

EL NOTU

MODÜL 1 – DERS 1

EL NOTLARI

Bilgi İşlemsel Düşünme Kavramları

Bilgi işlemsel düşünme kavramları, bilgi işlemsel düşünme alanının altında yatan temel fikir ve ilkeleri ifade eder.

Sıra

Sıra, bir sürecin adımlarının takip edildiği sıradır.



Şekil 1 – Sıra

Sıralı adımlar da dahil olmak üzere faaliyet örnekleri:

- Kek yapmak
- Kumaş katlama
- Deneme yazmak
- Diş fırçalamak
- Okula gitmek
- Dikdörtgen çizme
- Seksek oynamak

Bu süreçlerin adımlarını tanımlamak için [Çalışma Kağıdı 1](#)'i kullanın.

EL NOTU

Döngü

Aynı adım bir dizi üzerinde yinelendiğinde, bu diziye "döngü" veya "yineleme" adını veririz. Bilgisayar biliminde döngü, belirli bir koşul karşılanana kadar sürekli olarak tekrarlanan bir dizi talimattır.



Şekil 2 – Döngü

Döngü için faaliyet örnekleri:

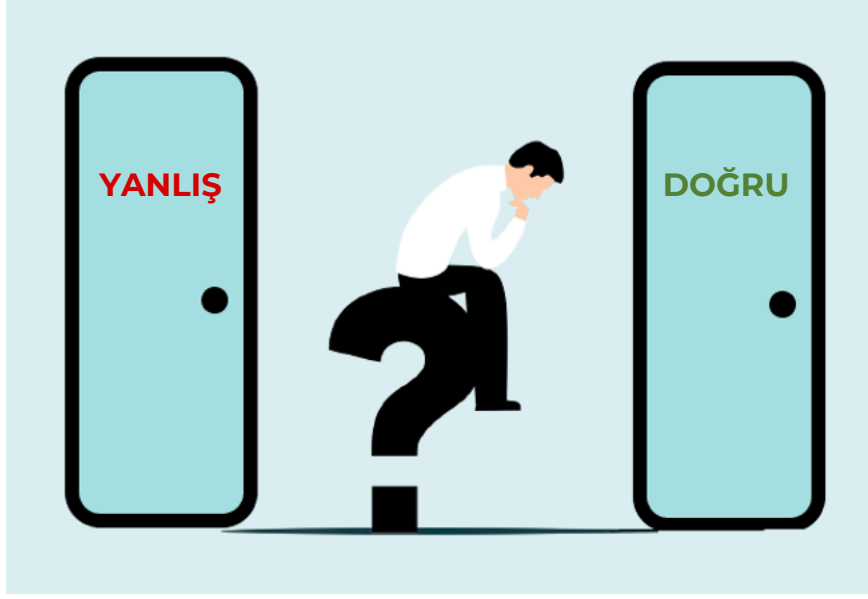
- Dans ederken sürekli zıplıyoruz
- Bir çiviye duvara çakmak için tekrar tekrar çekiçlemek
- Bir şarkıyı tekrar moduna alma
- Şampuanı suyla durulamak için saçı tekrar tekrar ovmak
- Kanal tıkama
- Yıkama tabakları
- Aynı istasyonlarda durmak (metro, halk otobüsü)

Bir döngü içinde döngüler olduğunda ne olduğunu görmek için bunu kullanın: [Çalışma Kağıdı 2](#)

EL NOTU

Koşullar

Bilgisayarlar önceden belirlenmiş koşullara göre ve bunların doğru ya da yanlış olmasına bağlı olarak karar verirler. Yalnızca belirli kriterler karşılandığında, yani doğru ya da yanlış olduklarında çalışırlar. Bir koşullu ifade "if", "else" veya "else if" ifadelerini içerir.



Şekil 3 – Koşullar

Koşullu ifadelere örnekler:

- "Eğer" ifadesi
- Eğer yağmur yağarsa, şemsiyenizi alın, (Evet, yağmur yağar)
- "Else" ifadesi
- eğer yağmur yağmazsa (else deyimi), şemsiyenizi almayın. (Hayır, yağmur yağmıyor)
- "Else if" ifadesi
- Yağmur yağmazsa, spor ayakkabılarınızı giyin. (Hayır, yağmur yağmazsa bu sefer başka bir şey yaparsınız).

