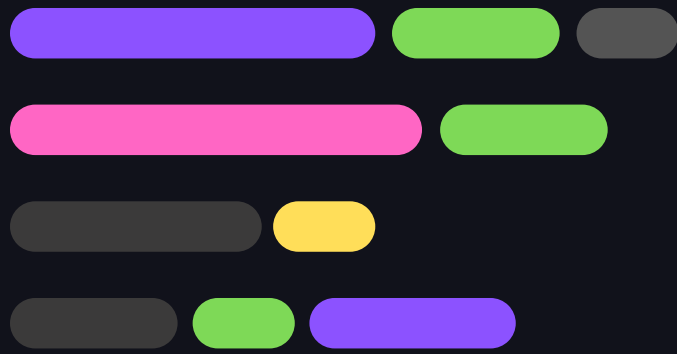




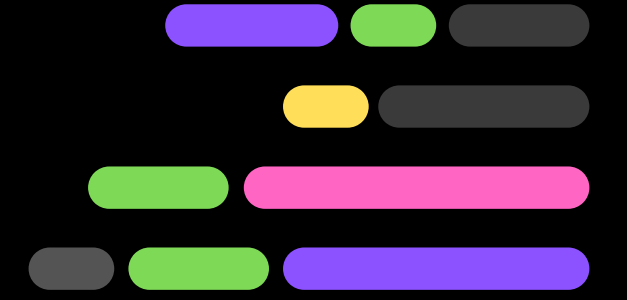
# { TEMEL PROGRAMLAMA II }

Samsun Üniversitesi  
Teknik Bilimler Meslek Yüksekokulu  
Arka-Yüz Yazılım Geliştirme Pr.





# Bu Haftanın Ders Kazanımları



01

Tarih Nesnesi

03

Proje Sunum Saati

02

Uygulama Saati

04

Proje Çalışma Saati

Tarih

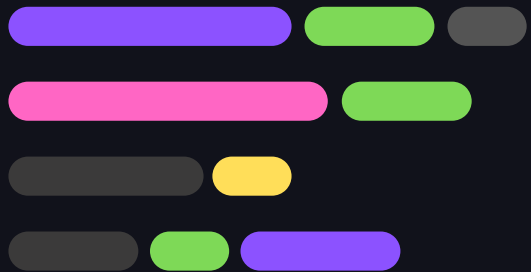
Nesnesi



01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?



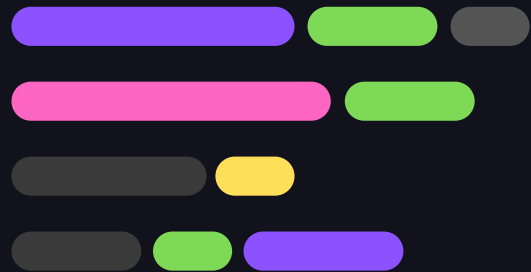


01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

Banka





01

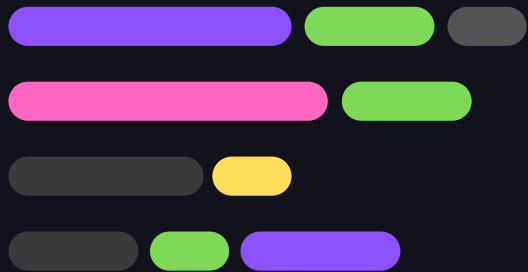
# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

