biri 10×10 boyutunda hazırlanmış senaryolara yönelik hatalı algoritmalar bölümlerin altlarında yazılmıştır. Bu algoritmaları deneyerek hataları düzeltmenizi istiyorum. Bu düzeltmeyi yaparken boş bırakılan bölüme doğru algoritmayı yazmalısınız.

Etkinliğin açıklamasını dikkatlice okumayı unutmayın.

AYIKLA PİRİNCİN TAŞINI

Aşağıda her biri 10×10 boyutunda hazırlanmış senaryolara yönelik hatalı algoritmalar bölümlerin altlarında yazılmıştır. Bu algoritmaları deneyerek hataları düzeltmenizi istiyorum. Bu düzeltmeyi yaparken boş bırakılan bölüme doğru algoritmayı yazmalısınız.

0: Çukur X: Ergel ■: Çukur Noktası

	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>
	<p>0: Çukur X: Ergel ■: Çukur Noktası</p>

Ayıkla Pirincin Taşını çalışma kağıdı

7.HAFTA

1- Benzer Sorun Benzer Çözüm

Günlük yaşantımızda çeşitli problemlerle karşılaşırız. Bu problemlerin bazılarını yakından incelediğimizde aslında çözüm yollarının çok da zor olmadığını görürüz. Dahası bu basit problemler için bulduğumuz çözümleri benzer özellikler taşıyan ve daha karmaşık gözükten problemler için de kullanabiliriz. Şimdi sizinle bu şekilde bir çalışma yapacağız.

Kantin Sırası Senaryosu:

Okulumuzda bulunan dilek ve şikâyet kutusuna atılan mektuplar incelenmiş ve kantinin teneffüslerde çok kalabalık olması ile ilgili şikayetlerin arttığı gözlenmiştir.

Kantindeki yoğunluğun sebepleri incelendiğinde şu sonuçlara varılmıştır;

Kantin teneffüslerde tüm öğrenciler tarafından aynı anda kullanılmaktadır.

Kantinde üç görevli ve bir ödeme noktası vardır.

En büyük zaman kaybı para ödemesi ve para üstü alınması sırasında yaşanmaktadır.

Sizlerden kantindeki yoğunluğun azaltılması ile ilgili çözüm önerileri istenmektedir. Lütfen bulduğunuz çözümleri yazınız.

Sizlerden kantindeki yoğunluğun azaltılması ile ilgili çözüm önerileri istenmektedir. Lütfen bulduğunuz çözümleri yazınız.

Çözüm önerileriniz muhtemelen şunlara benziyordur:

- Kantindeki görevli ve kasa sayısı artırılmalı.
- Tüm sınıfların kantini aynı anda kullanmaması için her bir sınıfın kantini kullanacağı teneffüsler belirlenmeli (5. ve 6.sınıflar 1. teneffüs, 7. ve 8. sınıflar ise 2. teneffüs gibi).
- Belirli katlarda, kantinde çok satılan ürünlerin yer aldığı bozuk para ile çalışan yiyecek otomatları konulmalı.
- Ödeme işleminin uzun sürmesi nedeniyle oluşan yoğunluğu azaltmak için farklı bir yol bulunmalı (deftere kaydetme, önceden ödenen paradan düşme vb.)

Şimdi bu problemi ve çözümünü sıklıkla yaşadığımız daha büyük bir problemi çözmeye kullanalım: “Trafik Sıkışıklığı”

Kantindeki yoğunluk ile trafik yoğunluğu arasındaki benzerlikleri bularak benzer bir çözüm önerisi yazabilir misiniz?

Örnek bir benzer çözüm şu şekilde olabilir:

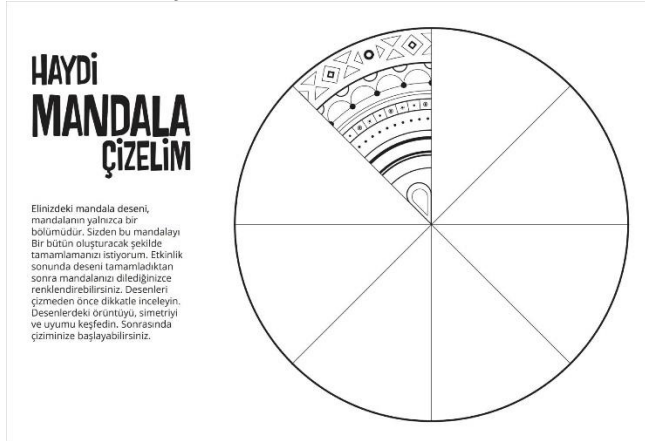
- Kantindeki görevli ve kasa sayısı artırılmalı çözümü, trafik problemine toplu taşıma araçlarının sayısı ve güzergahları artırılmalı.
- Tüm sınıfların kantini aynı anda kullanmaması için her bir sınıfın kantini kullanacağı teneffüsler belirlenmeli çözümü, işten çıkan tüm çalışanların aynı anda ulaşım araçlarını kullanmaması için iş çıkış saatleri düzenlenebilir.
- Ödeme işleminin uzun sürmesi nedeniyle oluşan yoğunluğu azaltmak için farklı bir yol bulunmalı çözümü, toplu taşıma araçlarında elektronik ödeme sistemlerine geçilmeli veya her kapiya bu sistemden koyulmalı.

2- Haydi Mandala Çizelim

– *Mandala kavramını daha önce duyan var mı?*

Mandala kökeni çok eski zamana dayanan bir çizim ve boyama tekniğidir. Mandala doğu dillerinde ‘enerjiyi tutan kap’ anlamına gelmektedir. Bizim kültürümüzde ise mandala örneklerine **dantel desenlerinde** ve **çini boyama sanatında** rastlayabiliriz.

Mandala çizimi **daire** şeklinde yapılır. Bir dairenin merkezinden başlayarak kullanmak istediğiniz çeşitli şekilleri simetrik ve uyumlu olarak, tekrarlı bir biçimde çizerek mandala desenleri oluşturabilirsiniz.



Mandala tamamlama çalışması

8.HAFTA

1- Blok Tabanlı Programlama



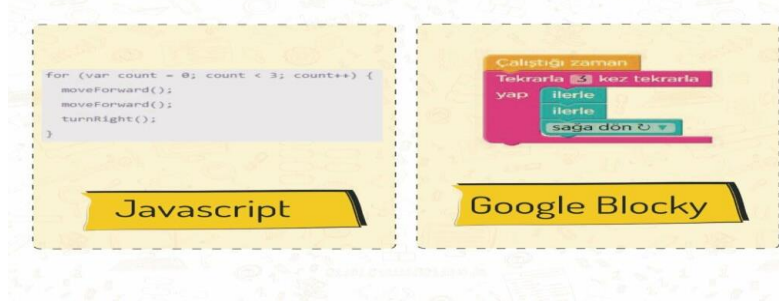
– Aranızda blok tabanlı programlama kavramını duyan var mı?



Bilgisayar programlama dilleri bir yabancı dil gibi öğrenilmesi zaman alan ve ilk bakışta karmaşık dillerdir. Bu dillerin daha kolay öğrenilmesini ve yaratıcı şekilde kullanılmasını desteklemek amacıyla çeşitli blok tabanlı programlama araçları geliştirilmiştir.



Blok tabanlı programlama araçları belirli bir program dilinde kullanılan komutu, işlevini anlatan bloklara çevirmektedir. Bu şekilde iletişim kolaylaşır.



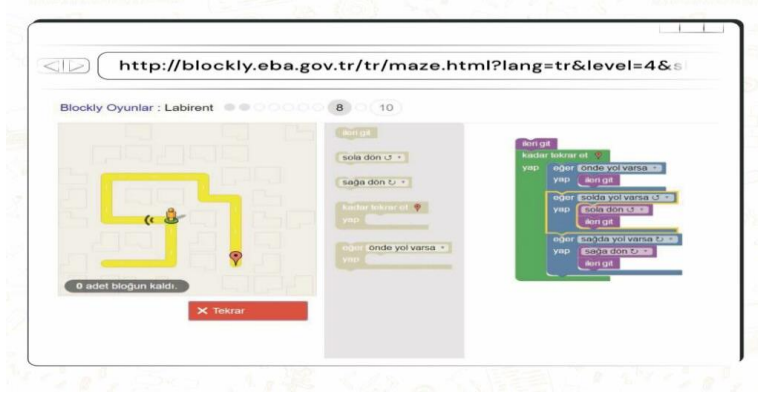
İşte size bir örnek: Soldaki JavaScript dilinde yazılmış olan programlama komutu, sağda Google Blockly kodları kullanarak yazılmış. Anlaması çok daha kolay öyle değil mi?



Kod blokları legolara benzer, iç içe geçer arka arkaya sıralanır ve programınızın verilen komutları yapmasını sağlarlar. Bir karakterin yürümesi, konuşması ya da harflerin canlanması vb.



Bunlar arasında en yaygın olanları Google Blockly, mBlock ve Scratch programlarıdır.



Geçen seneki derslerimizde EBA üzerinden de (<http://blockly.eba.gov.tr/>) erişebildiğimiz Google Blockly ile tanışmıştık.

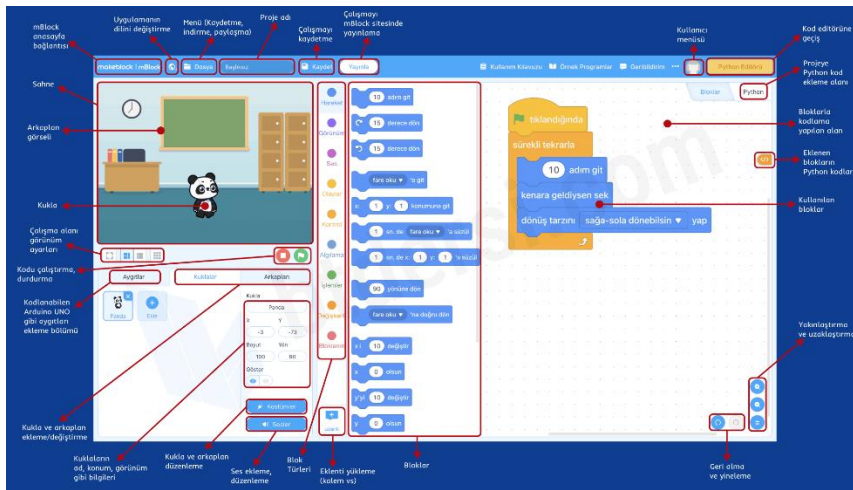
Bu yıl ise **mBlock** uygulamasını kullanacağız. mBlock Windows ve macOS işletim sistemlerine yüklenebilmekte, ayrıca Pardus gibi işletim sistemleri dahil olmak üzere tüm işletim sistemleri için tarayıcı üzerinden de kullanılabilir.

2- mBlock Programı

mBlock, MIT (Massachusetts Institute of Technology) tarafından geliştirilen, blok tabanlı yapısı sayesinde herhangi bir programlama dilini bilmeksizin, kod bloklarını yapboz birleştirir gibi bir araya getirerek, basit oyunlar geliştirme olanağı sağlayan bir yazılımdır. mBlock ile görerek ve sürükleyip bırakarak, robotlardan özel devre kartlarına kadar birçok şeyi programlayabiliriz.

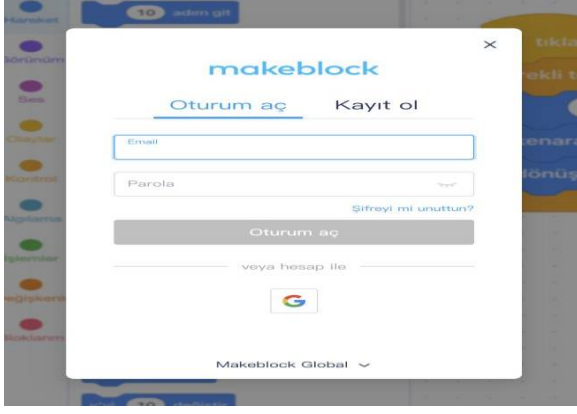
mBlock uygulamasına <https://mblock.makeblock.com> adresinden ulaşabilirsiniz. Eğer Windows veya macOS işletim sistemine sahip bir bilgisayar kullanıyorsanız <https://mblock.makeblock.com/en-us/download> adresinden bilgisayarınıza indirip kullanabilirsiniz.

Aşağıda mBlock uygulamasının arayüzü yer almaktadır. Arayüzü dikkatlice inceleyelim.



mBlock arayüzü

mBlock uygulamasında kod bloklarını uygun şekilde kullanarak bir kuklanın belirli görevleri yapmasını sağlayabiliyorsunuz. Ayrıca Arduino UNO gibi kodlanabilen elektronik kartları projeye ekleyerek hazırladığınız kodlarla çeşitli elektronik devre elemanlarını da kontrol edebilirsiniz.



mBlock giriş/kayıt ekranı

mBlock uygulamasını kullanabilmek için üye olmaya gerek yok ancak çalışmaların mBlock hesabında kaydedilebilmesi için üye olmak gerekiyor.

Üye olduğunuzda internet bağlantısı olan herhangi bir yerden projelerinize ulaşabilir, mBlock topluluklarında çalışmalarınızı yayımlayabilirsiniz.

3- mBlock'u Deniyorum

mBlock uygulamasını <https://mblock.makeblock.com> adresinden açarak serbest şekilde denemeler yapın. Yeni kuklalar, arkaplanlar eklemeye çalışın. Kuklaya ses eklemeye çalışın. Denemeler yapmaktan çekinmeyin. Bir problemle karşılaşırsanız yeni bir tarayıcı sekmesinde mBlock uygulamasını tekrar açabilirsiniz.

4- Merhaba Dünya

Şimdi hep birlikte ilk mBlock uygulamamızı yapalım.

İlk olarak kuklamızı hareket ettirerek başlayalım.

“**Hareket**” blokları içerisinde “**10 adım git**” bloğunu bulup sürükleyerek programlama alanına getirelim. Bloğun üzerine çift tıklayın. Kuklamız ileriye gidecek. Bunu birkaç kez deneyebilirsiniz.

