

## Ön Hazırlık: Python 3 - Syntax

- Programlama scope'ları C'de küme parantezleri {} ile belirtilirken, Python'da girinti (space yada tab) ile belirtilir.
- Girinti, bir kod satırının başındaki boşlukları ifade eder.
- Diğer programlama dillerinde koddaki girinti yalnızca okunabilirlik içindir, Python'daki girinti çok önemlidir.
- Python, bir kod bloğunu belirtmek için girinti kullanır.

```
if 5 > 2:  
    print("Five is greater than two!")
```

```
Five is greater than two!
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

7

7

## Ön Hazırlık: Python 3 - Syntax

Hata ayıklamak programlamanın en önemli parçalarından biridir.

**Soru:**

`debug_it.py` çalıştırıldığında istenen çıktıyı elde etmek üzere hataları giderin.

```
## debug_it.py
```

```
print("Day 1 - String Manipulation")  
print("String Concatenation is done with the "+" sign.")  
print('e.g. print("Hello " + "world")')  
print(("New lines can be created with a backslash and n."))
```

İstenen çıktı:

```
Day 1 - String Manipulation  
String Concatenation is done with the "+" sign.  
e.g. print("Hello " + "world")  
New lines can be created with a backslash and n.
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

8

8

## Ön Hazırlık: Python 3 - Syntax

Hata ayıklamak programlamanın en önemli parçalarından biridir.

**Soru:**

`debug_it.py` çalıştırıldığında istenen çıktıyı elde etmek üzere hataları giderin.

**Cevap:**

Girintiyi kaldır  
Fazla parantezi kaldır

```
## debug_it.py
print("Day 1 - String Manipulation")
print("String Concatenation is done with the + sign.")
print('e.g. print("Hello " + "world")')
print("New lines can be created with a backslash and n.")
```

İstenen çıktı:

```
Day 1 - String Manipulation
String Concatenation is done with the "+" sign.
e.g. print("Hello " + "world")
New lines can be created with a backslash and n.
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

9

9

## Ön Hazırlık: Python 3 – Yorum Satırları ve Blokları

- Yorum satırı # ile başlar:

```
#This is a comment.
print("Hello, World!")
```

```
Hello, World!
```

- Yorumlar bir satırın sonuna yerleştirilebilir ve Python satırın geri kalanını yok sayar:

```
print("Hello, World!") #This is a comment.
```

```
Hello, World!
```

- Python, bir değişkene atanmamış string sabitlerini yok sayacağından, kodunuza çok satırlı bir dize (üçlü tırnak) ekleyebilir ve yorumunuzu bunun içine yerleştirebilirsiniz.

```
"""
This is a comment
written in
more than just one line
"""
print("Hello, World!")
```

```
Hello, World!
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

10

10

## Ön Hazırlık: Python 3 - Syntax

- Girintiyi atlarsanız Python size bir hata verecektir:

```
if 5 > 2:
    print("Five is greater than two!")
```

```
File "demo_indentation_test.py", line 2
    print("Five is greater than two!")
    ^
IndentationError: expected an indented block
```

- Boşluk sayısına programcı karar verebilir, ancak en az bir tane olması gerekir.

```
if 5 > 2:
    print("Five is greater than two!")
if 5 > 2:
    print("Five is greater than two!")
```

```
Five is greater than two!
Five is greater than two!
```

- Aynı kod bloğunda aynı sayıda boşluk kullanmanız gerekir, aksi takdirde Python size bir hata verecektir:

```
if 5 > 2:
    print("Five is greater than two!")
    print("Five is greater than two!")
```

```
File "demo_indentation2_error.py", line 3
    print("Five is greater than two!")
    ^
IndentationError: unexpected indent
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

11

11

## Ön Hazırlık: Python 3 - Syntax

- Komut satırından girdi almak için: `input(komut)`
- Fonksiyon `komut` argümanı olmadan ya da argüman ile kullanılabilir.
- Soru:** Aşağıdaki iki örnek arasındaki fark nedir?

```
print("Enter your name:")
x = input()
print("Hello, " + x)
```

```
Enter your name:
Fatma
Hello, Fatma
```

```
x = input("Enter your name:")
print("Hello, " + x)
```

```
Enter your name:Fatma
Hello, Fatma
```

- Cevap:** Örnekte `print()` fonksiyonu sonunda `\n` karakteri varmış gibi davrandığı için kullanıcı girdisi yeni satırda alınır.

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

12

12

## Ön Hazırlık: Python 3 - Syntax

- Soru: Aşağıdaki örneklerin çıktısını bulunuz.

```
print("Hello "+ input("What is your name?") + "!!")
```

- Cevap:

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

13

13

## Ön Hazırlık: Python 3 - Syntax

- Soru: Aşağıdaki örneklerin çıktısını bulunuz.

```
print("Hello "+ input("What is your name?") + "!!")
```

- Cevap:

```
What is your name?Fatma  
Hello Fatma!!
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

14

14

## Ön Hazırlık: Python 3 - Syntax

### Alıştırma Ödevi

Kullanıcıdan adını aldıktan sonra, adındaki karakter sayısını yazdıran bir program yazın. Bir string'in uzunluğunu hesaplayan fonksiyon için Google araması yapmalısınız.