Banka



İşlem Tarihi





01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

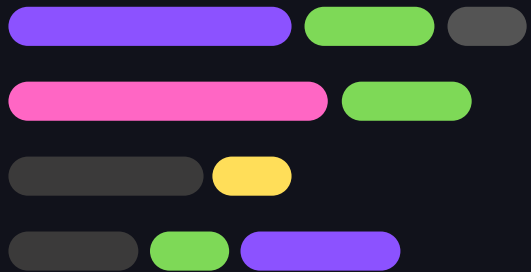
Banka

-----

İşlem Tarihi

E - Ticaret

-----





01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

Banka

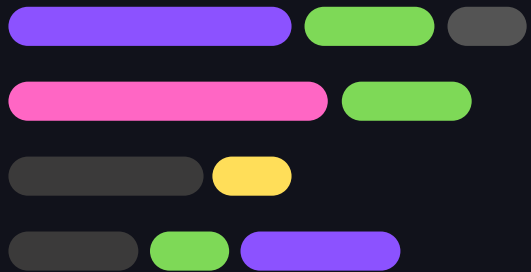
-----

İşlem Tarihi

E - Ticaret

-----

Sipariş Tarihi





01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

Banka

-----

E – Ticaret

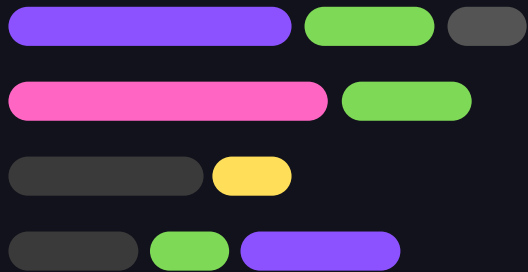
-----

Hastane

-----

İşlem Tarihi

Sipariş Tarihi





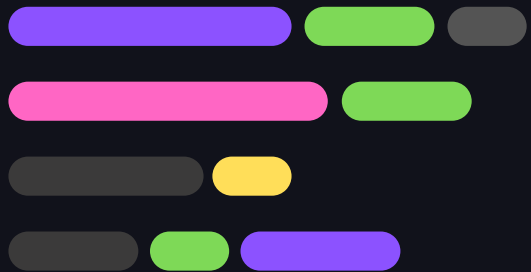
01

# Tarih Nesnesi

Günlük hayatta kullandığımız uygulama ve sistemlerde tarihi nerelerde görüyoruz?

Banka -----  
E – Ticaret -----  
Hastane -----

İşlem Tarihi  
Sipariş Tarihi  
Randevu Tarihi





01

# Tarih Nesnesi

Python dilinde zaman bilgisi tutmak için kendine ait bir veri tipi bulunmaz. Python zaman ve tarih bilgilerini bir değişkene atanmış veri değil de oluşturulmuş bir **nesne** olarak görüp işler.



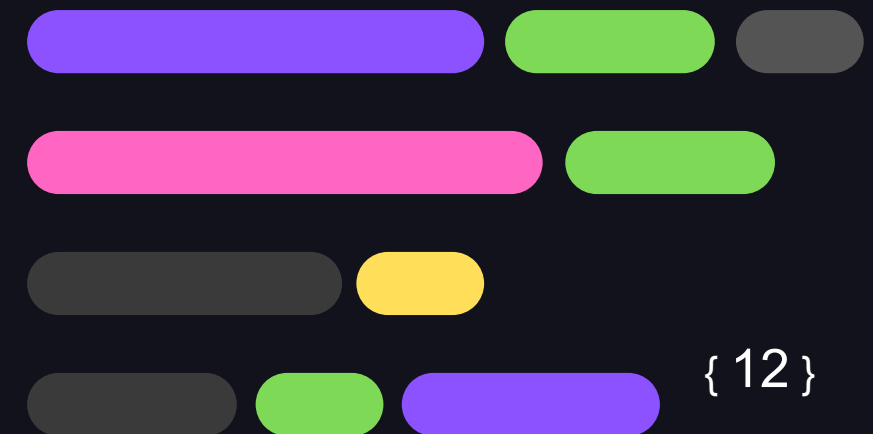


01

# Tarih Nesnesi

Bu nedenle zaman nesnesi ile çalışmak için programlama dili ile birlikte gelen **datetime** modülünü kullanmak gerekir.

**datetime** modülü; zaman, saat ve tarihlerle ilgili işlemler için çeşitli fonksiyonlar ve özellikler sağlayan sınıfları içerir.



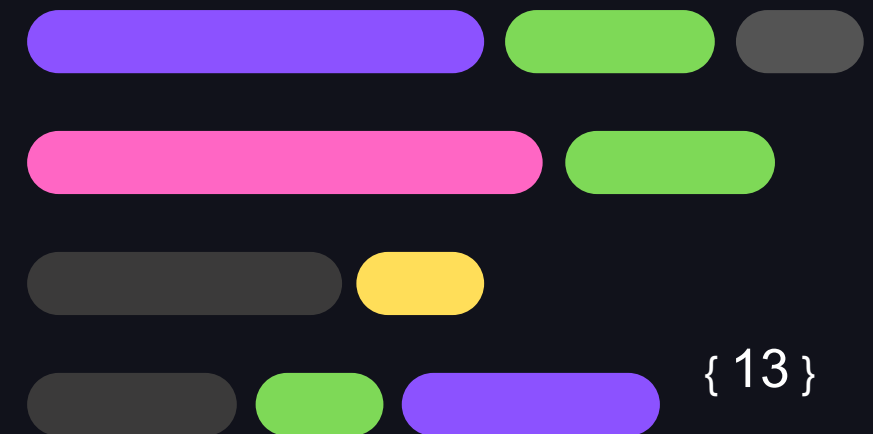


01

# Tarih Nesnesi

```
from datetime import date:
```

Tarihle ilgili nitelikleri ve fonksiyonları barındıran sınıftır. year (yıl), month (ay) ve day (gün) özelliklerini içerir.





01

# Tarih Nesnesi

```
from datetime import date:
```

Tarihle ilgili nitelikleri ve fonksiyonları barındıran sınıftır.