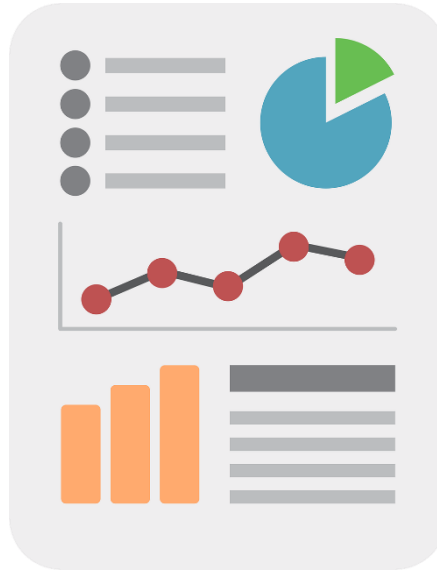
[Çalışma Kağıdı 3](#)'ü koşullar için daha fazla pratik yapmak amacıyla kullanabilirsiniz.

EL NOTU

Veri

Bilgi parçaları, olguları, kavramları veya görevleri sırayla içerebilen "verileri" oluşturur, böylece bunları kolayca yorumlayabilir veya işleyebiliriz.

Topladığımız veriler neyi sorguladığımıza bağlı olarak değişir ya da çeşitlenir. Örneğin toplantı katılımcılarımızı ilgi alanlarına göre sınıflandırarak toplantı sonunda onlara farklı hediyeler verebiliriz. O zaman ilgi alanı kişiden kişiye farklılık gösterdiği için veri toplama sürecinde değişkenimiz olacaktır.



Şekil 4 – Veri

Veri Örnekleri :

- Bahçede yaşayan hayvanların listesi
- Sınıftaki yeşil kalemler
- Harry Potter film serisini izleyenler
- Katı diyet kurallarına karşı olumlu bir tutum sergileyenler
- Sosyal medyayı aktif olarak kullananlar
- En sevdiği ilgi alanı spor olanlar

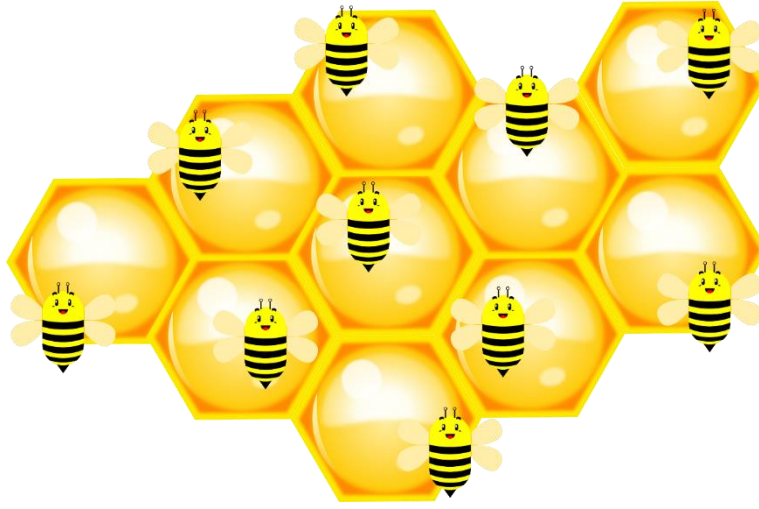
İsteğe bağlı etkinlik: [Çalışma Kağıdı 4](#) veri hakkında daha fazla pratik için kullanılabilir.

EL NOTU

Modülerlik

Modülerlik, görevleri veya prosedürleri daha küçük, daha basit ve daha yönetilebilir birimlere ayırmaktır.

Modülerlikte, bir bütünün her bir biriminin kendi başına işlevselliğine odaklanmamız gerekir. Bir odadaki üç elektrik prizini düşünelim. Çamaşır makinesi, bulaşık makinesi ve kurutma makinesi bu prize ayrı ayrı takılabilir ve mekanizmaları aynı çalışma prensibine dayandığı için her bir priz bağlı olduğu makinenin çalışmasını sağlar.



