

HAFTA 1

Algoritma nedir?

Bilgisayara bir iş yaptırmadan önce, o işi başka birinin de uygulayabileceği adımlarla anlatmayı öğreniyoruz.

💡 Düşün

☰ Sırala

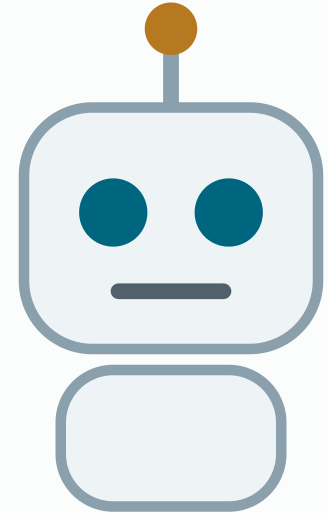
>_ İlk Python komutu

Bana bir tost yap.

Ben bir robotum. Sadece söylediğini **harfiyen** yaparım; ne demek istediğini tahmin etmem.

🗣 Bana adım adım söyleyin: tostı nasıl yapayım?

Örnek bir “hata”: **“Ekmeği al.”** → Robot paketin tamamını alır. Kaç dilim, hangi ekmek?



Bilgisayar, ne demek istediğini değil, ne yazdığını yapar.

İyi haber: Aynı adımları verersen her seferinde aynı sonucu alırsın.

Algoritma nedir?

Bir amaca ulaşmak için izlenen,
açık, sıralı ve sonlu adımlar dizisi.



Yemek tarifi

Malzemeler → adımlar →
yemek



Yol tarifi

Buradan çık → sağa dön →
vardın



Mobilya kurulumu

Parçalar → vidala → dolap

İyi bir algoritma nasıl olur?



Açık

Her adımın tek bir anlamı var.

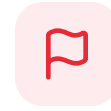
“Biraz tuz” değil,
“yarım çay kaşığı tuz”



Sıralı

Her adımın yeri belli.

Önce suyu kaynat,
sonra çayı demle.



Sonlu

Bir noktada biter.

“Bir süre bekle”
değil,
“10 dakika bekle”



Girdi ve çıktı

Neyle başladığı ve ne ürettiği belli.

Su ve çayla başla,
bir bardak çay elde et.

Hangisi belirsiz?

Her talimat için karar ver: Bir robot bunu olduğu gibi uygulayabilir mi?

A Biraz tuz ekle.

B Fırını 180 °C'ye ayarla.

C Bir süre bekle.

D Kütüphanenin 2. katındaki masaya git.

E Yakındaki durakta in.

F Suyu kaynayana kadar ısıt.

 Önce birlikte oylayalım. Sonra herkes kendi ekranında: **Problemi anla • 8 dakika**

Belirsizi açık hâle getir.

~~Biraz tuz ekle.~~

→ Yarım çay kaşığı tuz ekle.

~~Bir süre bekle.~~

→ 10 dakika bekle.

~~Yakındaki durakta in.~~

→ Belediye durağında in.

~~Ekmeği al.~~

→ Paketten iki dilim ekmek al.

❓ Belirsiz kelimeleri ara: **biraz, bir süre, yakında, yeterince, uygun şekilde**. Yerlerine sayı, birim, yer ya da bitiş koşulu koy.

Bunu yapabilmek için önce ne hazır olmalı?



Fişe takmanın başka ön koşulu yok: pişirmeden önceki herhangi bir anda olabilir.

Tek bir doğru sıra olmayabilir.

Plan A

✓ Tost hazır

- 1 **Fişe tak**
- 2 Ekmek al
- 3 Kaşar koy
- 4 Makineye koy
- 5 Pişir
- 6 Tabağa al

Plan B

✓ Tost hazır

- 1 Ekmek al
- 2 Kaşar koy
- 3 **Fişe tak**
- 4 Makineye koy
- 5 Pişir
- 6 Tabağa al

Plan C

✗ Tost soğuk

- 1 Ekmek al
- 2 Kaşar koy
- 3 Makineye koy
- 4 Pişir
- 5 **Fişe tak**
- 6 Tabağa al

Her programın üç parçası var.



Kantinden 3 simit ve 1 ayran aldın.