Yandaki kodu çalıştırınız.

```
from datetime import date  
  
d = date(2026, 3, 11)  
print(d.year, d.month, d.day)  
print(d)
```



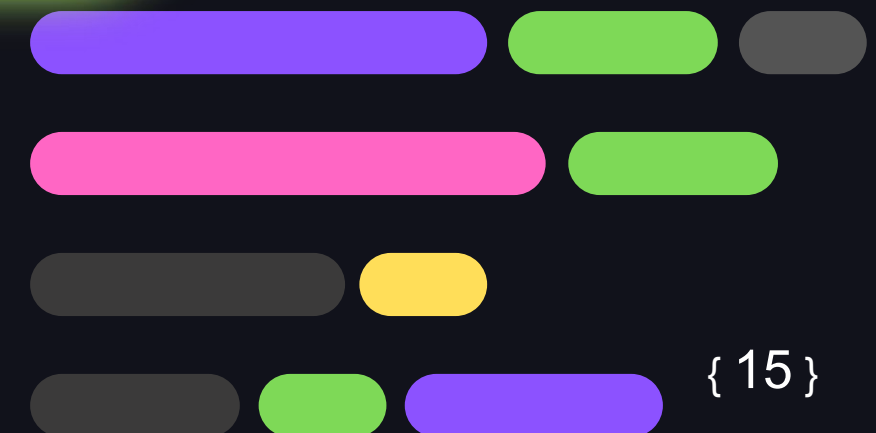
01

# Tarih Nesnesi

```
from datetime import date:
```

Doğru sıralamayı nerden bileceğiz.

```
from datetime import date  
  
d = date(2026, 3, 11)  
print(d.year, d.month, d.day)  
print(d)
```





01

# Tarih Nesnesi

```
from datetime import date:
```

İlgili sınıfın üzerine mouse ile geldiğinizde vs code size sınıf'ın hangi parametreler beklediği ile ilgili bilgi verir.

```
main.py > ...  
1 class date(  
2     year: SupportsIndex,  
3     month: SupportsIndex,  
4     day: SupportsIndex  
5 )  
6 from  
7 Constructor.  
8 d = date(2026, 3, 11)  
9 print(d.year, d.month, d.day)
```



01

# Tarih Nesnesi

```
from datetime import date:
```

İlgili sınıfın üzerine mouse ile geldiğinizde vs code size sınıf'ın hangi parametreler beklediği ile ilgili bilgi verir.

```
main.py > ...  
1 class date(  
2     year: SupportsIndex,  
3     month: SupportsIndex,  
4     day: SupportsIndex  
5 )  
6 from  
7     Constructor.  
8 d = date(2026, 3, 11)  
9 print(d.year, d.month, d.day)
```



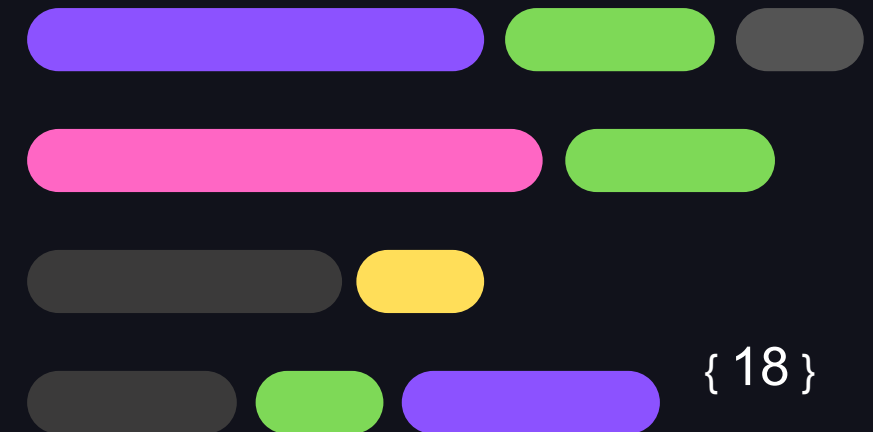
01

# Tarih Nesnesi

```
from datetime import date:
```

İlgili sınıfın üzerine mouse ile geldiğinizde vs code size sınıf'ın hangi parametreler beklediği ile ilgili bilgi verir. Yada kwargs ile göndereceğimiz değerleri istediğimiz sırada gönderebiliriz.

```
import datetime  
  
d = datetime.date(day=3, month=1, year=2026)  
print(d.year, d.month, d.day)
```





01

# Tarih Nesnesi

from datetime import **date**:

```
import datetime  
  
d = datetime.date(2026, 3, 11)  
print(d.year, d.month, d.day)
```



```
from datetime import date  
  
d = date(2026, 3, 11)  
print(d.year, d.month, d.day)
```

Modülleri 2 şekilde de kullanabilirsiniz.