“**Görünüm**” blokları içinden “**2 saniye Merhaba! de**” bloğunu ilk bloğumuzun altına ekleyelim. “Merhaba!” yazısına tıklayarak onu “**Merhaba Dünya**” şeklinde değiştirelim. Şimdi bloklarımıza çift tıkladığımızda kuklamız önce 10 adım gidecek sonra Merhaba Dünya diyecektir.



Merhaba Dünya kod blokları

Kuklamızın sahnenin hemen altındaki **yeşil bayrağa** tıkladığımızda harekete başlaması için üst kısımda yer alan “**Olaylar**” grubundan “**tıklandığında**” bloğunu alalım ve bloklarımızın üzerine yerleştirelim.

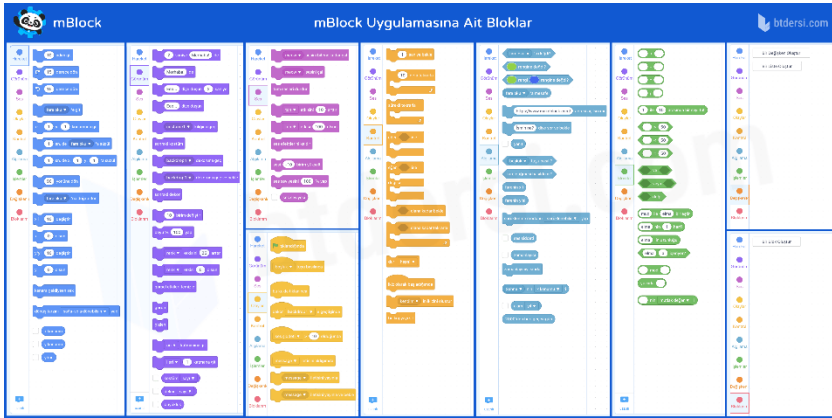
Eklediğimiz blokların sayı kısımlarını değiştirerek daha fazla adım atmasını, daha uzun süre Merhaba Dünya demesini sağlayabiliriz.

Son olarak kuklamızın yaptığı hareketleri **sürekli tekrarlamasını** istiyorsak “**Kontrol**” grubunda yer alan “**sürekli tekrarla**” bloğunu, tekrarlanmasını istediğimiz blokları içine alacak şekilde ekleyebiliriz.

9.HAFTA

1- Blokları Keşfedelim

Scratch veya mBlock gibi blok tabanlı programların bir çoğu benzer arayüze ve benzer bloklara sahiptir. Aşağıda mBlock uygulamasına ait tüm blokların yer aldığı bir görsel mevcuttur. Bu görseli dikkatlice inceleyin. Hangi bloğun hangi grup altında olduğunu kolayca bulabilirsiniz. Blokları daha net görmek için görsele tıklayıp büyütebilirsiniz.

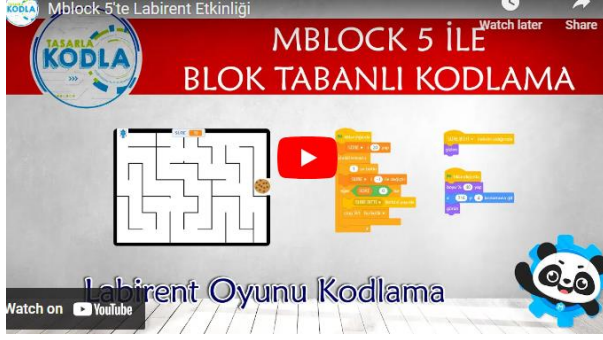


Bu görselde yer alan bloklar **kuklalara** eklenen bloklardır. Arkaplan ve Aygıtlar için benzer bloklar olduğu gibi bunlara özel bloklar da vardır.

2- Biraz Hareketlenelim

Geçen hafta ve bu hafta mBlock’u keşfetmeye çalıştık. Bu noktadan sonra birlikte mBlock projeleri yapacağız ve bunları kişiselleştireceğiz. mBlock ile programlama yapmak bir tiyatro oyunu yönetmek gibidir. Tiyatroda olduğu gibi mBlock’ta da karakterler vardır (kuklalar), kostümler (kılıklar), arkaplan ve bir sahne vardır. mBlock programı “olaylar” adı altında bir dizi komutu projede neyin ne zaman olması gerektiğini planlamak için kullanır. Örneğin yeşil bayrağa tıkladığımızda proje başlar, ya da “bu kuklaya tıkladığında” bloğu ise tıklanan kuklayı harekete geçirir.

Şimdi bir böceğin labirentte gezerek kurabiyeye ulaşmasını sağlayacak etkinliği yapacağız. Nasıl yapıldığını aşağıdaki videodan izleyip deneyebilirsiniz



<https://youtu.be/1XQRZtUBTww>

videoyu yukarıdaki linkten izleyebilirsiniz.

10.HAFTA

1- Animasyon Nedir?

Animasyon resimlerin ya da hareketsiz nesnelere arka arkaya hızlı bir şekilde gösterilmesiyle karakterlerin hareket ediyormuş gibi görünmesini sağlayan bir gösterim şeklidir. İlk animasyonlar birkaç kâğıda istenen resimlerin çizilmesi ve kâğıtların hızlıca geçirilmesi veya bir çemberin içine yerleştirilip döndürülmesi ile yapılıyordu. Çizgi filmler de birer animasyondur.

Benzer bir çizimi sizin de oluşturmanız mümkün. Mesela defterinizin bir köşesine bir daire çizin. Hemen sonraki sayfaya da aynı daireyi, önceki dairenin bulunduğu hizadan 3-5 cm uzağa çizin. Örnek bir çizim aşağıda verilmiştir. Bu iki çizimi ardarda tekrarlayarak 20-30 sayfa ilerledikten sonra defterin sayfalarını hızlıca çeviriniz. Top iki nokta arasında zıplıyormuş gibi görünecektir. Eğer isterseniz benzer bir çalışmayı çöp adam çizerek de yapabilirsiniz. Bir sayfaya bacakları kapalı, sonraki sayfaya da bacakları açık bir çöp adam çizebilirsiniz. Bu çizimi 20-30 sayfa devam ettirdiğinizde ve sayfaları çevirdiğinizde çöp adam yürüyor gibi görünecektir.

Şimdi mBlock programında bir animasyon projesi yapacağız. Bir kuşun iki farklı görüntüsünü hızlı hızlı değiştirdiğimizde uçuyormuş gibi görünmesini sağlayacağız.

2- Papağanı Uçuralım

Bu çalışmamızda bir papağanın kanat çırparak uçuşmasını sağlayacağız. Adım adım yapalım.

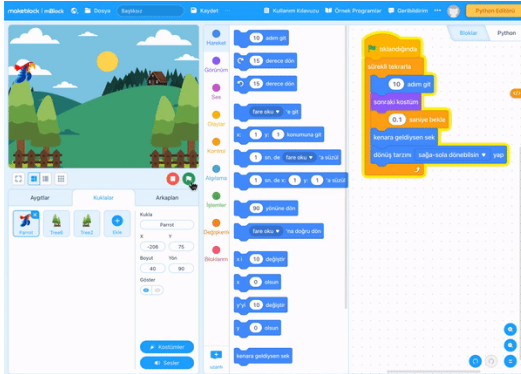
1. Öncelikle pandamızı bir önceki derste öğrendiğimiz gibi silelim.
2. Kukla kütüphanesinden papağanı seçelim. Arama kutusuna **parrot** yazabiliriz.
3. Kostümler butonuna tıkladığımızda kuklamızın parrot-a ve parrot-b olarak 2 kılığının bulunduğunu göreceksiniz. Papağanımızın uçuşması için bu iki şekli arka arkaya kullandığımız bir kod yazacağız.
4. Papağanımızı uçurmadan önce sahnemizi hazırlayalım. Bunun için arkaplan bölümüne basıp bir arkaplan seçelim.

5. Sahneye dilersek birkaç tane ağaç ekleyebiliriz. Bunun için kostüm ekle butonuna basalım. Kütüphaneden tree yazıp bir ağaç kuklası seçelim ve sahnemize ekleyip uygun bir yere konumlandıralım.
6. Ağaç ve papağanın boyutlarını sahnede uygun ölçülere getirelim.
7. Sonra aşağıdaki blokları sahnemize ekleyelim.



Blokları açıklayalım:

- o Bayrağa tıklandığında animasyon başlar.
- o Animasyonun sürekli oynaması için yani papağanın sürekli uçması için ilgili komutlar sürekli tekrarlar bloğu içine koyulur.
- o 10 adım git bloğu, papağanı bulunduğu noktadan 10 birim ileriye götürür.
- o Sonraki kostüm bloğu, papağanın diğer kostümdeki görüntüsüne geçer.
- o 0.1 saniye bekle bloğu, kostümler arası geçişin yani kanat çırpma hızını ayarlar. Bu değer büyüdükçe kostümler arası geçiş yavaşlar, böylece daha yavaş kanat çırpır.
- o Kenara geldiysen sek bloğu, papağanın sahnenin kenarına ulaştığında geri dönmesini sağlar.
- o Dönüş tarzını sağa-sola dönebilsin yap bloğu ise kuklanın kenara değip dönmesi esnasında dikey olarak ters dönmesini engellemiş olur.



– Aynı animasyonda bir de kelebek uçurabilir misiniz?

3- Kendi Animasyonumuzu Oluşturalım

Şimdi sıra sizde. Öncelikle fikrinizi belirleyin. Bu fikre uygun akış şemasını çizin. Sonra bir kukla seçin. Bu kuklayı uygun bir sahnede akış şemanıza göre hareket ettirin. Yapabildiğiniz kadar iyisini yapmaya çalışın. Kolay gelsin.

11.HAFTA

1- Şakacı Penguen

Aşağıdaki şakacı penguen videosunu izleyelim.



<https://youtu.be/xn11cJhRTYo>

Yukarıdaki linkten izleyebilirsiniz

Yukarıda izlediğiniz animasyonun benzerini mBlock kullanarak yapacağız. Animasyondaki kuklalar Scratch uygulamasında mevcut. mBlock uygulamasında ise benzer ya da farklı kuklalar seçerek animasyonumuzu yapacağız.

Animasyonumuzu iki farklı yöntemle yapabiliriz. Birinci yöntem kuklaları **zamana göre** konuşturmak. İkinci yöntem **İleti Yayınlama** (diğer adıyla habersal) ile konuşturmak.

Birinci Yöntem – Zamana Göre

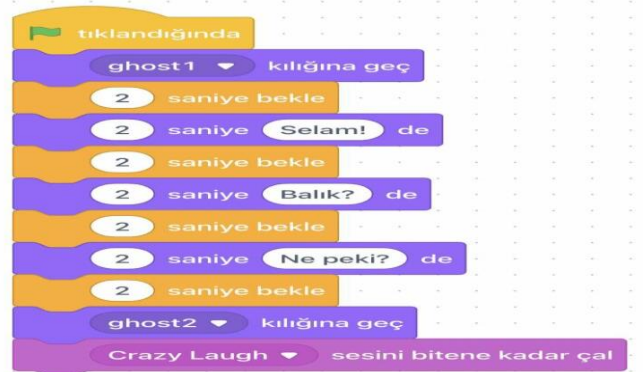
Bu yöntem için aşağıdaki adımları uygulayalım.

- mBlock uygulamasını açalım ve ilk olarak panda kuklasını silelim.
- Penguen kuklasını [buraya](#) tıklayıp bilgisayarınıza indirin ve kukla kütüphanesindeki yükle butonunu kullanarak penguen kuklasını projenize ekleyin. Penguenin boyutunu 35 yapın.
- İkinci kuklayı da bu sefer kukla kütüphanesinden seçelim. Arama kutusuna **ghost1** yazalım ve çıkan kuklayı projemize ekleyelim.
- Pengueni ve hayalet kuklalarını uygun boyutlara getirip konumlandıralım. Hayalet kuklasının yönüne dikkat edin, yüzünü pengüene dönmesi için kostümlerden gerekli düzenlemeyi yapalım.
- Sıra geldi kod bloklarını eklemeye. Kuklalara aşağıdaki kod bloklarını ekleyelim. Ses bloklarından **Crazy Laugh** sesinin eklendiğine dikkat edelim. Ayrıca **ghost2** kılığının da eklendiğine dikkat edelim.

Penguen Kuklasına Eklenecek Bloklar



Hayalet Kuklasına Eklenecek Bloklar



Eğer animasyonumuz başarılı bir şekilde çalıştıysa şimdi ikinci yönteme geçebiliriz.

İkinci Yöntem – İleti Yayınlama (Habersal)

İkinci yöntem ile animasyonumuzu yapmak için aşağıdaki adımları uygulayalım

- İlk yöntemde anlatılanlar gibi sahnemiz ve kuklalarımız hazırladıysak, kuklalara aşağıdaki kod bloklarını ekleyelim

Penguen Kuklasına Eklenecek Bloklar



Hayalet Kuklasına Eklenecek Bloklar



İleti Yayınlama (habersal) yöntemi öğelerin (kuklaların ve sahnelerin) birbirleri ile iletişim kurmalarını sağlar. Bir kukla için yazılan ileti yayınlama komutu diğer kuklaların eylemlerini de etkiler.

Arkaplanlar (Sahne Dekorları) Arasında Geçiş

Blok tabanlı kodlama programlarında kod blokları sadece kuklalara eklenmez. Arkaplanlara da blok eklenebilir. Yukarıdaki örnekte arkaplana tıkladığımızda başka arkaplanları gösterebilmek için aşağıdaki adımları uygulayalım

- Arkaplan sekmesine tıklayıp Kostümler butonuna basalım.
- Altaki Dekor Ekle butonuna basıp dilediğimiz arkaplanları ekleyelim
- Kostümler ekranından çıkıp blokların listelendiği ekrana dönelim.
- Arkaplan seçiliyken aşağıdaki blokları ekleyelim.