Şekil 5 – Modülerlik

Modülerlik örnekleri :

- Bir fabrikanın aynı birimindeki işçilerin veya robotların faaliyetleri/görevleri
- Bir düğün töreninde davetliler için servis tabaklarının hazırlanması
- Bir araba (çıkarılabilir parçalarla)
- Bir masaüstü bilgisayar (çıkarılabilir ve yükseltilebilir parçalara sahip)
- Prefabrik evler

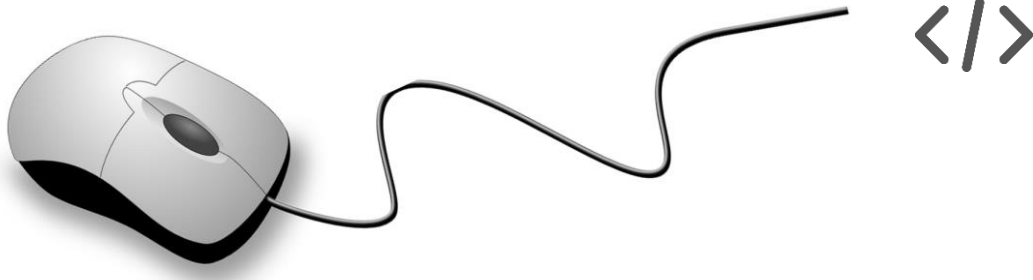
Donanım/Yazılım

Donanım, bir hesaplama sisteminin çalışması için yazılıma veya talimatlara ihtiyaç duyan fiziksel parçalarını ifade eder. Biz insanlar, bedenlerimiz sayesinde fiziksel bir bütünlüğe

EL NOTU

sahibiz (Donanım). İletişim için kullandığımız dil bizi insan yapar (Yazılım). Sözlü, sözsüz, yazılı veya görsel (Programlar/Uygulamalar) gibi farklı iletişim araçları kullanırız.

Burada önemli olan, bilgisayarların ve diğer akıllı cihazların sihirle çalışmadığının farkına varmaktır. Tüm bilgisayarlar, tabletler ve telefonlar insan mühendisliğinin ürünüdür ve erken eğitimin ana hedefi çocukların bunları anlamasını sağlamak olmalıdır.



Şekil 6 – Donanım/Yazılım

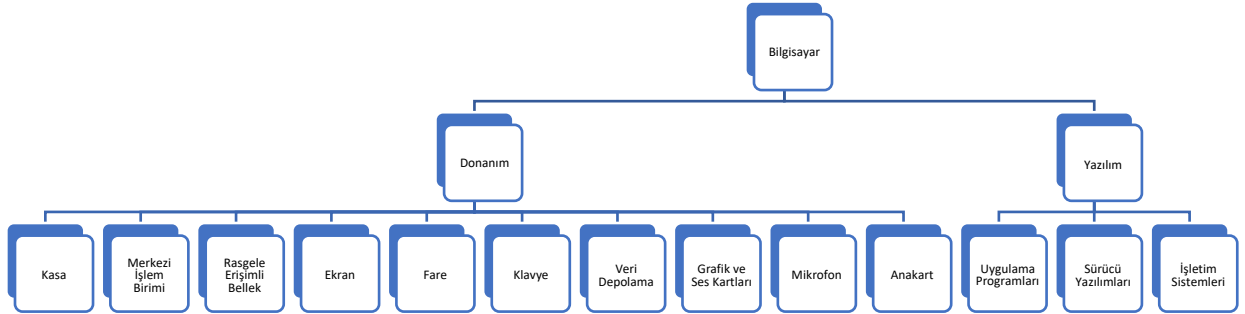


Diagram 1 – Donanım/Yazılım

Hata ayıklama

Hata ayıklama, bir hedefe ulaşmak için sorunları tanımlamak ve hataları ortadan kaldırmaktır. Başka bir deyişle hata ayıklama, bir sorunun çözümüne yönelik adımların doğruluğunun ve uygunluğunun değerlendirilmesidir.

Örneğin, bir yazar yazısını bitirdikten sonra dilbilgisi kurallarına, noktalama işaretlerine veya hedef okuyuculara uygunluğuna göre gözden geçirerek ve gerekli düzeltmeleri yaparak hata ayıklaması yapar.