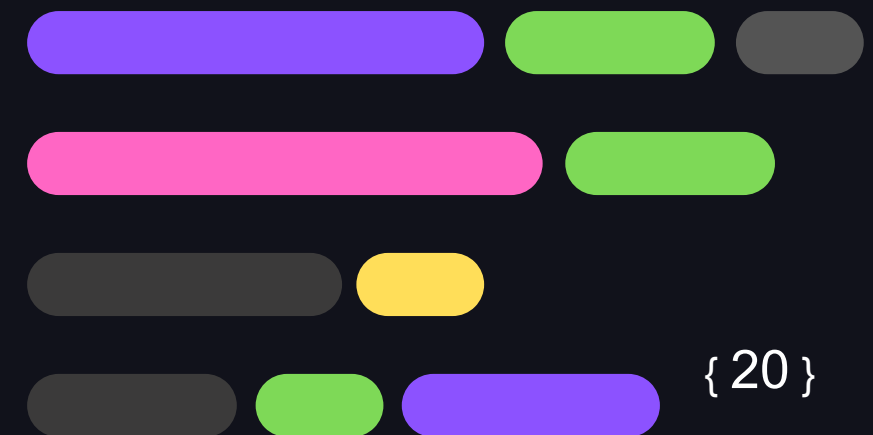
01

# Tarih Nesnesi

```
from datetime import time:
```

Zamanla ilgili nitelikleri ve fonksiyonları barındıran sınıftır.

hour (saat), minute (dakika), second (saniye) , microsecond (mikrosaniye)  
ve tzinfo (saat dilimi) özelliklerini içerir.





01

# Tarih Nesnesi

```
from datetime import time:
```

Zamanla ilgili nitelikleri ve  
fonksiyonları barındıran sınıftır.

Yandaki kodu çalıştırınız.

```
from datetime import time  
  
t = time(14, 30, 45)  
print(t.hour, t.minute, t.second)  
print(t)
```



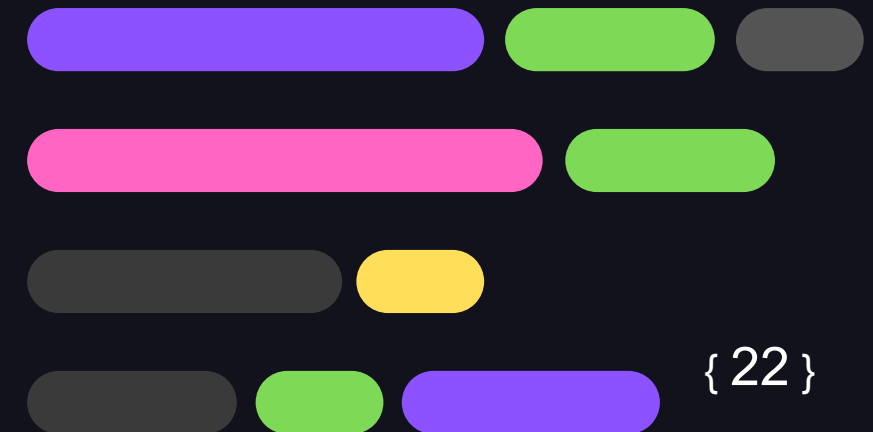
01

# Tarih Nesnesi

```
from datetime import datetime:
```

date ve time sınıflarının birleşiminden ve ilave birkaç fonksiyondan oluşur.

Uygulamalarda sıklıkla kullanılacak sınıf budur.





01

# Tarih Nesnesi

```
from datetime import datetime:
```

date ve time sınıflarının birleşiminden ve ilave birkaç fonksiyondan oluşur.

```
from datetime import datetime  
  
dt = datetime(2026, 3, 11, 14, 30)  
print(dt)
```

Yandaki kodu çalıştırınız.

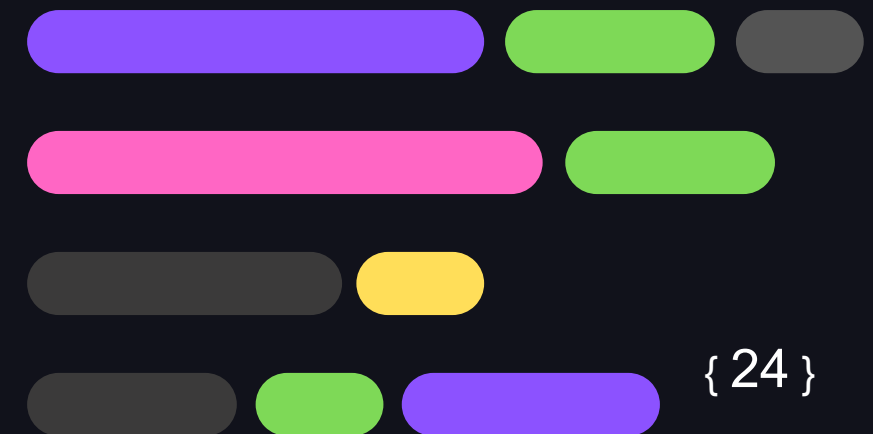


01

# Tarih Nesnesi

```
from datetime import timedelta:
```

Tarih veya saatler arasındaki zaman farkını temsil eden ve tarih üzerine zaman ekleyip çıkarmayı sağlayan sınıftır.