Eğer tüm adımları doğru bir şekilde uyguladıysak animasyonumuz aşağıdaki gibi olacaktır



<https://youtu.be/ahjf6EYGP28>

Yukarıdaki linkten izleyebilirsiniz.

Video sonunda kahkaha sesi çıktığında hayalet kuklasının görüntüsündeki değişikliği farkettiler mi?

2- Öğrendiklerini Uygula

Öğrendiklerimizi başka bir örnekle uygulama zamanı.

Tamamen kendi fikrinize göre oluşturacağımız projede;

- En az 2 farklı dekor geçişi olacak
- İçinde 2 karakterin birbiriyle konuştuğu bir bölüm olacak.

Projeye başlamadan önce şu sorulara cevap vermelisiniz:

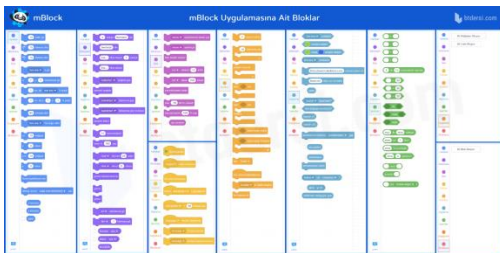
- Karakterler birbirlerine ne diyecekler?
- Hangi dekorları kullanacaksınız?
- Konuşmalarını sağlamak için habersal bloklarını mı yoksa zamanlamayı mı kullanacaksınız?

Kolay gelsin...

Penguen ve Hayalet animasyonuna ait mBlock dosyalarını aşağıdaki bağlantıdan indirip kendi bloklarınızla karşılaştırabilirsiniz.

12.HAFTA

1- İşlemler ve Algılama Blokları



Yukarıdaki görselde Algılama ve İşlemler blok gruplarını bulup inceleyelim.

Algılama Bloğu: Bu blok bize nesnelere birbiriyle iletişime geçmesini sağlayan işlemleri gerçekleştirmemizi sağlayan komutları sunar. Mesela yolda yürürken önümüze bir engel çıkınca ne yapacağımız bellidir.

İşte burada yer alan komutlar da kuklalarımız için benzer durumlarda ne yapacağımızı belirlememizi sağlar. “Ağaca yaklaştığında” ya da “Ağaca dokununca” gibi komutlar yardımıyla projemizi daha gerçekçi hâle getirmemiz mümkündür.

İşlemler Bloğu: Bu blokta yer alan komutlar bizim matematiksel işaretleri projelerimizde kullanmamıza yardımcı olur.

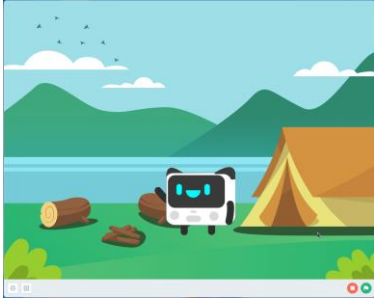
“+, -, *, /, =, >, <” gibi işlemleri bu blok yardımıyla projelerimizde kullanabiliriz.

Bir başka görevi de algılama bloğundaki komutları kullanırken bize yardımcı olmasıdır. Bir oyun tasarlarken karşınıza çıkan engel bir renk içeriyorsa bu blokta yer alan komutlar yardımıyla gerekli tanımlamamızı yapabiliriz.

2- Karakterle Sohbet

Geçtiğimiz hafta iki karakteri birbirleriyle konuştuğu bir animasyon yapmıştık. Peki acaba oluşturacağımız karakter bize soru sorsa ve bizimle konuşsa sizce nasıl olurdu?

Projemiz aşağıdaki animasyona benzemeli.



Haydi deneyelim.

Karakterle Sohbet Projesi Uygulama Adımları

Bugün bizimle konuşan bir karakter animasyonu yapacağız. Projeye başlamadan önce şu sorulara cevap vermelisiniz:

- Karakterinizin adı ne?
- Nerede yaşıyor?
- Nasıl bir karakter? Mutlu mu? Ciddi mi? Komik mi?

Soruları cevapladıysanız animasyonumuzu yapmaya başlayalım.

1. [mBlock](#) uygulamasını açtıktan sonra ilk işimiz panda karakterini silmek.
2. Karakter kütüphanesinden Fantezi kategorisindeki dilediğiniz bir karakteri seçin. Biz burada robot karakterlerinden **C-codey-rocky** karakterini kullanacağız. Karakteri seçerken ikinci kostüme sahip olmasına özen gösterin. Eğer seçtiğiniz karakterin ikinci kostümü (mesela gülümseyen hali) yoksa o zaman kostüm düzenleme ekranında ikinci kostümü ekleyerek gülümseyen halini oluşturmalısınız. **Not:** Karakterinizin merkez noktasını, kostüm düzenleme ekranının merkezine yerleştirmeyi unutmayın. Bunu diğer kostümler için de yapmalısınız.
3. Arkaplan sekmesinden bir dekor seçin.

4. Haydi şimdi karakterimizin bizimle konuşmasını sağlayalım! Karaktere tıklayın ve aşağıdaki blokları kullanın. Açık mavi renkli Algılama bloklarından olan “İsmin ne?” diye sor ve bekle bloğu ekrana soru yazdırmamızı sağlar.



Bakalım işe yarıyor mu? Kuklaya tıkla sana “İsmin ne?” diye soracak, cevabı yaz ve bekle.

5. Karakterimiz her seferinde aynı cevabı veriyor “Ne güzel bir isim”. İsmimizi öğrenip ona göre cevap vermesini de sağlayabiliriz. Bunu 3 adımda yapacağız.

Adım 1: Algılama grubundan (açık mavi bloklardan) “yanıt” tuşunu ve İşlemler grubundan (yeşil bloklardan) “-muz- ile -elma- birleştir” bloklarını ekrana sürükleyelim.

Adım 2: “yanıt” tuşunu “elma” yazan yere sürükleyelim ve “muz” yazan yeri “Merhaba” diye güncelleyelim.



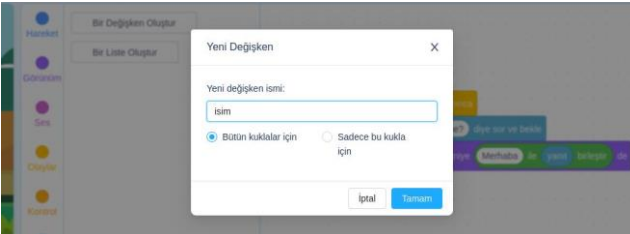
Adım 3: Bu düzenlediğimiz yeşil bloğu da daha önce kullandığımız mor renkli “2 saniye Ne kadar güzel bir isim!” bloğundaki “Ne kadar güzel bir isim!” yazan alanın içine yerleştirelim. Aynen aşağıdaki gibi.



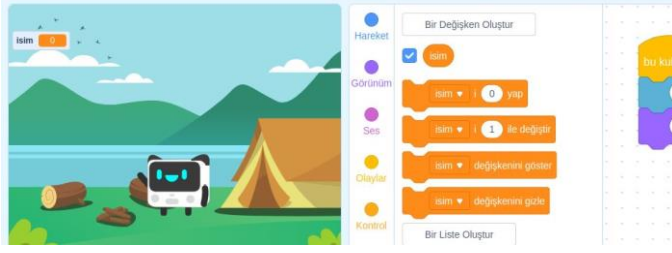
Şimdi tekrar deneyelim. Nasıl oldu? Herhangi bir problem var mı? (Merhaba yazarken bir boşluk eklemen gerebilir.)

6. **İsmin ne?** sorusuna gelen cevabı, uygulamamızın tamamında **ihtiyaç duydukça** kullanabilmek için onu bir **değişken** olarak tanımlamamız gerekiyor.

Şimdi **isim** adında bir değişken oluşturalım. Bunun için turuncu renkli **Değişkenler** blok grubuna tıklayın ve **Bir Değişken Oluştur** butonuna basın. Açılan penceredeki kutuya **isim** yazın ve “Bütün Kuklalar İçin” seçeneğinin seçili olduğuna emin olun.



7. Değişkeni ekledikten sonra Değişkenler blok listesi aşağıdaki gibi görünecektir.

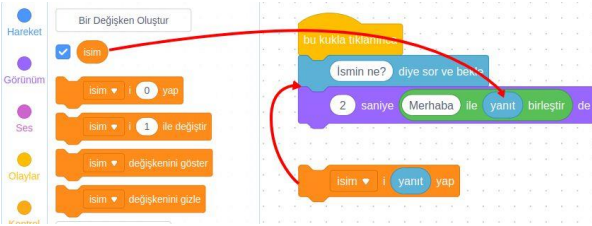


Şimdi burada yeni oluşan turuncu bloklardan **-isim-i-0- yap** bloğunu ve **Algılama** bloklarından **yanıt** bloğunu sağ tarafa blok alanına ekleyelim.

Aşağıdaki görselde gösterildiği gibi **yanıt** bloğunu **-0-** yazan yere yerleştirelim.



Aşağıdaki görselde gösterildiği gibi **isim** değişkenini daha önce eklediğimiz **-Merhaba- ile -yanıt- birleştir** bloğu içindeki **yanıt** bloğu üstüne koyalım. Turuncu renkli **-isim-i-yanıt- yap** bloğunu da **İsmin ne?** bloğunun hemen altına koyalım.



8. Son yaptığınız değişiklikler sonrasında bloklarınızın görüntüsü aşağıdaki gibi olacaktır. (Eğer ekranda **yanıt** bloğu boşa çıktıysa onu silelim.)



9. **isim** değişkeninin başındaki **tık** işareti değişkenin sürekli sahnede görünmesini sağlar. Sahneden gizlemek için tik işaretini kaldırabilirsiniz.
10. Oluşturduğumuz **isim** değişkenini başka kodlar için de kullanabiliriz demiştik. Şimdi karakterimizin başka bir soru daha sormasını sağlayalım. “Okula gidiyor musun?”



11. Bu soruya cevabımız **Evet** ise karakterimizin “**Harika!** demesini, **Hayır** ise “**Hmm!**” demesini istiyoruz. Bunun için bir “**eğer-değilse**” bloğu ve bir “**eşitlik**” bloğu kullanmamız gerekecek. Önce aşağıdaki blokları bulup ekleyin ve sonra gösterildiği gibi gerekli düzenlemeleri yapın.



İsim ile soru arasında bir boşluk bırakmak için okula... ifadesinden hemen önce bir boşluk bırakmayı unutmayın.

12. Blokları uygun yerlere eklediğimizde son görüntü aşağıdaki gibi olmalıdır.



13. Şimdi de karakterimizin bize soracağı bir soruya göre arkaplanın (dekorun) değişmesini sağlayalım. Bunun için projemize yeni bir dekor eklemeliyiz. Örneğin **gökyüzü** arkaplanı eklemek için arama kutusuna **Sky1** yazalım ve çıkan arkaplanı ekleyelim.
14. Karakterimizin sorduğu soruya vereceğimiz cevaba göre arkaplanı değiştirmek için aşağıdaki blokları kullanabiliriz. Cevap **Evet** ise hem arkaplan değişsin hem de karakterimizin kılığı değişsin. Eğer **Hayır** cevabı verirse de “**Peki o zaman, görüşmek üzere...**” desin ve animasyon dursun.

```

Ben şimdi gökyüzüne çıkacağım. Sen de gelmek ister misin? diye sor ve bekle
eğer yanıt = Evet ise
  Sky1 dekoruna geç
  C-codey-rocky-c kılığına geç
değilse
  2 saniye Peki o zaman, görüşmek üzere... de
dur hepsi

```

15. Yukarıdaki soruya **Evet** cevabı verdiğimizde karakterimizin sevinçten zıplamasını da sağlayabiliriz. Bunun için aşağıdaki kod bloklarını ekleyelim. Bu

```

5 defa tekrarla
  y'yi 10 değiştir
  0.1 saniye bekle
  y'yi -10 değiştir
  0.1 saniye bekle

```

16. Yukarıdaki 13. ve 14. adımlarda arkaplan ve karakterimizin kostümünü değiştirmiştik. Animasyonumuzu en baştan başlattığımızda ilk arkaplan ve ilk kostüme dönmesi için aşağıdaki blokları da en başa (Bu kukla tıkladığında bloğundan hemen sonra) eklememiz gerekiyor.

```

Camp2 dekoruna geç
C-codey-rocky-a kılığına geç

```

17. Projemizdeki blokların son görüntüsü aşağıdaki gibi olması gerekiyor. Dilerseniz projenizi **bu kukla tıklanınca** yerine **bayrak tıkladığında** ile başlatabilirsiniz.

```

bu kukla tıklanınca
  Camp2 dekoruna geç
  C-codey-rocky-a kılığına geç
  ismin ne? diye sor ve bekle
  isim i yanıt yap
  2 saniye Merhaba ile isim birleştir de
  isim ile okula gidiyor musun? birleştir diye sor ve bekle
  eğer yanıt = Evet ise
    2 saniye Harika! de
  değilse
    2 saniye Hmm! de
  Ben şimdi gökyüzüne çıkacağım. Sen de gelmek ister misin? diye sor ve bekle
  eğer yanıt = Evet ise
    Sky1 dekoruna geç
    C-codey-rocky-c kılığına geç
    5 defa tekrarla
      y'yi 10 değiştir
      0.1 saniye bekle
      y'yi -10 değiştir
      0.1 saniye bekle
  değilse
    2 saniye Peki o zaman, görüşmek üzere... de
dur hepsi

```

23 Nisan Ulusal Egemenlik ve Çocuk Bayramı için bir proje hazırlayacağız. (Dilerseniz bu projeyi **uygun görseller** kullanarak 19 Mayıs Gençlik ve Spor Bayramı için de yapabilirsiniz.)

Projemiz aşağıdaki animasyona benzemeli.