EL NOTU



Şekil 7 – Hata ayıklama

Hata ayıklama örnekleri :

- Araç çalışmadığında akü veya yakıt seviyesinin kontrol edilmesi
- Dilbilgisi hataları için metin kontrolü
- Bir tarifteki hataları kontrol etmek (örneğin, 1 bardak süt yerine 2 bardak yazılması)

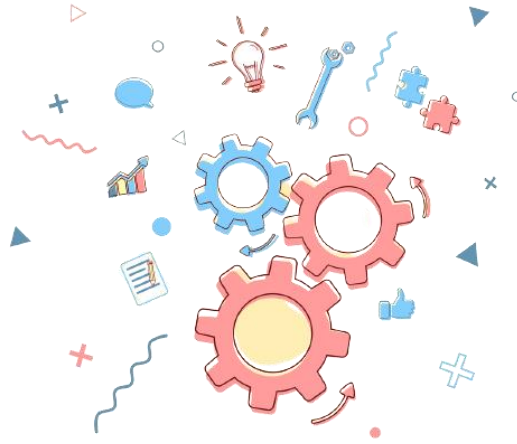
Daha fazla pratik yapmak için [Çalışma Kağıdı 5](#)'i kullanabilirsiniz.

EL NOTU

Tasarım Süreci

Tasarım süreci, bir tasarımın kapsamlı bir şekilde ele alınması ve planlanması anlamına gelir. Mühendislerin ve tasarımcıların sorunlara çözüm geliştirmek için kullandıkları bir dizi adımdır. Örneğin, yazarlar kendilerini yazılı iletişim (kitap, tez vb.) yoluyla ifade etmek için kullandıkları bir dizi işlem adımı kullanırlar.

Hayal gücü, planlama, yaratım, revizyon ve paylaşım gibi yaratıcı süreçlerin farkına varmak çok önemlidir. Bu süreçler döngüsel ve kesin bir başlangıç ya da bitiş noktası olmaksızın yinelenir.



Şekil 8 – Tasarım Süreci

Tasarım Süreci Örnekleri :

- Sırt ağrısı çekenler için rahat bir sandalye tasarımı (endüstriyel tasarım)
- Kurs tasarlama (eğitim tasarımı)
- Bir evin odalarının tasarlanması (mimari çalışma)
- Bir kuruluştaki iş sürecinin tasarlanması (Yönetim)

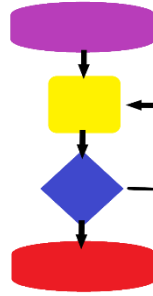
EL NOTU

Bilgi işlemsel düşünmenin beceri boyutları

"Bilgi işlemsel düşünmenin beceri boyutları", bilgi işlemsel düşünme pratiğiyle ilişkili belirli becerileri veya yetenekleri ifade eder. Bilgi işlemsel düşünme, kavramları anlamaktan daha fazlasını içerir; aynı zamanda problem çözme ve karar verme süreçlerinde bilgi işlemsel düşünmeyi etkin bir şekilde kullanmak için belirli becerilerin geliştirilmesini ve uygulanmasını gerektirir.

Algoritmik Düşünme

Algoritma, bir sorunu çözmek veya bir hedefe ulaşmak için atılan bir dizi adımdır. Bir algoritmada, her adım tam bir talimat olarak tanımlanmalıdır, böylece herkes anlayabilir ve takip edebilir. Buna dayanarak, algoritmik düşünme, algoritma oluşturmaya bağlı bir yetenekler havuzudur (Futschek, 2006).



Şekil 1 – Algoritmik Düşünme

Bir algoritma oluştururken aşağıdakilere dikkat edin.

- sorunun ne olduğunun (veya amacın ne olduğunun) farkına varmak
- sorunu veya durumu analiz etmek
- sorunu farklı açılardan görmek
- Sorunu çözmeye (veya hedefe ulaşmaya) yardımcı olabilecek farklı adımların farkında olmak
- sorunu çözmek (veya hedefe ulaşmak) için gerekli adımları oluşturmak
- algoritmayı farklı değişkenlere göre iyileştirmek

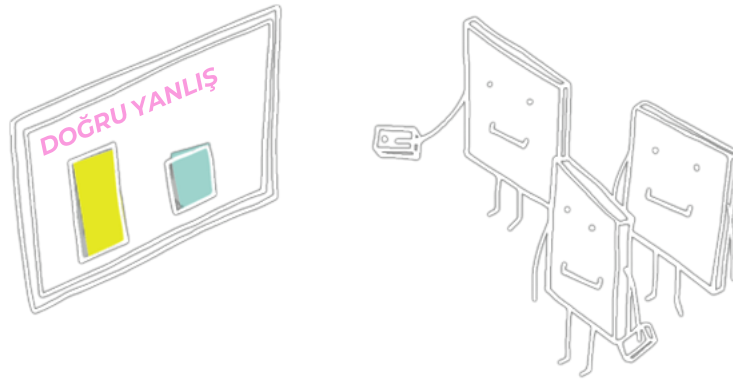
İş bulma algoritmasına bakın, tablodakilerin hangilerini görebilirsiniz?