01

# Tarih Nesnesi

```
from datetime import timedelta:
```

Tarih veya saatler arasındaki zaman farkını temsil eden ve tarih üzerine zaman ekleyip çıkarmayı sağlayan sınıftır.

```
from datetime import datetime, timedelta  
  
now = datetime(2026, 3, 12)  
future = now + timedelta(days=5)  
  
print(future)
```

Yandaki kodu çalıştırınız.



01

# Tarih Nesnesi

```
from datetime import timedelta:
```

Yandaki kodu çalıştırınız.

```
from datetime import datetime, timedelta  
  
now = datetime(2026, 3, 12)  
future = now + timedelta(days=5, hours=3, minutes=30)  
  
print(future)
```

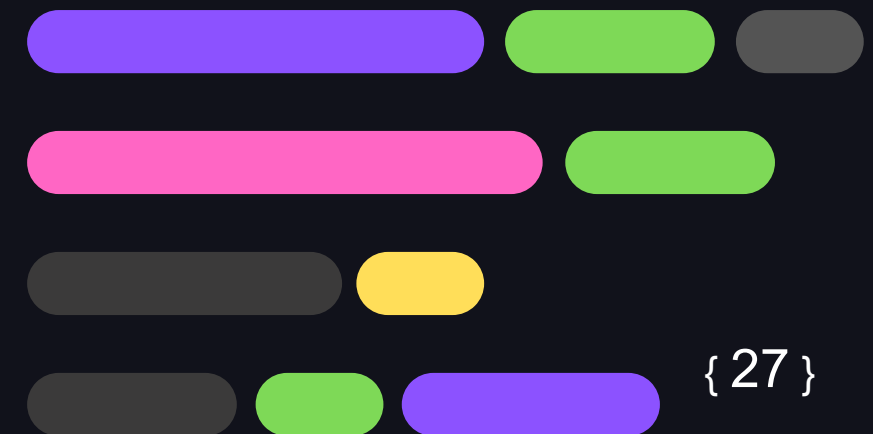


01

# Tarih Nesnesi

```
from datetime import timedelta:
```

Timedelta ile başka hangi değerleri ekleyebiliriz.





01

# Tarih Nesnesi

from datetime import **timedelta**:

Mause ile ilgili class'ın üzerine geldiğimizde, class ile ilgili başlıca detaylara ulaşabiliriz.

Class ile ilgili tüm detaylara ulaşmak istersek ctrl'ye basılı tutarak sınıf'a tıklanır.

```
class timedelta(  
    days: float = 0,  
    seconds: float = 0,  
    microseconds: float = 0,  
    milliseconds: float = 0,  
    minutes: float = 0,  
    hours: float = 0,  
    weeks: float = 0  
)
```



01

# Tarih Nesnesi

İngilizcede şuan'ın karşılığı?





01

# Tarih Nesnesi

**now()** :

Python'daki `datetime` sınıfının bir metodudur ve bilgisayarın sistem saatine göre şu anki tarih ve saat bilgisini döndürür.





01

# Tarih Nesnesi

**now()** :

Yandaki kodu çalıştırınız.

```
from datetime import datetime  
  
suan = datetime.now()  
print(suan)
```



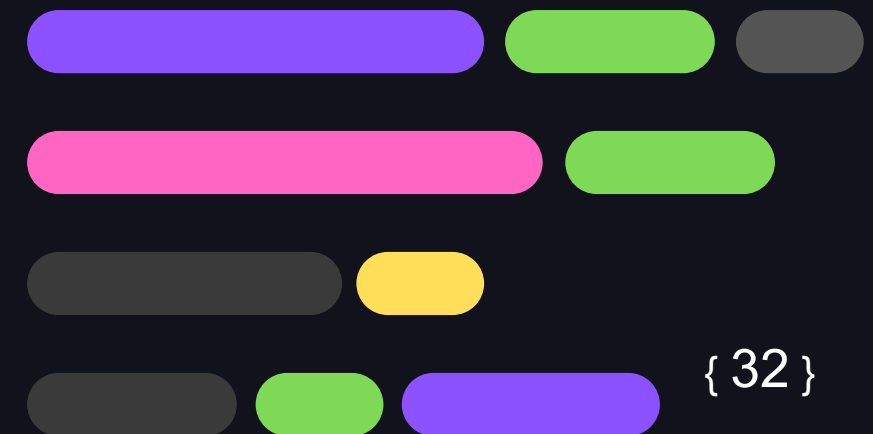
01

# Tarih Nesnesi

**now()** :

Unutulmamalı ! now metodu datetime sınıfına ait bir metottur.

Dolayısıyla date.now() gibi kullanımı hata verir.