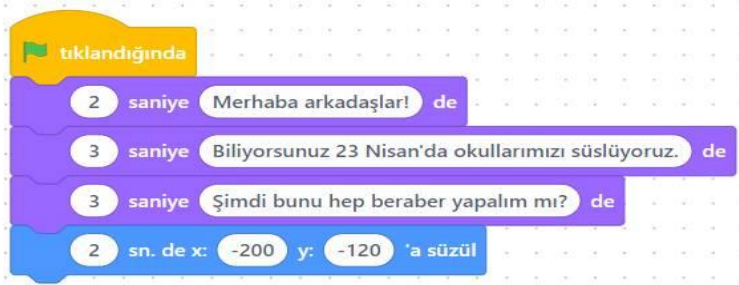


Bunun için aşağıdaki adımları sırasıyla uygulayalım.

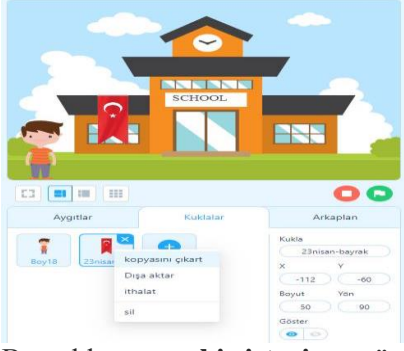
1. [mBlock](#) uygulamasını açtıktan sonra ilk işimiz panda karakterini silmek.
2. Arkaplan (Dekor) kütüphanesinden **Okul** kategorisindeki **School1** isimli okul görselini seçelim.
3. Karakter kütüphanesinden **İnsanlar** kategorisinden dilediğiniz bir karakteri seçebilirsiniz. Biz **Boy18** isimli karakterle devam edeceğiz.
4. Projemiz 23 Nisan Ulusal Egemenlik ve Çocuk Bayramı için okulu süslemek. Karakterimiz bize yol gösterecek. Karakterimizin proje başında kısa bir açıklama yapmasını istiyoruz. Bunun için onu konuşturmalıyız.

Konuşma kısmından sonra projemize biraz hareket katmak için de karakterimizin yerini değiştirelim. Bunun için **hareket** semesinde yer alan **süzül** kodunu kullanabiliriz. Yeni yerini belirlemek için **x'e -200 y'ye ise -120** yazıyoruz. Belirttiğimiz noktaya **2 saniyede** gitmesini istediğimiz için ilk kutucuğa **2** yazıyoruz.

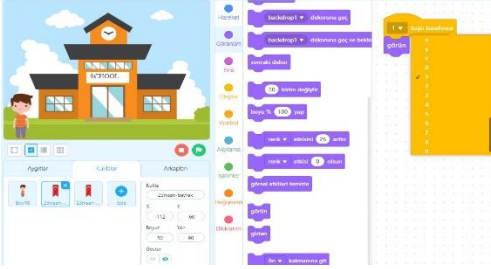
Aşağıdaki blokları ekranımıza ekleyelim. Projemiz bayrak tıklandığında başlayacak ve hemen ardından karakterimiz bizimle konuşacak (metinler sırayla gelecek, ayrıca konuşma sürelerine de dikkat edelim). Konuşma bitince de sol tarafa doğru süzülecek.



5. Artık okulun süslenmesi kısmına geçebiliriz. Ancak küçük bir problemimiz var. mBlock kütüphanesinde yer alan kuklalardan okul süslemesi için kullanabileceğimiz kukla sayısı az. Bu nedenle bazı kuklaları kendimiz bulmalıyız. Bu sebeple bayrak için [buraya](#) tıklayarak ve afiş için [buraya](#) tıklayarak görselleri bilgisayarımıza indirelim. Eğer başka kuklalar kullanmak isterseniz İnternet'ten de indirebilirsiniz.
6. Önce bilgisayarımıza indirdiğimiz bayrağı kukla ekleme kısmında 'Dosya yükle' kısmını tıklayarak ekliyoruz. Eklediğimiz bayrak resmini okula göre küçültüp sol tarafa yerleştiriyoruz. İki tane bayrak kullanacağımız için 'kopyasını çıkart' seçeneğini kullanarak 2. bayrağımızı elde edebiliriz. İkinci bayrağımız da sağ tarafa yerleştirilelim. Aşağıdaki görseli inceleyelim.



- Bayraklarımızın **biz isteyince görünmesini sağlayacağız**, bunun için önce bayrak kuklası seçiliyken Boyut-Yön bilgilerinin hemen altındaki **Göster** kısmından **kapalı göz** seçeneğine tıklıyoruz. Bu ayarı üstteki görselde sağ altta bulabilirsiniz. Bu ayarı diğer bayrak için de yapmalıyız.
- Şimdi gelelim tuşlara basınca gizlediğimiz bayrakları göstermeye. İlk bayrak seçiliyken; **Olaylar** bölümünden **“boşluk tuşu basılınca”** bloğunu ve **Görünüm** bölümünden **“görün”** bloğunu ekliyoruz. Eklediğimiz ilk bloktaki **boşluk** yazan yere tıklayalım ve açılan listeden **“1”** değerini seçelim.



Kukla listesinden ikinci bayrağı seçerek aynı işlemleri ikinci bayrak için de yapalım. **Boşluk tuşu** yazan yere tıklayalım ve bu sefer açılan listeden **“2”** değerini seçelim.

- Karakterimizin (Boy18) işlemler sırasında bizi beklemesi için küçük bir ekleme yapmamız gerekiyor. Bunun için **“Kontrol”** sekmesinden **“olana kadar bekle”** bloğunu koyalım. İçine de **“Algılama”** sekmesinden **“boşluk tuşu basılı?”** komutunu ekleyelim.

Biz bayraklarımızın 1 ve 2 tuşuna basılınca görünmesini istediğimiz için **olana kadar bekle** bloğundaki boşluğa **Algılama** grubundan **boşluk tuşu basılı** bloğunu ekleyip **boşluk** yazan yeri **2** şeklinde değiştiriyoruz. Böylece karakter biz 2 tuşuna basmadan bir sonraki adıma geçmeyecek.



- Okulumuzu süslemek için bir de 23 Nisan afişi asalım. Bayraklarda yaptığımız gibi bilgisayarımızdan 23 Nisan afişini bularak projemize ekliyoruz. Afişi, seçilen bina dekorunda uygun bir boşluğa yerleştirebiliriz.

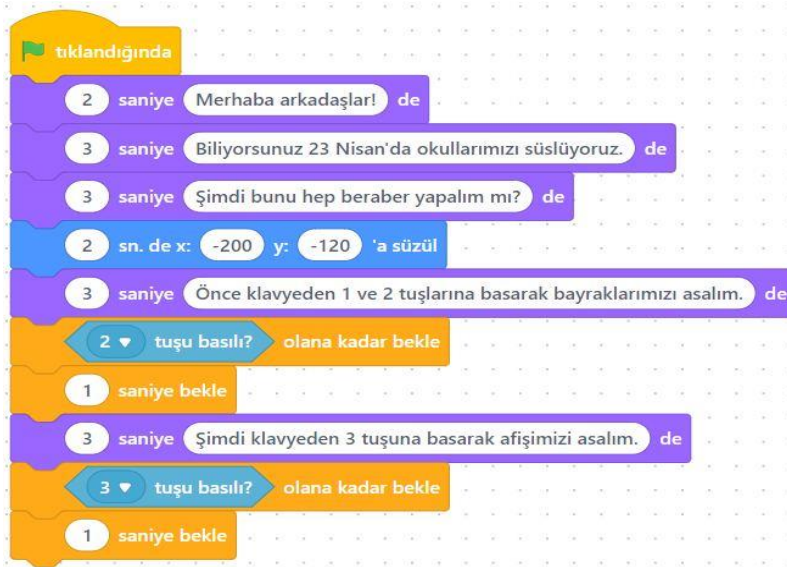
Yine bayraklarda olduğu gibi **bir tuşa basınca görünmesini istiyoruz**. Afişimizin biz isteyince görünmesini sağlamak için önce bayraklarda yaptığımız gibi afiş kuklası seçiliyken Boyut-Yön bilgilerinin hemen altındaki **Göster** kısmından **kapalı göz** seçeneğine tıklıyoruz.

Daha sonra **Olaylar** bölümünden “**boşluk tuşu basınca**” bloğunu ve **Görünüm** bölümünden “**görün**” bloğunu ekliyoruz. Eklediğimiz ilk bloktaki **boşluk** yazan yere tıklayalım ve açılan listeden “**3**” değerini seçelim. Bayraklarda 1 ve 2 tuşlarını kullanmıştık.



11. Yukarıda **9. adımda** 2 tuşuna basıncaya kadar karakterimizi (Boy18) bekletmiştik. Aynı şekilde **3 tuşuna basıncaya kadar** yine bekletmemiz gerekiyor. İlgili blokları 9. adımdaki gibi ekleyelim.

Adımlar arası geçişlerin çok hızlı olmaması için kukla seçiliyken eklediğimiz kod bloklarına “**1 saniye bekle**” komutunu ekliyoruz. Aşağıdaki blokları inceleyelim.



12. Okulumuza 4 tane de balon asarak süslememizi tamamlayalım. Bunun için mBlock kütüphanesinde bulunan balonları kullanabiliriz. Balonun 3 farklı rengi var. Aynı balondan 4 tane ekleyelim ve kostümler kısmından istediğimiz rengi seçelim. Balonları da okula göre küçülttükten sonra uygun yerlere yerleştiriyoruz. Ardından diğer kuklalarda olduğu gibi **istediğimiz zaman görünmeleri için gizliyoruz**.



Balonlarımızı da 4 tuşuna basınca görünecek şekilde ayarlıyoruz. Bu işlemi tüm balonlar için yapmayı unutmayalım.

13. Karakterimizi (Boy18) seçiyoruz. 9. adımdaki mor bloklar gibi “Şimdi 4’e basarak afişimizi asalım” demesi için bir konuşma bloğu ekliyoruz.

Bayrak süslemesinde karakterimizin bizi beklemesi için bir kod yazmıştık. Bu kodu tekrar kullanacağız. Bu nedenle koda sağ tıklayıp kopyasını çıkartabiliriz. Karakterimizin balonlarımızı asana kadar beklemesi için “4 tuşu basılı olana kadar bekle” komutunu ekliyoruz. Son olarak da karakterimiz bize teşekkür etsin. İşte karakterimize eklediğimiz blokların son hali:



14. Proje sonunda okulun ilk baştaki hâline dönmesini istediğimiz için tüm süsleri kaldıralım.

Bunun için **süs olarak yerleştirilen bütün kuklaların** (bayraklar, afiş ve balonlar), boşluk tuşuna basınca gizlenmesini istiyorum.

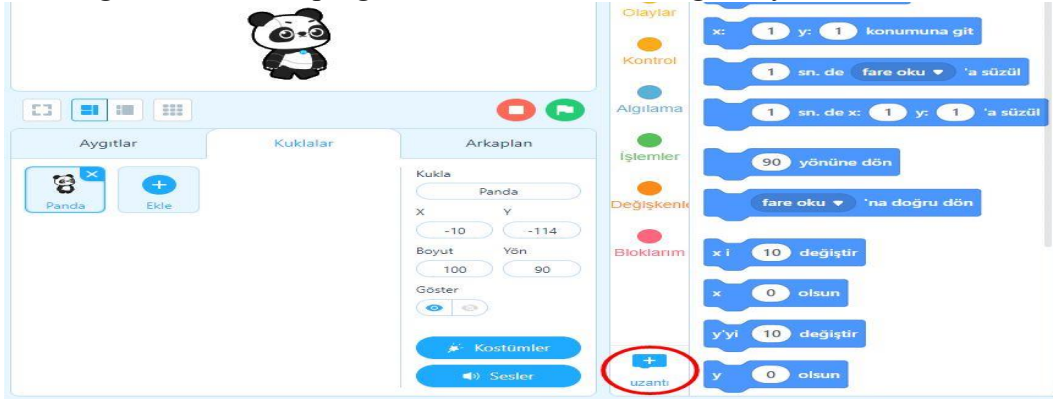
Projenin sonunda, **süs olarak yerleştirilen tüm kuklalarımızın gizlenmesi için** kodlama alanlarına aşağıdaki blokları ekliyoruz. Böylece süs olarak yerleştirilen tüm kuklalarımızın gizlenmesini sağlamış oluruz.



13.HAFTA

1- Çizim Yapıyorum

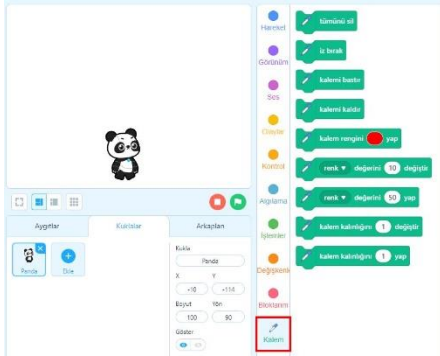
Scratch veya mBlock gibi programlarda **Kalem** grubu blokları kullanarak çizim yapabiliriz. Ancak kalem grubu blokları ilk başta herhangi bir yerde gözükmemekte. Bu sebeple öncelikle Kalem grubu bloklarını programımıza dahil etmemiz gerekiyor.



Bunu yapabilmek için blok gruplarının en altındaki **uzantı** butonunu kullanacağız. Görseli inceleyelim ve alt taraftaki **uzantı** yazan kısmı bulalım. Ardından mBlock uygulamasını açarak bu butona basalım.