EL NOTU

Değerlendirme

Bir algoritmayı, iyi bir çözüm sunup sunmadığını ve amaçlanan amaç için oluşturulup oluşturulmadığını görmek için değerlendiririz. Değerlendirme becerisi, algoritmaların çeşitli özelliklerinin doğru olup olmadığını (örneğin, açıkça belirtilmiş, düzgün sıralanmış, vb.) ve yeterince hızlı ve ekonomik olup olmadıklarını değerlendirme yeteneğidir.

Değerlendirme, algoritmanın çalıştığından emin olmamızı sağlar, hataları bulabilir ve çözebiliriz (Test Etme ve Hata Ayıklama).



Şekil 3 – Değerlendirme

- Başarı göstergelerinin gerçekleşme düzeyi, oluşturduğunuz sürecin veya algoritmanın verimliliğini gösterir.
- Kullanıcılardan veya uygulayıcılardan gelen geri bildirimler değerlendirme sürecine katkıda bulunur.
- Özetleyici değerlendirme bizi biçimlendirici değerlendirmeye götürür. Dolayısıyla, her iki değerlendirme süreci de dikkate alınmalıdır.

| Notlarım |
|----------|
| |
| |
| |
| |
| |

EL NOTU

Genelleştirme

Genelleme, önceki problem çözme deneyimlerinden yararlanarak yeni problemleri hızlı bir şekilde çözmenin bir yoludur. Bir problemi çözmek için oluşturulan bir algoritma, benzer problemlerin çözüm süreçlerinde uygulanabilir.

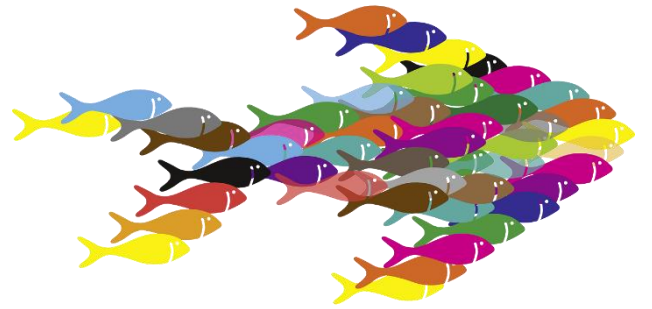
Örneğin, papatya çayı yapmak için kullanılan algoritma melisa çayı veya diğer bitki çayı türlerini yapmak için kullanılabilir. Bu, yeni bir algoritma oluşturmak yerine mevcut bir algoritmayı yeniden kullanmamızı sağlar. Ya da yeni bir algoritma oluşturmak için mevcut bir algoritmanın üzerine inşa edebiliriz. Ayrıca genelleme yoluyla daha karmaşık algoritmalar oluşturmak için algoritmaları karıştırabiliriz. Bu şekilde işimiz kolaylaşır (Yeniden Kullanma ve Karıştırma).



Şekil 4 – Genelleştirme

Günlük hayatınızda neleri geliştiriyorsunuz (benzer süreçler için aynı adımları izliyorsunuz)? Birkaç örnek verebilir misiniz?