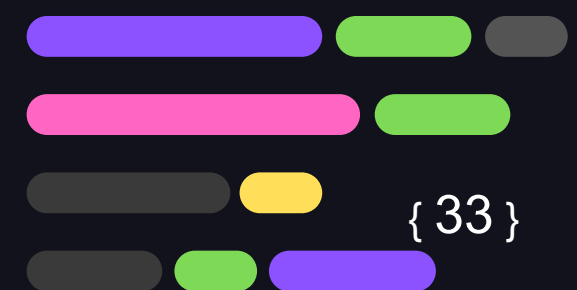
01

# Tarih Nesnesi

**today():**

now() fonksiyonu ile aynı işleve sahiptir. Bulunulan günün tarih ve saat bilgilerini verir. Eğer datetime sınıfı ile değil de date sınıfı ile beraber kullanılırsa sadece yıl, ay ve gün bilgisini verecektir.

now'ın aksine today hem datetime ile hem date ile kullanılabilir.





01

# Tarih Nesnesi

**today():**

Yandaki kodu çalıştırınız.

```
import datetime

suan1 = datetime.datetime.today()
suan2 = datetime.date.today()
print(suan1)
print(suan2)
```



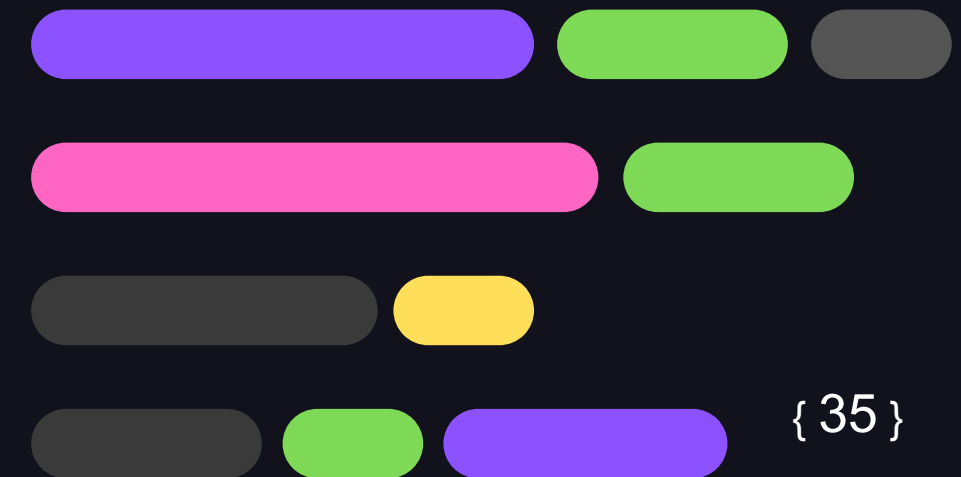
01

# Tarih Nesnesi

**weekday():**

Haftanın kaçınıcı gününde olunduğunu verir.

Pazartesi 0 - . . . - Pazar 6





01

# Tarih Nesnesi

**weekday():**

Yandaki kodu çalıştırınız.

```
from datetime import datetime
bugun = datetime.today()
print("bugun = ",bugun)
print(bugun.weekday())
print(bugun.year)
print(bugun.month)
print(bugun.day)
print(bugun.hour)
print(bugun.minute)
print(bugun.second)
```



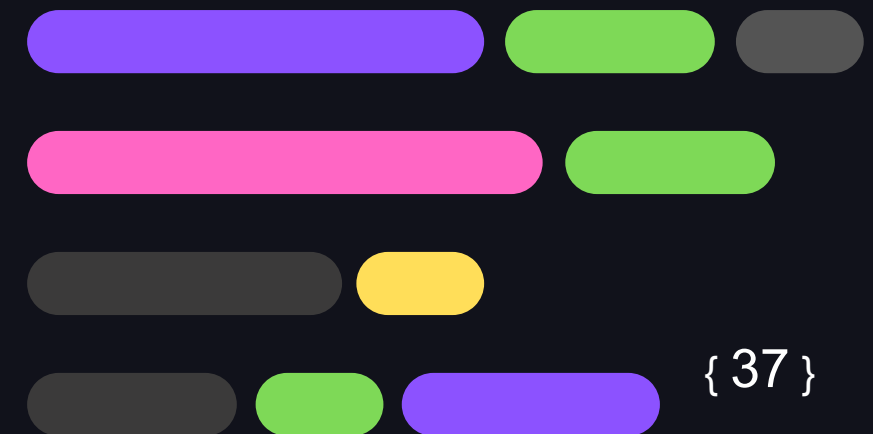
01

# Tarih Nesnesi

**Aynı sınıfa** ait iki tarih nesnesi birbirinden **çıkartılabilir**.

Örneğin `datetime - datetime` veya `date - date`.