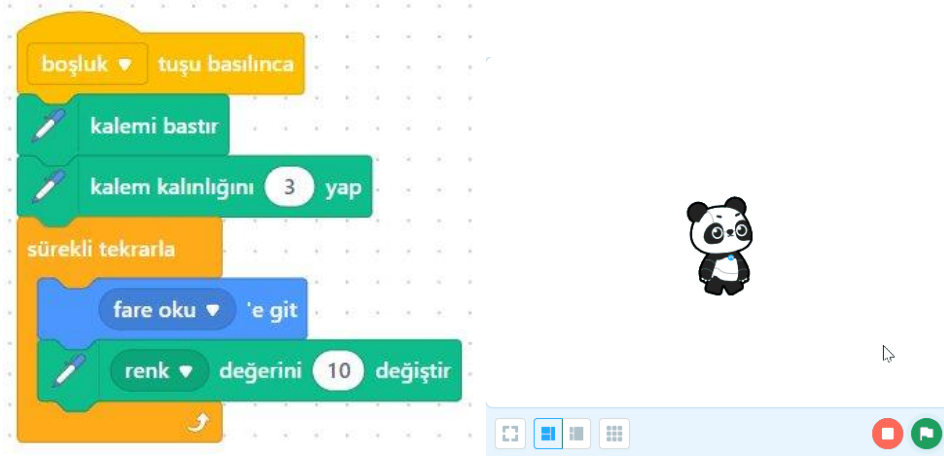
Bu butona bastığımızda uygulamamıza ekleyebileceğimiz çeşitli eklentilerin yer aldığı bir sayfa açılır. Bu sayfada kalem eklentisini bulup Ekle butonuna basalım.



Ekleme işlemi bittiğinde görseldeki gibi Kalem grubu blokları çalışmamıza eklenmiş olur. Şimdi bu blokları kullanarak çizim yapabiliriz. Daha net görebilmek için görselin üzerine tıklayın.

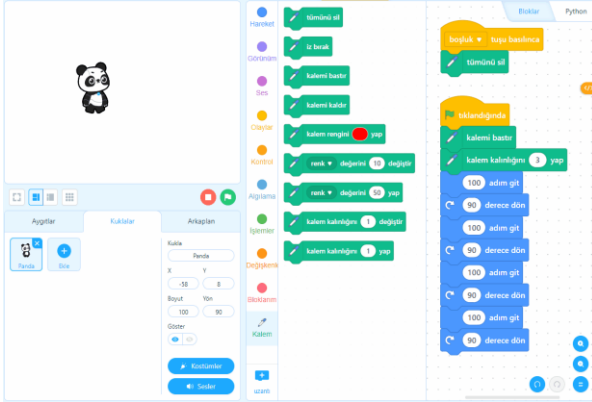
İlk Çizimimiz

Şimdi bir çizim yapalım. [mBlock](#) uygulaması açıksa aşağıdaki görseldeki blokları ekleyin. Farenizi hareket ettirdiğinizde kuklanız rengarenk çizgiler çiziyor mu?

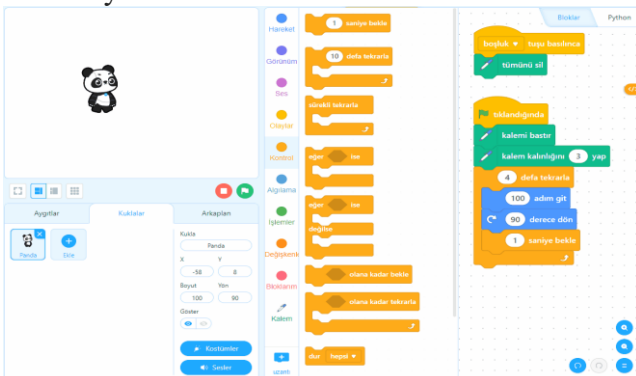


Şimdi de kuklamıza bir **kare** çizdirelim.

Yeni bir [mBlock](#) projesi açalım. Öncelikle çizdiklerimizi silmek için boşluk tuşu basılınca ekranın temizlenmesini sağlayalım. **Olaylar** blok gurubundaki **Boşluk tuşu basılınca** bloğunu ve **Kalem** bloklarından **tümünü sil** bloğunu aşağıdaki gibi ekleyelim. Böylece sahnede bir çizim varsa boşluk tuşuna bastığımızda hepsi silinecek. Sıra geldi kare çizmeye. **Kare** çizmek için, kuklamızın önce bir çizgi çizip, dik açıda 90 derece dönmesi gerekli. Karenin 4 kenarı olduğu için bunu **4 kez tekrarlayacağız**.



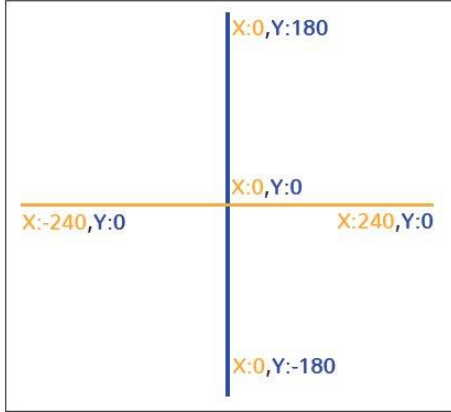
Yukarıdaki animasyonda kuklamız bir kare çiziyor ama **oldukça hızlı**. Hem daha yavaş çizmesi hem de tekrar eden blokları daha kısa hale getirmek için bloklarımızı aşağıdaki gibi düzenleyelim.



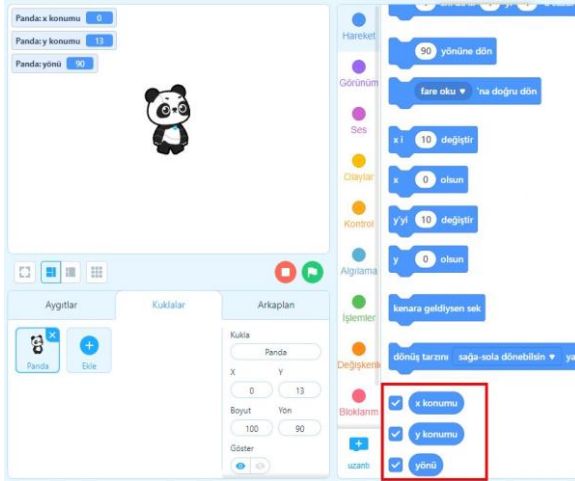
Gördüğünüz gibi kuklamız artık daha yavaş şekilde kare çiziyor.

Nedir Bu x ve y?

Scratch veya mBlock gibi uygulamalarda kuklaların konum veya hareketlerini belirlemek için bir konum belirleme sistemi vardır. Animasyon veya oyun yapımı için oldukça önemlidir.



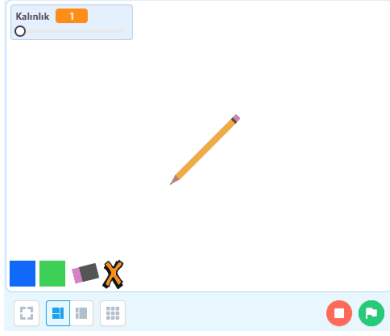
Bu sistem bize bir kuklanın **konumu** hakkında bilgi verir. x **yatay** konumunu, y ise **dikey** konumunu söyler. Sahnenin **tam ortası**, x ile y'nin kesişim noktasında x: 0, y: 0'dır.



Hareket bloklarının en altında **x konumu**, **y konumu** ve **yönü** isminde 3 adet blok var. **x konumu** ve **y konumu** bloklarının başındaki kutucukları işaretlersek, uygulamamızı çalıştırdığımızda kuklamızın konumunu anlık olarak takip edebiliriz. Bu bilgileri ekrandan kaldırmak için kutucukların işaretini kaldırmamız gerekir.

2- Kendi Çizim Programımızı Oluşturalım

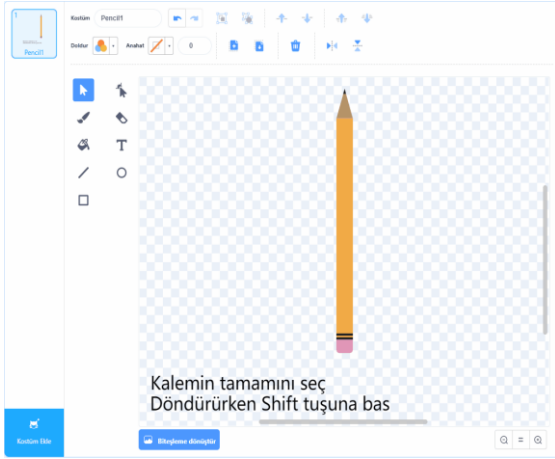
Şimdi kendi çizim programımızı nasıl oluşturacağımızı öğreneceğiz. Çalışma sonunda çizim programımız aşağıdakine benzemeli.



Yeni bir [mBlock](#) sayfası açalım ve aşağıdaki adımları uygulayalım.

Kalemimizi Oluşturalım

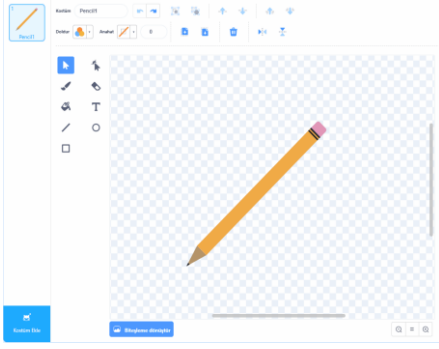
1. Önce panda kuklamızı silelim ve Kukla kütüphanesinden **kalem** (pencil) kuklasını ekleyelim.
2. Eklediğimiz kalemın yönünü çizim yapmaya uygun olacak şekilde döndürelim. Kalemın ucu 45 derece sol altta olacak şekilde aşağıda gösterildiği gibi döndürelim.



3. Yukarıdaki ilk çizimimiz alıştırmasında yaptığımız gibi kalem kuklasının fare okunun peşinden gitmesini istiyoruz. Bunun için sürekli tekrarlar ve fare okuna git bloklarını aşağıdaki gibi yerleştiriyoruz.

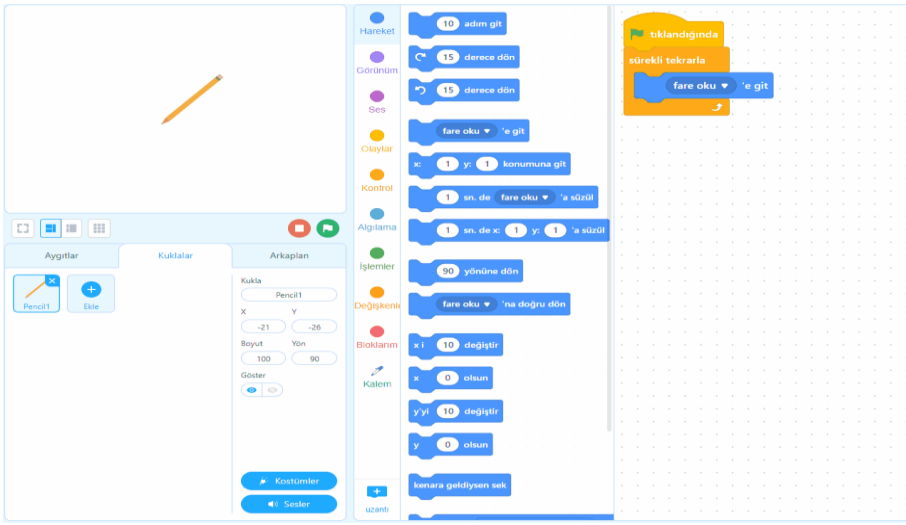


4. Dikkat ettiyseniz kalem fareyi takip ediyor ama farenin ucu kalemın ortasına denk geliyor. **Eğer bu şekilde kalırsa çizimlerimiz kalemın ucuyla değil ortasıyla yapılmış olur.** Bunu düzeltmek için aşağıda gösterildiği gibi, kostüm ekranında kalemın ucunu merkez noktaya taşıyalım.



Animasyonu bayrağa tıklayıp tekrar çalıştırın. Fare okunun ucu, kalem ucu ile birlikte hareket ediyor olmalı.

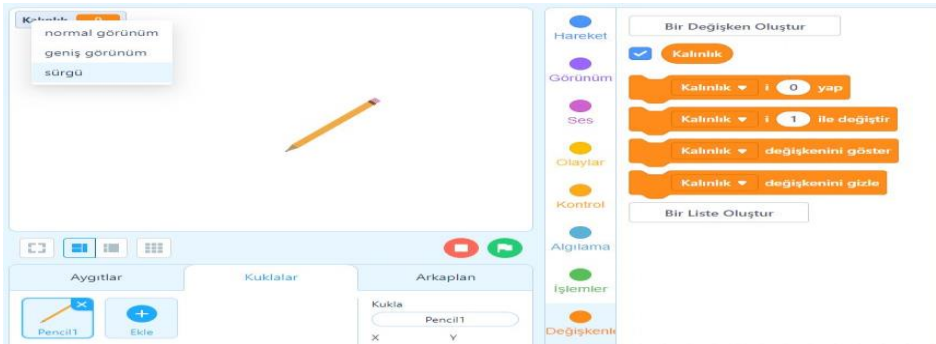
5. Kalemle ilk çizimimizi yapabilmek için bloklarımızı aşağıdaki gibi düzenleyelim.



Blokları yukarıda gösterildiği gibi yerleştirdiyseniz bayrağa tıklayın ve **ilk çizgilerinizi çizin**.

6. Kalem ucu biraz ince oldu. Hadi bunu kalınlaştırabileceğimiz bir ayar yapalım. Bunun için bir değişkene ihtiyacımız var. **Değişkenler** grubundan **Bir değişken oluştur** butonuna basın ve **Kalınlık** isiminde bir değişken oluşturun.

Sol üst tarafta **Kalınlık: 0** şeklinde bir yazı çıktı. Şimdi buna sağ tıklayalım ve çıkan seçeneklerden **Sürgü** seçeneğini seçelim.



Artık sürükleyerek kalınlığı ayarlayabileceğiz. Ama hala çizim yaptığımızda ince çizdiğini fark edeceksiniz. **Kalınlık değerine göre** çizim yapabilmek için oluşturduğumuz değişkenimizden faydalanacağız.