Aşağıya yazabilirsiniz

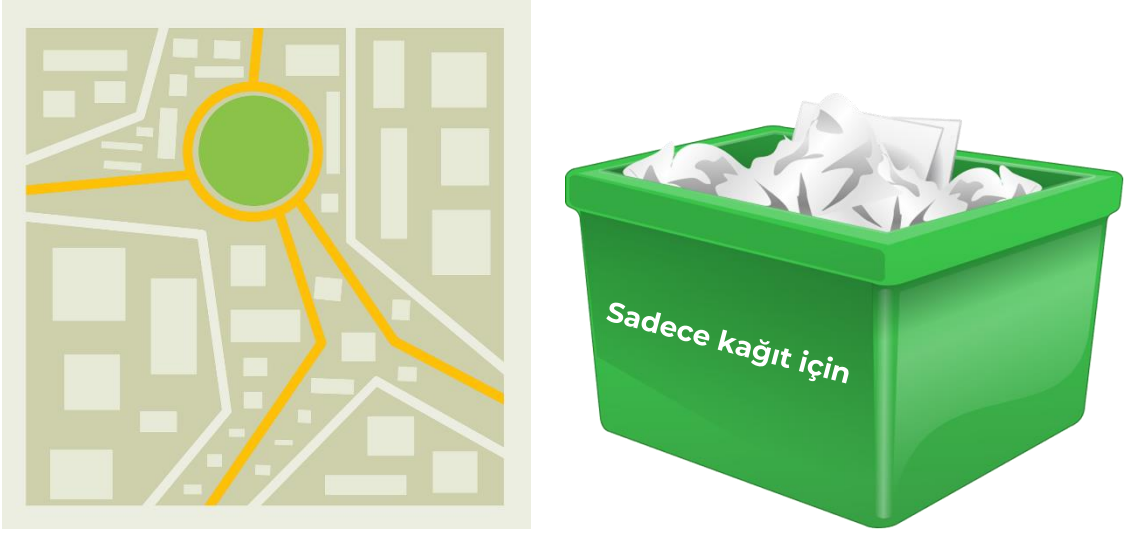


EL NOTU

Soyutlama

Soyutlama, sorunlar ve sistemler hakkında düşünmenin başka bir yoludur. Ayrıntıları gizler (veya göz ardı eder) ve gereksiz karmaşıklığı ortadan kaldırır. Soyutlama becerisi, odaklanılması gereken gerçek sorunu ortaya çıkarmak için gizlenmesi gereken doğru ayrıntıları seçmemizi sağlar. Karmaşık algoritmalar yaratmanın bir yoludur.

Bir problemin farklı temsilleri, farklı görevleri yerine getirmemizi kolaylaştırır. Problemin amacını tanımlamak için temsili seçmek gerekir. Bu beceri boyutu, bütün ve parçalar arasındaki bağlantıları keşfetme becerisini ifade eder (Soyutlama ve Modülerleştirme).



Şekil 5 – Soyutlama

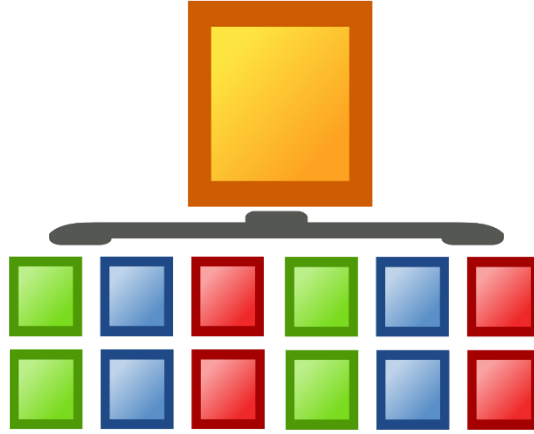
Dikkat:

- Çocuklardan bize raflardan sadece küçük boyutlu hikaye kitapları getirmelerini istersek, bu bir soyutlama mıdır?
- Ya da onlardan sadece sarı oyuncakları götürmelerini istersek, bu bir soyutlama örneği midir?
- Erken eğitimde farklı soyutlama örnekleri bulabilir ve bunların bu beceri boyutunu öğretmek için uygun olup olmadığını tartışabilirsiniz.

EL NOTU

Ayrıştırma

Karmaşık bir problemi veya yapıyı daha küçük parçalara ayırma yeteneğidir. Problemleri, algoritmaları, süreçleri, ürünleri ve sistemleri kendi parçaları açısından düşünme biçimidir. Ayrıştırılan parçalar, ayrıştırma sürecinden sonra ayrı ayrı anlaşılabilir, geliştirilebilir ve değerlendirilebilir.



Şekil 6 – Ayrıştırma

İsteğe bağlı etkinlik: [Çalışma Kağıdı 6](#)

- Ayrıştırma süreci örneklerini inceleyin.
- Kendi ayrıştırma sürecinizi oluşturun.
- Ayrıştırmayı günlük hayatımızda nasıl kullandığımızı düşünün.

| Notlarım |
|----------|
| |
| |
| |
| |
| |