**Farklı sınıflara** ait nesnelere (örneğin `datetime - date` gibi) arasında **çıkarma** işlemi **yapılamaz**.

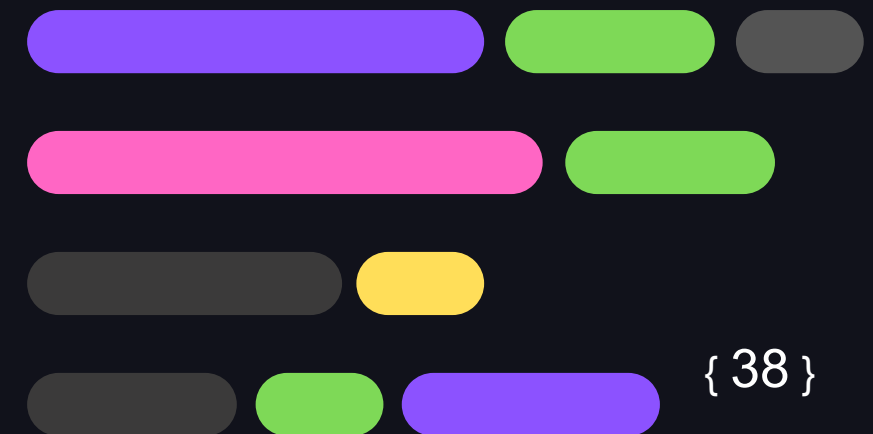




01

# Tarih Nesnesi

Time - time yapılamaz. Sebebi sizce ne olabilir.





01

# Tarih Nesnesi

Yandaki kodu çalıştırınız.

```
from datetime import datetime
date1 = datetime(1970, 10, 22)
date2 = datetime(2025, 3, 12)
yas = date2 - date1
print(yas)
```



01

# Tarih Nesnesi

İki tarih arasında aritmetiksel bir işlem yapıldığında sonuç **timedelta** yani zaman farkı nesnesi ile tutulur.

**timedelta** nesnesi ile iki tarih arasındaki farka gün, saniye veya mikrosaniye biçiminde ulaşılabilir.

```
from datetime import datetime
date1 = datetime(1970, 10, 22)
date2 = datetime(2025, 3, 12)
yas = date2 - date1
print(yas)
```

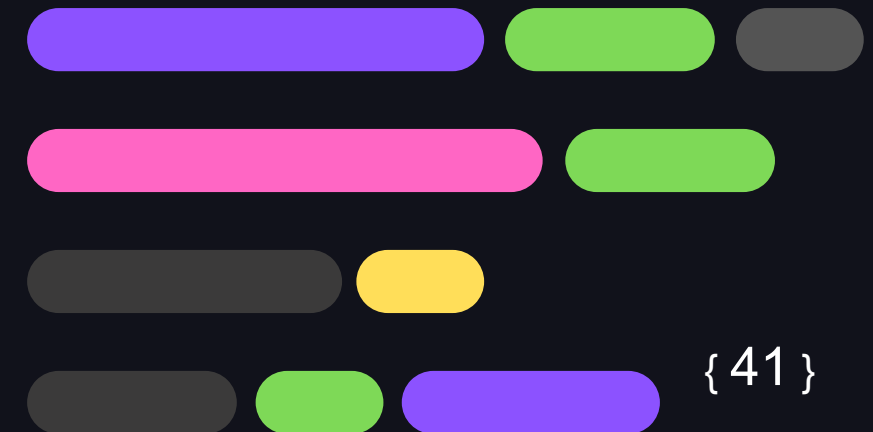


01

# Tarih Nesnesi

Kullanıcıdan doğum tarihini gün ay yıl olarak alınız ve doğduğundan beri kaç gün geçtiğini bulunuz.

```
Bu günün tarihi 2026-03-11 16:20:09.167382
Lütfen doğduğunuz yılı girin:1950
Lütfen doğduğunuz ayı girin:1
Lütfen doğduğunuz günü girin:19
27810 days, 16:20:09.167382
```





01

# Tarih Nesnesi

Doğumdan itibaren geçen gün sayısı :

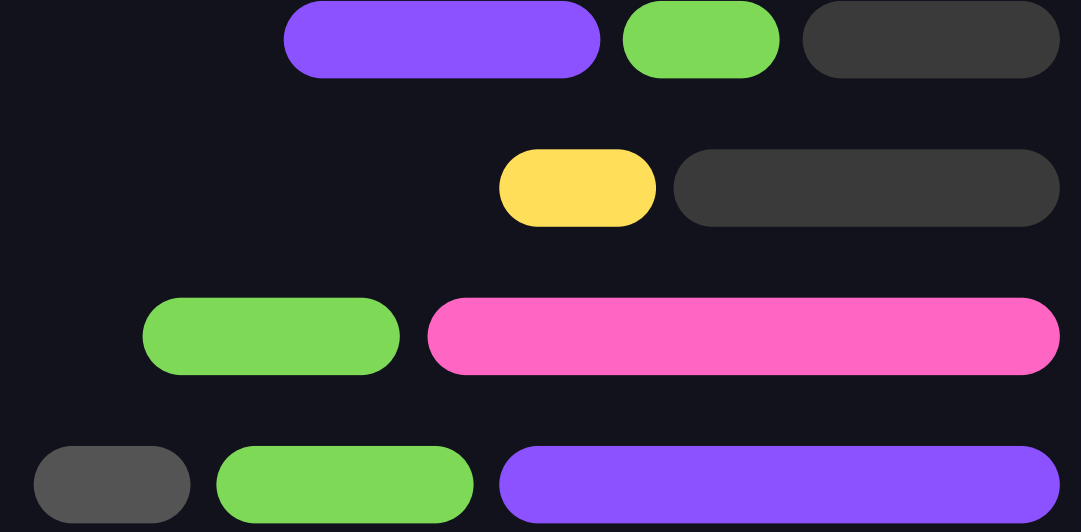
```
from datetime import datetime
bugun = datetime.today()
print("Bu günün tarihi",      bugün)
yil =  int(input("Lütfen doğduğunuz yılı girin:"))
ay =   int(input("Lütfen doğduğunuz ayı  girin:"))
gun =  int(input("Lütfen doğduğunuz günü girin:"))
dogum = datetime(year=yil,month=ay,day=gun)
yas =  bugün-dogum
print(yas)
```