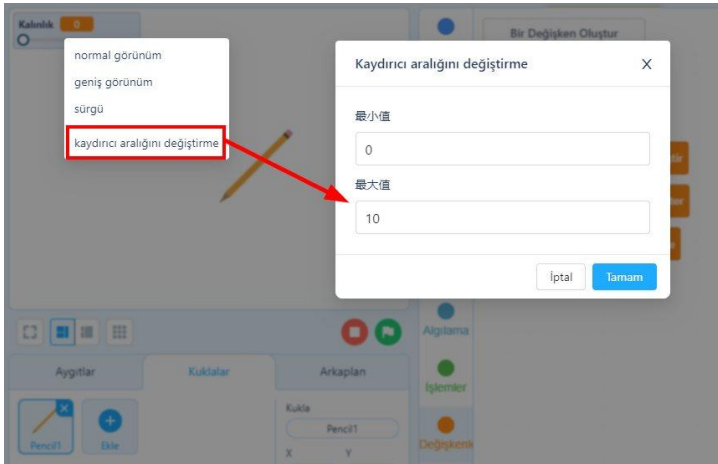
Değişken grubundan **Kalınlık** bloğunu, **Kalem** grubundan **kalem kalınlığını 1 yap** bloğunu ekrana sürükleyip aşağıdaki gibi birleştirelim. (1. ve 2. adım)

2. adımda oluşan blok grubunu eğer bloğu içine ok ile gösterilen yere yerleştirelim.

Ayrıca bayrağa tıkladığımızda ekrandaki çizimleri temizlemesi için **Kalem** grubundan **tümünü sil** bloğunu ve kalem kalınlığını her başlangıçta 1 olarak ayarlamak için **Değişkenler** grubundan **Kalınlık-i 1 yap** bloğunu aşağıdaki 3. adımda gösterildiği gibi birleştirelim ve okla gösterilen yere koyalım.



7. Şimdi de kalınlık değerine **sınırlandırma** yapalım. Sürgüyü hareket ettirdiğimizde 100 değerine kadar gittiğini göreceksiniz. Sınırlama yapmak için tekrar soldaki **Kalınlık: 0** yazan yere, yani sürgüye sağ tıklayalım, açılan seçeneklerden **kaydırıcı aralığını değiştirme** seçeneğine tıklayalım. Açılan pencereden 100 yazan yere 10 yazalım. Aşağıdaki resmi inceleyelim.

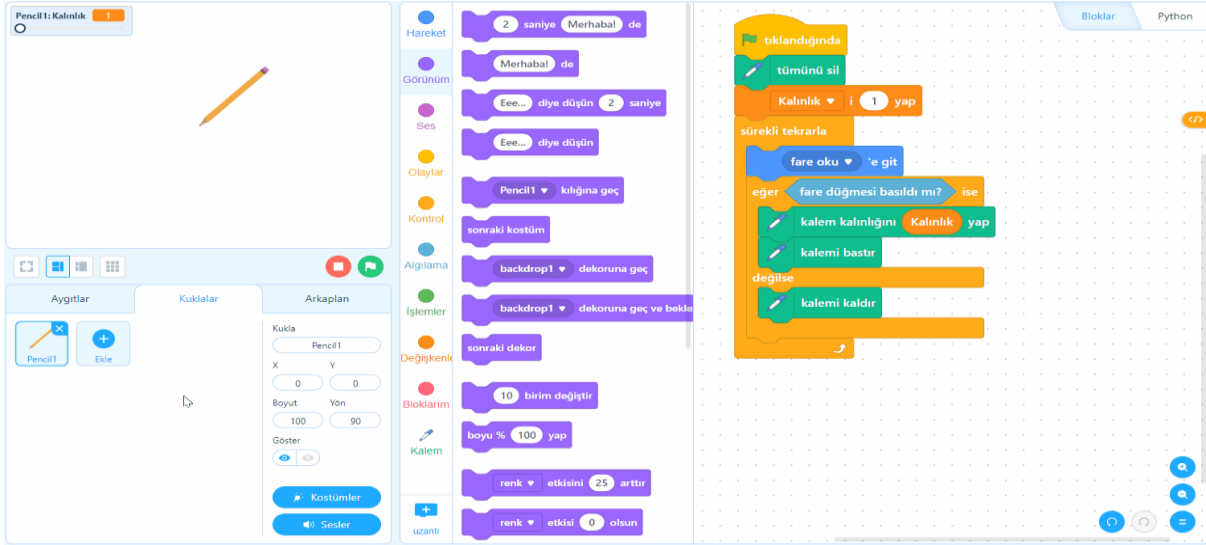


Böylece kalem ile çizim yapabildiğimiz ve kalem kalınlığını ayarlayabildiğimiz bir çizim programı oluşturmuş olduk. Peki farklı renlerde çizim yapmak isteseydik ne yapabiliriz? Şimdi bunu yapalım.

Renk Paleti Oluşturma

Burada 2 farklı renk seçeneği olan bir palet oluşturacağız. Dilerseniz diğer renkler için benzer adımları uygulayarak daha fazla renk seçeneği ekleyebilirsiniz.

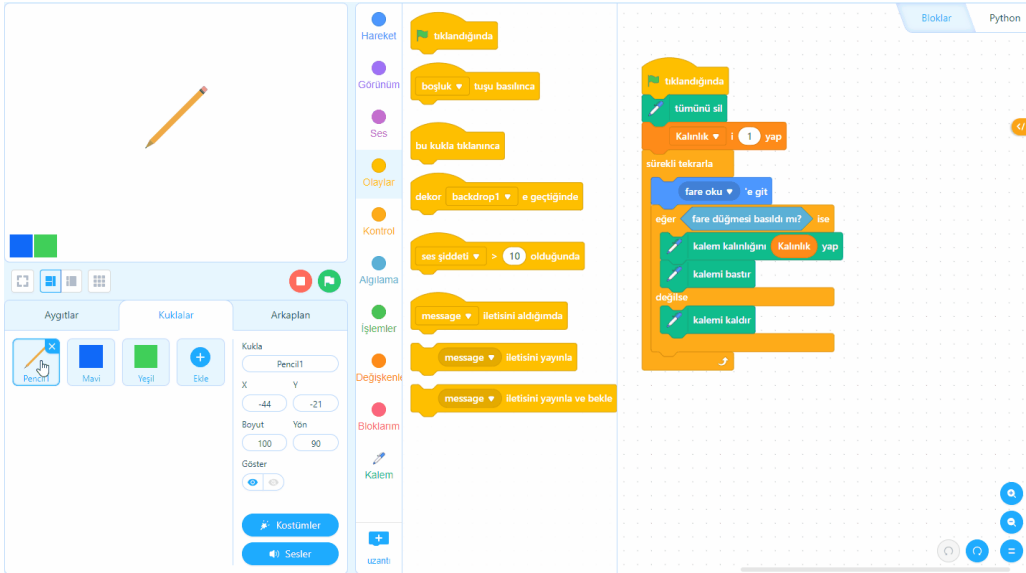
1. Renk paleti oluşturmak için **Kuklalar** bölümünden **Ekle** butonuna basalım. Açılan pencerenin sağ üst tarafındaki **Boya** sekmesine tıklayalım. Karşımıza kukla çizim alanı çıkacak. Burada bir mavi kare oluşturalım ve adımı da mavi olarak değiştirelim.



2. Bir tane de benzer adımları uygulayarak yeşil renk seçeneği oluşturalım. Onu da sol alt tarafa mavinin yanına yerleştirelim. (Mavi karenin bir kopyasını oluşturup onu yeşil renge boyayarak da ikinci rengimizi oluşturabiliriz.) Oluşturduğumuz yeşil karenin adını **yesil** yapmayı unutmayalım.
3. Şimdi de kalemın, seçtiğimiz renkte görünmesini sağlayalım. Yani mavi renkle çizim yapacağımız zaman kalemimiz de mavi olsun.

Bunun için kalem kuklasını seçelim ve kostüm düzenleme ekranında bir kopyasını oluşturarak gövdesini mavi renge boyayalım. Üst kısımdan adını **mavi-kalem** yapalım.

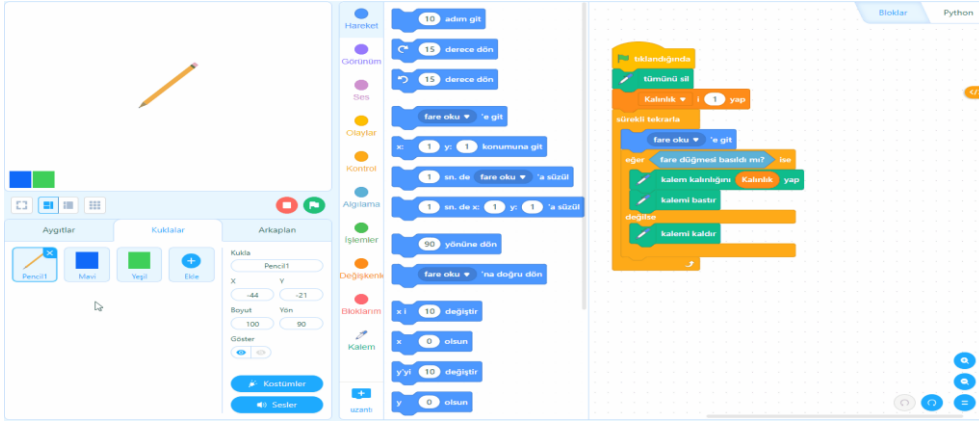
Yeşil rengi seçtiğimizde kalemın yeşil gözükmesi için de yine kalem kuklası seçiliyken kostüm düzenleme ekranında mavi kalemde bir kopya oluşturup gövde rengini **yeşil** yapalım. Üst kısımdan adını **yesil-kalem** yapmayı unutmayalım.



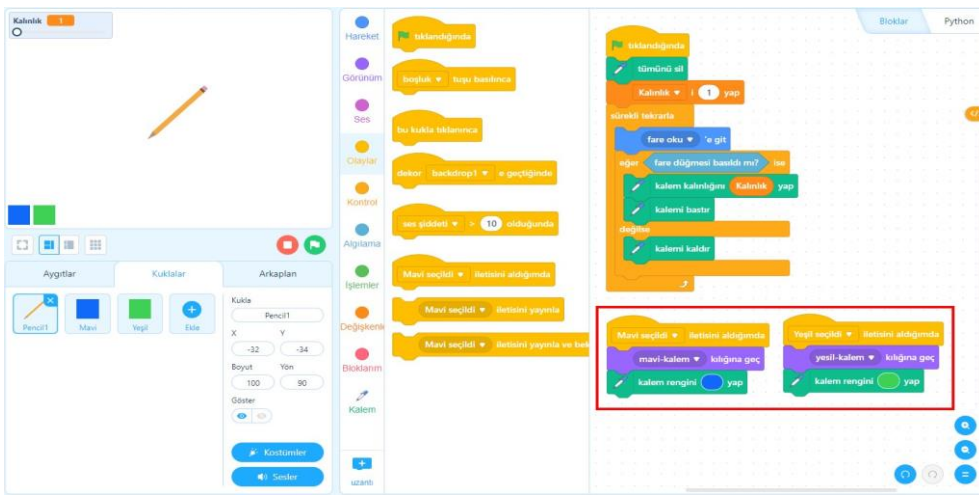
4. Peki kalemimiz nasıl renk değiştirecek?

Bunun için **ileti yayınlama** (habersal) bloklarını kullanacağız. Sol alta yerleştirdiğimiz Mavi kare kuklamız seçiliyken **Olaylar** grubundaki **bu kukla tıklanınca** ve **message iletisini yayımla** bloklarını ekleyelim. **message** yazan yere tıklayıp yeni bir ileti oluşturalım ve adını **"Mavi seçildi"** koyalım.

Aynısını yeşil kare için de yapmamız lazım. Sadece burada oluşturacağımız haber iletisi **"Yeşil seçildi"** şeklinde olmalı.



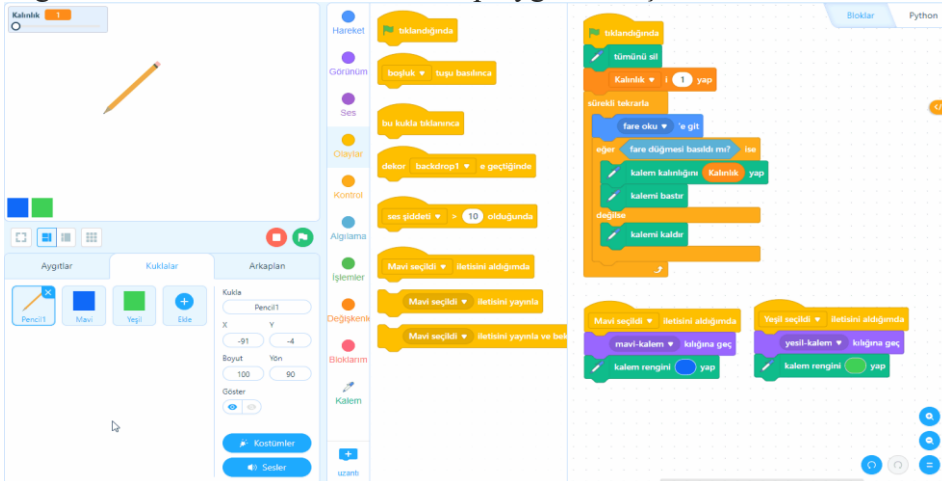
5. Şimdi de kalemimizin bu haberleri alınca **kendi rengini** ve **çizgi rengini** değiştirmesini sağlamamız gerekiyor Bu yüzden aşağıdaki blokları **kalemimiz** seçiliyken eklememiz gerekiyor.



Silgimizi Yapalım

Çizdiklerimizi sileceğimiz silgimizi yapabilmek için önce sol alttaki renk paletimizin yanına bir silgi kuklası eklemeliyiz.