GİRDİ

Simit sayısı: **3**

Simit fiyatı: **10 TL**

Ayran fiyatı: **15 TL**

İŞLEM

Simitler: $3 \times 10 = 30$

Ayrani ekle: $30 + 15$

$$3 * 10 + 15$$

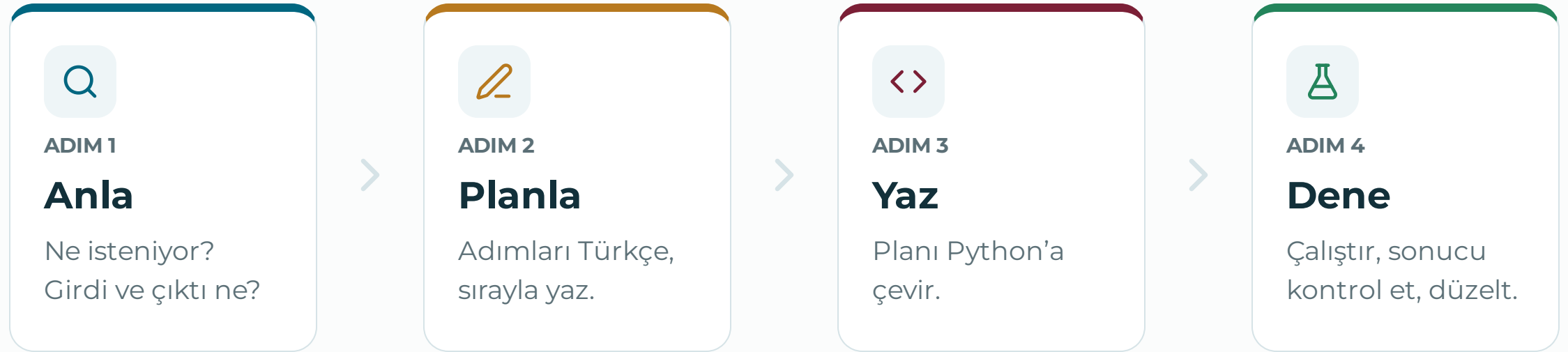
ÇIKTI

45 TL

Aynı üçlü, farklı durumlar

DURUM	GİRDİ	İŞLEM	ÇIKTI
Çay demlemek	Su, çay	Kaynat, demle	Bir bardak çay
Harita uygulaması	Başlangıç ve varış yeri	En kısa yolu bul	Yol tarifi, tahmini süre
Not ortalaması	Vize ve final notu	Ağırlıklı ortalamayı hesapla	Dönem notu
Telefon kilidi	Girilen şifre	Kayıtlı şifreyle karşılaştır	Kilit açıldı / açılmadı

Problemden programa dört adım



↺ Dene adımında bir hata çıkarsa geri dönüp planı düzeltiriz. Bu döngü normaldir.

İKİNCİ BÖLÜM

Python'la tanışalım.

Algoritmayı bilgisayarın anlayacağı dilde yazmanın ilk adımı: ekrana bir şey yazdırmak.

```
print("Merhaba, dünya!")
```

— İLK KOMUT

print() ekrana yazar.

```
print("Merhaba, dünya!")  
print("Bugün ilk dersim.")
```

EKRAN

```
Merhaba, dünya!  
Bugün ilk dersim.
```

- 1 **print** komutun adı: “yazdır”.
- 2 Yazdırılacak şey **parantezin içine** konur.
- 3 Metin **tırnak içinde** yazılır.
- 4 Her **print** yeni bir satıra yazar.

Tırnak her şeyi değiştirir.

Tırnak var → metin

```
print("2 + 3")
```

EKRAN

2 + 3

Python yazıyı olduğu gibi gösterir.

Tırnak yok → hesap

```
print(2 + 3)
```

EKRAN

5

Python önce hesaplar, sonucu gösterir.

Bu program ekrana ne yazar?

```
print("Kantin")  
print(3 * 10 + 15)  
print("TL")
```

A Kantin 45 TL (tek satır)

B Kantin / 3 * 10 + 15 / TL

C Kantin / 45 / TL (üç satır)



Önce tahmin et, sonra çalıştır. Tahminin tutmazsa nedenini bul: öğrenme tam orada olur.

Kırmızı yazı düşman değil, ipucu.

KOD

```
print("Ders başladı")  
print("Yoklama alındı")
```

EKRAN

```
SyntaxError: unterminated string literal  
  (1. satır)
```

- 1 **Hangi satır?** Mesaj 1. satırı gösteriyor.
- 2 **Ne tür hata?** `SyntaxError`: yazım hatası.
- 3 **Ne diyor?** “Metin bitmedi”: kapanış tırnağı yok.
- 4 **Düzeltil:** `print("Ders başladı")`

Kod da yukarıdan aşağıya çalışır.

× Karışık

```
print("Çayı bardağa koy")  
print("Suyu kaynat")  
print("Demliğe çay koy")
```



✓ Doğru sıra

```
print("Suyu kaynat")  
print("Demliğe çay koy")  
print("Çayı bardağa koy")
```

💡 Python satırları **yazdığın sırayla** çalıştırır. “Hangisi önce olmalı?” sorusu kodda da geçerli.

Yapay zekâ da bir talimat alıcıdır.

❓ Belirsiz istek

“Bana kod yaz.”

Ne için? Hangi dilde? Ne yapmalı? Cevap şansa kalır.

✓ Açık istek

“Python’da, 3 simit (10 TL) ve 1 ayranın (15 TL) toplamını print ile yazdıran tek satırlık bir kod yaz. Her parçasını açıkla.”

✓ Gelen cevabı **kendin çalıştır ve kontrol et**. Anlamadığın kodu kabul etme.

Özet

- 1 Algoritma: **açık, sıralı, sonlu** adımlar.
- 2 Her adımda sor: **önce ne hazır olmalı?**
- 3 Her program: **girdi → işlem → çıktı.**
- 4 **print()** yazar; tırnak = metin.
- 5 Hata mesajı bir **ipucudur.**

Çıkış sorusu

Bir arkadaşına evinden okula nasıl geleceğini anlatan 5 adımlık bir algoritma yaz.

En az bir yerde **belirsiz** bir kelime kullandıysan onu düzelt.