Uygulama Saati



02

## Uygulama Saati



Kullanıcıdan kargonun en, boy ve yükseklik bilgileri alınacaktır.

- Program kargonun **hacmini** ( $\text{cm}^3$ ) **hesaplayacaktır**.
- Hacim  $100 \text{ cm}^3$ 'ten **büyükse** → **5 günde** teslim edilir.
- Hacim  $100 \text{ cm}^3$  veya **küçükse** → **3 günde** teslim edilir.
- Kargo çıkış tarihi **bugünün** tarihi kabul edilir.
- Eğer teslim tarihi **hafta sonuna denk gelirse**, teslim **pazartesi** yapılır.

## KARGO TESLİM HESAPLAMA

Kargonun eni (cm): 4

Kargonun boyu (cm): 6

Kargonun yüksekliği (cm): 2

Kargo hacmi:  $48 \text{ cm}^3$

Tahmini teslim tarihi: 2026-03-16 22:47:03.576444



02

## Uygulama Saati

Kargo teslim uygulaması:

```
from datetime import datetime, timedelta

print("KARGO TESLİM HESAPLAMA")

en = int(input("Kargonun eni (cm): "))
boy = int(input("Kargonun boyu (cm): "))
yukseklık = int(input("Kargonun yüksekliđi (cm): "))

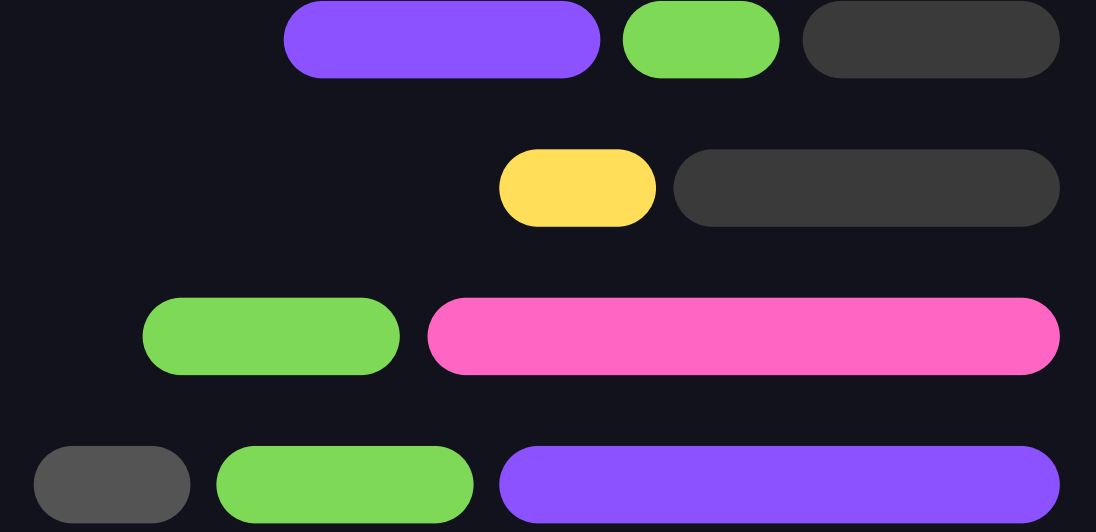
hacim = en * boy * yukseklik

bugun = datetime.now()

✓ if hacim > 100:
    teslim = bugün + timedelta(days=5)
✓ else:
    teslim = bugün + timedelta(days=3)

# hafta sonu kontrolü
✓ if teslim.weekday() == 5: # cumartesi
    teslim = teslim + timedelta(days=2)
✓ elif teslim.weekday() == 6: # pazar
    teslim = teslim + timedelta(days=1)

print("Kargo hacmi:", hacim, "cm^3")
print("Tahmini teslim tarihi:", teslim)
```





Son...