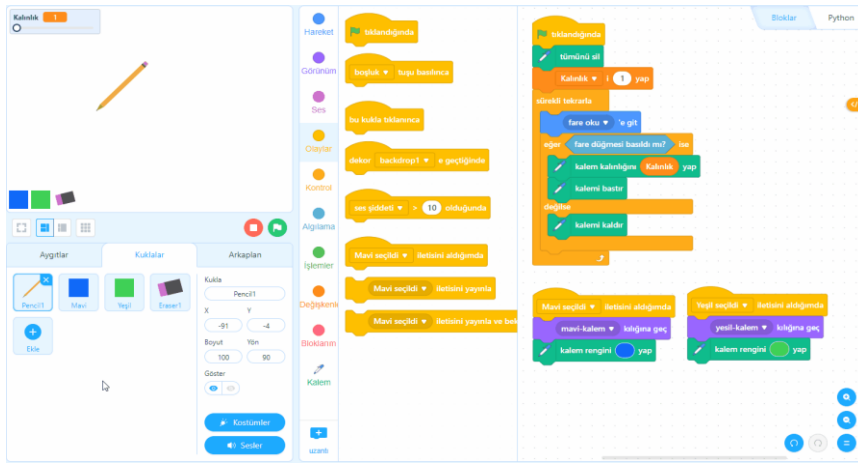
Kukla kütüphanesinden **eraser** yazarak çıkan silgi kuklasını ekleyelim. Dilersek eklediğimiz silgi kuklasını biraz daha renklendirip uygun bir açıda döndürebiliriz.



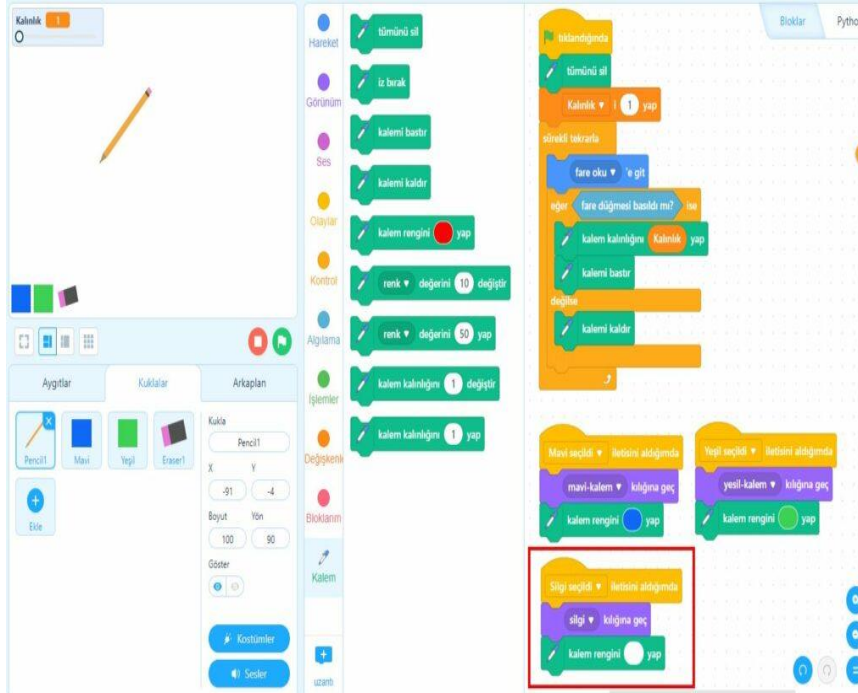
Bu kukla seçiliyken **Olaylar** grubundaki **bu kukla tıklanınca** ve **message iletisini yayımla** bloklarını eklemeliyiz. Yeni ileti olarak **silgi seçildi** iletisini eklemeliyiz.



Şimdi de silgiyi seçtiğimizde kalemimizin silgi gibi görünmesini sağlamalıyız. Bunun için kalem kuklasını seçip Kostüm ekranında mavi ve yeşil kostümlerin altına bir silgi kostümü eklemeliyiz. Daha önce eklediğimiz silgi kostümünü kopyalayıp kalem içine kostüm olarak yapılandırabiliriz.



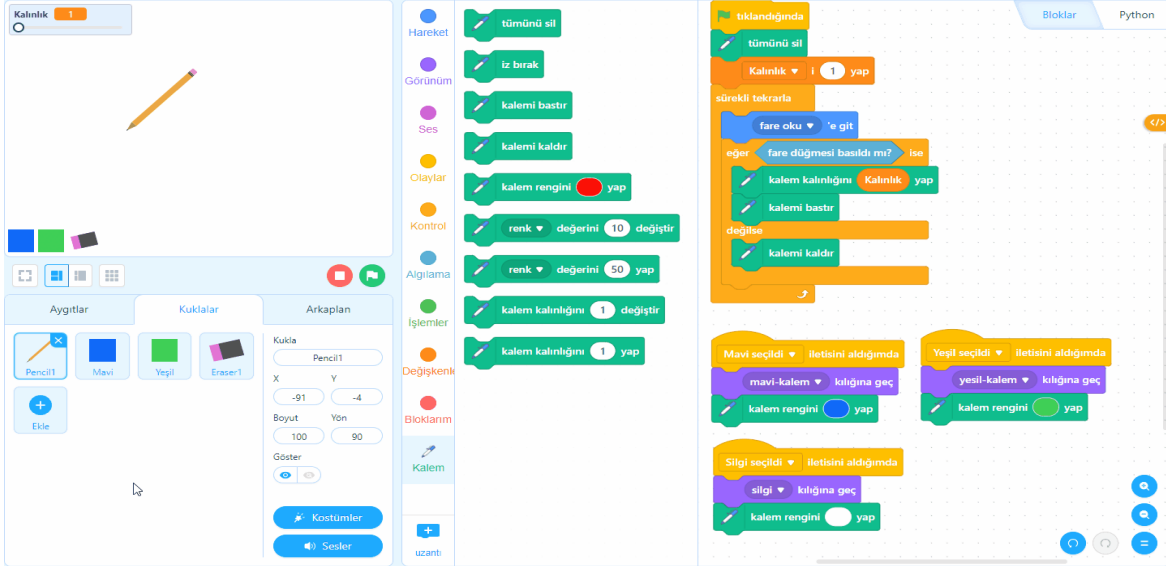
Son olarak bloklarımızı ekleyeceğiz. Kostüm ekranından çıkalım ve kalem kuklası seçiliyken **Olaylar** grubundan **silgi seçildi iletisini aldığımda** bloğunu, **Görünüm** grubundan **silgi kılığına geç** bloğunu ve **Kalem** grubundan **kalem rengini beyaz yap** bloğunu eklemeliyiz.



Silginin çalışma mantığı aslında beyaz çizgi çizmekten ibaret. Çalışma alanımız beyaz renkte olduğu için siliniyormuş gibi görünüyor.

Tümünü Sil Seçeneği Ekleyelim

Bazen belli yerleri silmek yerine bütün resmi temizlemek de isteyebiliriz. Bunun için de bir düğme ekleyelim. Temizleme düğmesi olarak kuklalardan X harifini seçebiliriz. X kuklasını silginin yanına uygun boyutta yerleştirelim.



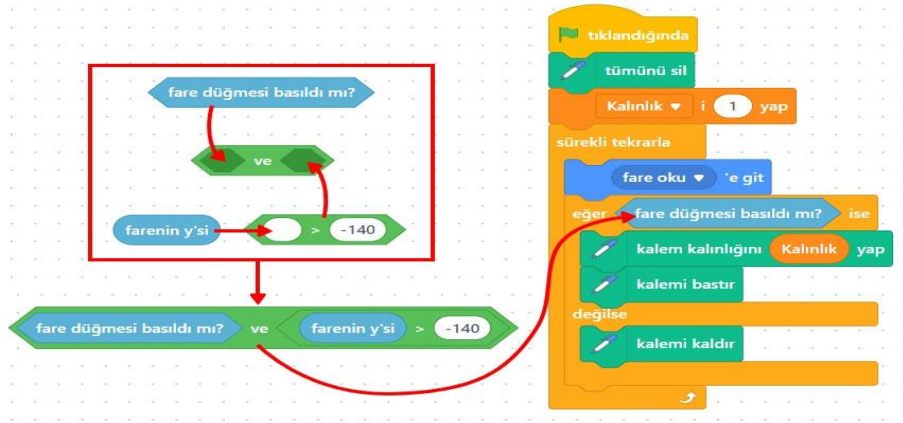
Ardından bu kukla seçiliyken **Olaylar** grubundan **bu kukla tıklanınca** ve **Kalem** grubu bloklarından **tümünü sil** bloklarını ekleyelim.



Çizimi Belirli Alanda Sınırlamak

Çizerken göreceğiniz gibi kalemimiz renk paleti ve silgi arkasında da çizim yapmaya devam ediyor. Bu alana çizim yapmasını engelleyebiliriz. Bunun için kaleme eklediğimiz bloklardaki **eğer** bloğunda ufak bir değişiklik yapacağız. Bu değişikliği yaparken yukarıda anlatılan **x** ve **y** değerlerinden ve **algılama** ile **işlem** bloklarından yararlanacağız. Renk paleti kuklaları ve silgi kuklalarının üst hizaları, y ekseninde yaklaşık olarak -140 konumuna denk gelmekte. Bu sizin hazırladığımız çalışmada farklı bir değer olabilir. Kalemın y konumu -140 değerinden daha büyük olduğunda çizecek -140 ve daha küçük bir değere denk geldiğinde ise çizim yapmayacak.

Aşağıdaki blokları inceleyelim ve kalem bloğuna eklediğimiz **eğer** kısmında gerekli düzenlemeyi yapalım.



Basit düzeyde hazırladığımız çizim programımızı böylece tamamlamış olduk. Dilerseniz renk paletine farklı renkler ekleyerek çizim programınızı daha da geliştirebilirsiniz.

14.HAFTA

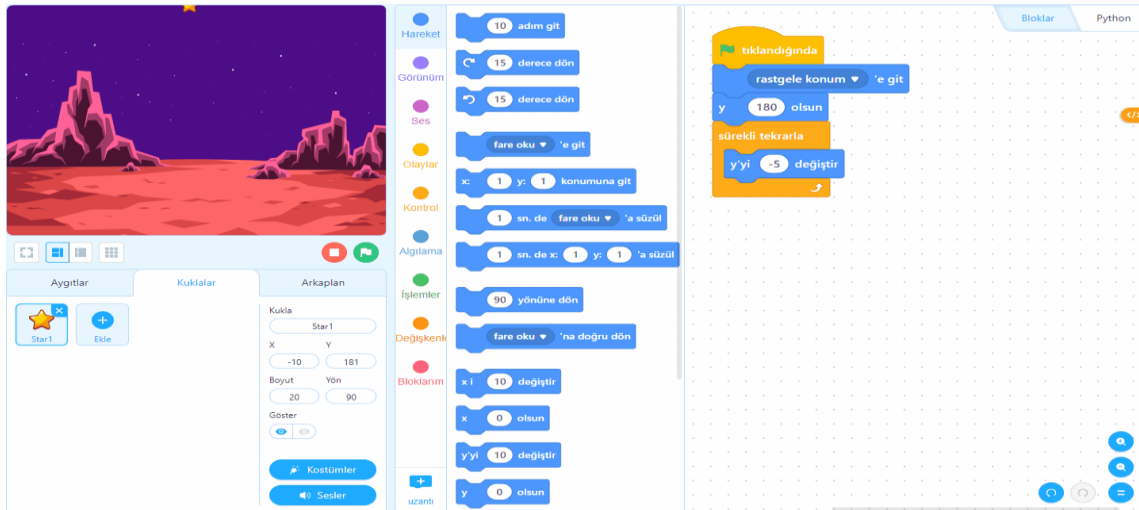
- Yıldız Toplamaca

mBlock programında gittikçe ustalaşyoruz. Eğer önceki konularda eksikler varsa mutlaka tekrar ederek konuları daha iyi kavrayalım.

Şimdi gökten düşen yıldızları yakalayabileceğimiz bir oyun yapacağız. Bu oyunu derseniz düşen elmaları sepete toplama fikriyle de yapabilirsiniz.

Aşağıdaki adımları sırayla uygulayalım.

1. [mBlock](#) programını açalım ve öncelikle panda kuklamızı silelim.
2. Gökyüzünden yıldız düşme fikrine uygun bir arkaplan görseli ayarlayalım. Örneğin mBlock arkaplan kütüphanesindeki **Planet1** arkaplanı oldukça uygun bir arkaplan.
3. Şimdi de düşecek nesnemizi yani yıldız kuklasını ekleyelim. Kukla kütüphanesinde arama kutusuna **star** yazalım ve çıkan kuklalardan **Star1** kuklasını ekleyelim.
4. Yıldız kuklamız biraz büyük. Boyut ayarı kısmından 100 olan boyutunu **20** yapalım.
5. Şimdi yıldızın yukarıdan herhangi bir yerden düşmesini sağlamak için aşağıdaki blokları kullanalım.



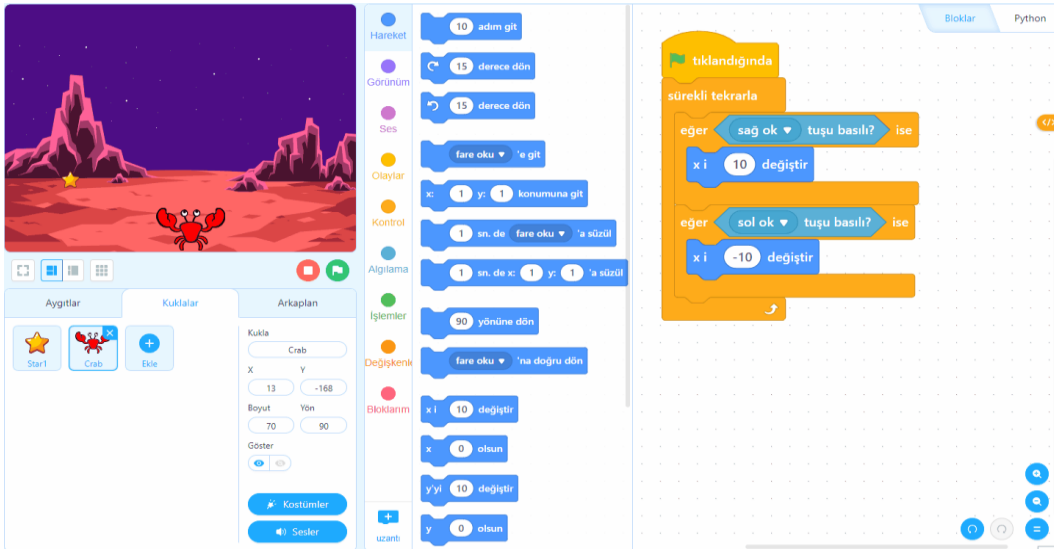
“Rastgele konum’a git” bloğunu bulabildiniz mi? (fare oku’na git bloğunu inceleyin)

“Rastgele konum’a git” bloğu yıldızın rastgele bir yerden düşmesini, “y, 180 olsun” bloğu yıldızın en yukarıdan düşmeye başlamasını, “y’yi -5 değiştir” de yıldızın aşağı doğru hareket etmesini sağlar.

6. Yıldızımız yere düşüyor ancak oyunun sürekliliğini sağlamak için ekranın en altına düştüğünde yukarıdan yeniden başlaması gerekiyor. Bunu sağlamak için kod bloğumuza aşağıdaki gibi düzenliyoruz. (sürekli tekrarla içine eğer bloğu grubunu ekliyoruz.)

```
when green flag clicked
  go to random position
  set y to 180
  repeat (forever)
    change y by -5
  if (y position < -170)
    go to random position
    set y to 180
```

7. Oyunumuz için yıldız yakalayacak bir de **kuklaya** ihtiyacımız var. Bunun için kukla kütüphanesinden **Crab** (yengeç) kuklasını ekleyelim. Kuklamızın **Boyut** değerini de **70** yapalım.
8. Şimdi kuklamızı klavyeden sağ ve sol ok tuşlarıyla kontrol edebileceğimiz bir hâle getirelim. Aşağıdaki görselde yer alan blokları inceleyelim. Bu bloklar sayesinde sağ ve sol ok tuşları ile kuklamızı hareket ettiriyoruz. (Bu blokların Crab kuklasına eklendiğini unutmayın.)

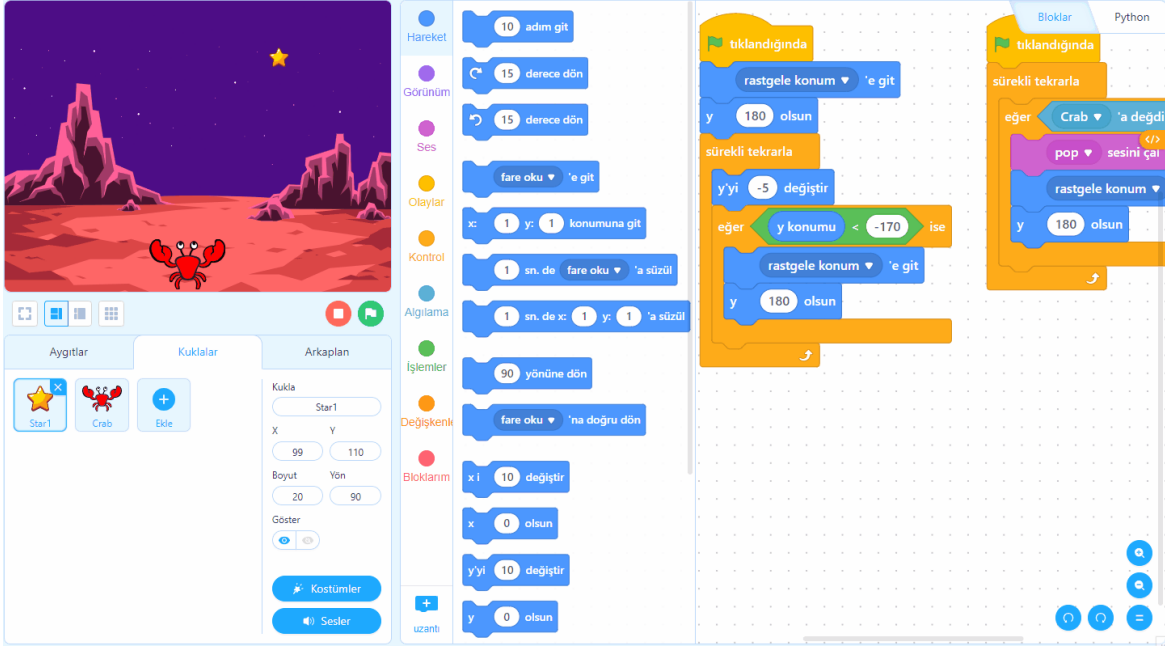


9. Kuklamız yıldızla dokunduğunda ses çıkması için, **yıldız kuklamızı seçelim** ve aşağıdaki kırmızı kutu içindeki blokları (önceki bloklarımızın yanına) ekleyelim.

```
when green flag clicked
  go to random position
  set y to 180
  repeat (forever)
    change y by -5
    if (y position < -170)
      go to random position
      set y to 180
  if (Crab 'a' touched)
    play sound
    go to random position
    set y to 180
```

Yıldız kuklasına eklenecek yeni bloklar
Yıldız, Crab'a değerse ses çal ve rastgele konumuna git, ekranın üstünde y=180 konumuna git

10. Oyunu biraz daha eğlenceli hâle getirelim. Yıldız topladıkça artan bir puanlama sistemi ekleyeceğiz. Böylece topladığımız yıldız sayısını takip edebiliriz. **Değişkenler** grubundan “Bir Değişken Oluştur” butonuna basalım, açılan pencerede değişken adına “**Puan**” yazalım.



11. Bloklarımızı puan hesaplayacak şekilde aşağıdaki gibi güncelleyelim. (Kırmızı kutu içindeki blokları önceki bloklarımızın arasına ekliyoruz.)

Aşağıdaki kod ile yıldız kuklası, Crab kuklasına her değdiğinde bir puan arttırıyoruz. Oyun yeniden başladığında ise puanları sıfırlıyoruz.



12. Şimdi oyunu biraz daha heyecanlı hale getirmek için yıldızların sayısını arttıralım. Bunun için yıldız kuklasına sağ tıklayıp **kopyasını çıkart** seçeneğini seçelim. Böylece yıldız kuklası, ona eklediğimiz **bloklarla beraber** kopyalanmış olacak. Bu işlemi bir defa daha yapalım. Toplam 3 yıldız oldu.

Yıldızların farklı hızlarda düşmesini istiyorsanız blokların içindeki **y'yi -5 değiştir** bloğundaki değeri her yıldız için farklı yapabilirsiniz. Örneğin yıldız kopyalarından biri için -6, diğeri için -7 yazabilirsiniz.

Yıldız toplamaca oyunu bu kadar. Eğer adımları dikkatlice uyguladıysanız sayfanın başında yer alan animasyondaki gibi yıldız toplama oyununu yapmış olmalısınız.

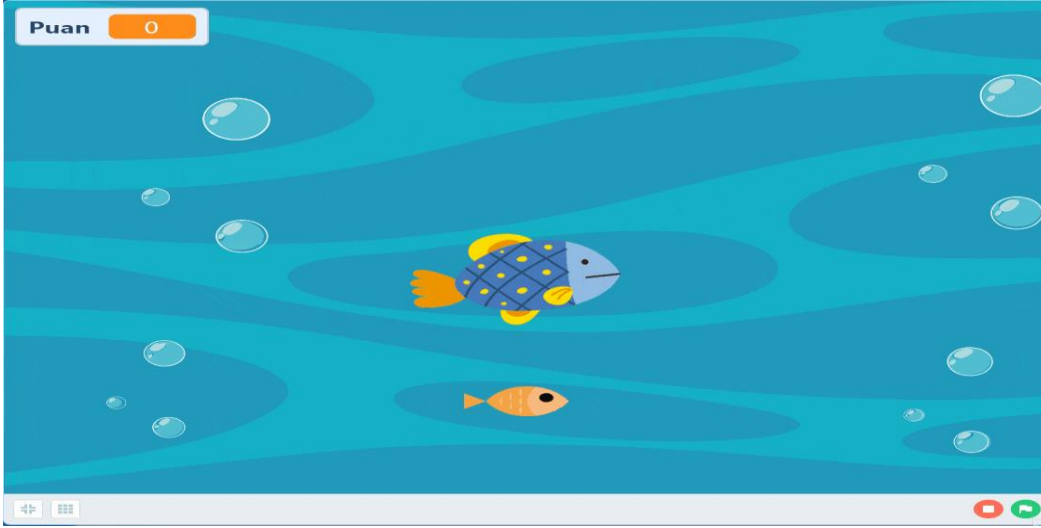
Oyunu başarılı bir şekilde yaptıysanız şu özellikleri de ekleyebilirsiniz:

- Yıldız sayısını artırma

- Yakalanamayan yıldızlar için farklı ses çalma
- Yakalanamayan yıldızlar için Puandan 1 eksiltme
- Süre sınırı koyma. (Örneğin 20'den veya 10'dan geriye sayacak ve süre 0 olduğunda oyun bitecek.)
- Süre sıfır olduğunda arkaplan değişecek. (Oyunun bittiğini gösteren farklı bir arkaplan gözükecek.)

2- Akvaryum Oyunu

Yeni bir yakalama oyunu daha yapalım. Akvaryum içinde büyük balık küçük balıkları yakalasin. Oyunu tamamladığınızda aşağıdaki gibi olmalı.



Aşağıdaki adımları sırayla uygulayalım.

1. mBlock uygulamasını açalım ve panda kuklamızı silelim.
2. Arkaplan kütüphanesinde arama kutusuna **water** yazalım ve çıkan **water1** isimli arkaplanı ekleyelim.
3. Şimdi biri büyük biri küçük olmak üzere iki tane farklı balık ekleyeceğiz. Öncelikle küçük balığımızı ekleyelim. Kukla kütüphanesinde **Hayvanlar** kategorisini seçelim ve arama kutusuna **fish** yazalım, çıkan sonuçlardan **Fish7** isimli balığı ekleyelim. Balığımızı ekledikten sonra boyutunu **40** yapalım.



4. Sıra büyük balığı eklemeye geldi. Yine Kukla kütüphanesinden **Hayvanlar** kategorisinde **fish** araması yapalım ve çıkan sonuçlardan **Fish2** kuklasını seçip ekleyelim ve boyutunu **80** yapalım.

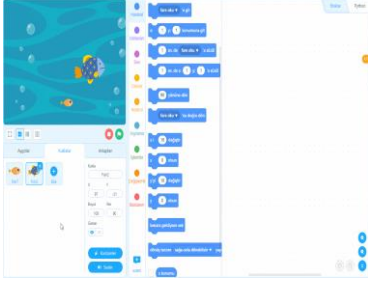
Fish2 balığı seçiliyken **Kostümler** ekranında önce adını **büyük balık 1** olarak değiştirelim. Daha sonra biraz düzenleme yapmalıyız.

Balığın bir kopyasını oluşturalım. Yeni oluşturduğumuz kopya kostümün adı otomatik olarak **büyük balık 2** oldu. Bu kostüme açık bir ağız çiziceğiz. Silgimizin boyutunu **60** yapıp balığın ağız kısmını

silelim (iki defa ağız kısmına silgi ile tıklayabiliriz) ve fırça ile açık bir ağız çizelim. Çizimlerinizde

komik olmak serbest izleyebilirsiniz.

Tüm bu adımların nasıl yapıldığını aşağıdaki animasyondan



5. **Şimdi geldik kodlama kısmına.** Önce kırmızı küçük balığımız için bir kod yazalım. Balığımız akvaryumda dolaşacak. Eklediğimiz **rastgele sayı tut** sayesinde balığımızın hareketleri rastgele olacak ve daha doğal görünecek. **Küçük balık kuklasına aşağıdaki blokları ekleyelim.**



6. Şimdi de küçük balığın yakalanmasını sağlayalım. Küçük balığın herhangi bir yeri (kafası, gövdesi veya kuyruğu) büyük balığın sadece kafa kısmına değdiğinde yakalanmasını istiyoruz, büyük balığın kuyruğuna ya da yüzgeçlerine değdiğinde değil. Bu nedenle **Algılama** grubundaki **“... rengi... rengine değdi”** bloğunu kullanacağız. Burada küçük balık için dikkat etmemiz gereken bir konu var. Küçük balığın kafa kısmı açık turuncu iken gövde ve kuyruğu daha koyu bir turuncu renkte. Bu sebeple küçük balık için iki rengi de içerecek şekilde blokları düzenlemeliyiz.

Küçük balığımıza ait bloklarımızı aşağıdaki gibi güncelleyelim. (Blok içindeki renklere birine tıkladığımızda açılan renk seçme penceresindeki damlalık aracıyla sahneden istediğiniz rengi seçebilirsiniz.)



Yukarıdaki bloklarla küçük balık, büyük balığın ağızına dokunduğunda **ileti yayımla (habersal)** bloğu ile **yakalandın** haberini gönderiyoruz ve küçük balık “**gizlen**” bloğu ile kayboluyor. Küçük balık **3 sn** sonra rastgele bir yerde tekrar ortaya çıkıyor.

7. Sıra geldi büyük balığımıza. Balığımızın hareketlerini **fare oku** ile kontrol etmek istiyoruz. Ayrıca **yakalandın** iletisi alınınca **Chomp** sesi çalacak ve büyük balık ağızını açıp kapayacak. Büyük balığı seçip aşağıdaki blokları ekleyelim..



Dikkat ettiyseniz bir önceki Yıldız Toplamaca oyununda olduğu gibi burada da bir puan değişkeni oluşturduk, böylece kaç balık yakaladığımızı da görebileceğiz.

Akvaryum oyunu da bu kadar.

Oyunu başarılı bir şekilde yaptıysanız şu özellikleri de ekleyebilirsiniz:

- Küçük balık sayısını artırma (Örneğin 3 küçük balık ile oynama)
- Büyük balığın boyutunu, her küçük balığı yakaladığında 1 birim büyütme.
- Süre sınırı koyma. (Örneğin 20'den veya 10'dan geriye sayacak ve süre 0 olduğunda oyun bitecek.)
- Süre sıfır olduğunda arkaplan değişecek. (Oyunun bittiğini gösteren farklı bir arkaplan gözükecek.)

Kaynak: btdersi.com

Site adminine teşekkür ederiz.