**Uyarı!** Programınız farklı girdiler için doğru çalışmalıdır.

**İpucu:** 1-Fonksiyonları diğer fonksiyonların içine koyabilirsiniz. 2- Uzunluk dışında bir şey yazdırmaya çalışmayın.

**Örnek Çıktı:**

```
What is your name?Fatma
5
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

15

15

## Ön Hazırlık: Python 3 - Syntax

### Alıştırma Ödevi (CEVAP)

```
print(len(input("What is your name?")))
```

```
What is your name?Fatma
5
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

16

16

## Ön Hazırlık: Python 3 - Değişkenler

- Python'da, değer atadığınızda değişkenler oluşturulur. Python'un değişken bildirmek için bir komutu yoktur.

```
x = 5
y = "Hello, World!"

print(x)
print(y)
```

```
5
Hello, World!
```

- Değişkenlerin belirli bir türle bildirilmesine gerek yoktur ve hatta değer atandıktan sonra türü değiştirilebilir.

```
x = 4
x = "Sally"
print(x)
```

```
Sally
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

17

17

## Ön Hazırlık: Python 3 – Değişkenler

Değişken tip dönüşümü (casting):

```
x = str(3)    # x string olarak '3' değerini alacaktır
y = int(3)    # y integer olarak 3 değerini alacaktır
z = float(3)  # z float olarak 3.0 değerini alacaktır

print(x)
print(y)
print(z)
```

```
3
3
3.0
```

Değişken tipini type() fonksiyonu ile bulabilirsiniz.

```
x = 5
y = "John"
print(type(x))
print(type(y))
```

```
<class 'int'>
<class 'str'>
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

18

18

## Ön Hazırlık: Python 3 – Değişkenler

- String değişkenleri, tek veya çift tırnak kullanılarak tanımlanabilir:

```
x = "John"
print(x)
#double quotes are the same as single
quotes:
x = 'John'
print(x)
```

```
John
John
```

- Değişken isimleri büyük/küçük harf duyarlıdır.

```
a = 4
A = "Sally"

print(a)
print(A)
```

```
4
Sally
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

19

19

## Ön Hazırlık: Python 3 – Değişkenler

### Alıştırma Ödevi

a ve b değişkenlerinde saklanan değerleri değiştiren bir program yazın.

**Uyarı!** 5-8 ve 19-21. satırlardaki kodu **değiştirmeyin**. Programınız farklı girdiler için çalışmalıdır.

Örnek Çıktı:

```
a: 123
b: 99978

a: 99978
b: 123
```

İpucu: Bir bardak süt ve bir fincan kahve düşünün. Bardak ve fincan eşit hacimde sıvı alabiliyor olsun. Sütü fincana, kahveyi bardağa nasıl transfer edersiniz?

```
5 # Don't change the code below
6 a = input("a: ")
7 b = input("b: ")
8 # Don't change the code above
9
10 #####
11 #Write your code below this line
12
13
14
15
16 #Write your code above this line
17 #####
18
19 # Don't change the code below
20 print("a: " + a)
21 print("b: " + b)
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

20

20

## Ön Hazırlık: Python 3 – Değişkenler

### Alıştırma Ödevi (Cevap)

```

5  # Don't change the code below
6  a = input("a: ")
7  b = input("b: ")
8  # Don't change the code above
9
10 #####
11 #Write your code below this line
12 print()
13 temp = a
14 a = b
15 b = temp
16 #Write your code above this line
17 #####
18
19 # Don't change the code below
20 print("a: " + a)
21 print("b: " + b)

```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

21

21

## Ön Hazırlık: Python 3 – Değişkenler

Python değişkenleri için kurallar:

- Değişken adı bir harf veya alt çizgi karakteri ile başlamalıdır
- Değişken adı bir sayı ile başlayamaz
- Değişken adı yalnızca alfanümerik karakterler ve alt çizgiler içerebilir (A-z, 0-9 ve \_)
- Değişken adları büyük/küçük harfe duyarlıdır (age, Age ve AGE üç farklı değişkendir)

```

myvar = "John"
my_var = "John"
_my_var = "John"
myVar = "John"
MYVAR = "John"
myvar2 = "John"

```

```

print(myvar)
print(my_var)
print(_my_var)
print(myVar)
print(MYVAR)
print(myvar2)

```

```

John
John
John
John
John
John
John

```

```

2myvar = "John"
my-var = "John"
my var = "John"

#This example will produce an error in the
result

```

Python Eğitimi - Dr. Fatma GÜMÜŞ

```

Traceback (most recent call last):
  File "/usr/lib/python3.7/py_compile.py", line 1, in <module>
    _optimize=optimize)
  File "<frozen importlib._bootstrap>"
  File "<frozen importlib._bootstrap>"
  File "./prog.py", line 1
    2myvar = "John"
    ^
SyntaxError: invalid syntax

```

22

22



## Ön Hazırlık: Python 3 – Değişkenler

### PEP8'e göre Değişken İsimlendirme Düzeni (Variable Naming Conventions)

Bazen PEP8 veya PEP-8 olarak da yazılan PEP 8, Python kodunun nasıl yazılacağına ilişkin yönergeler ve en iyi uygulama prensipleri sağlayan bir belgedir.

PEP, Python Geliştirme Önerisi (**Python Enhancement Proposal**) anlamına gelir ve konusuna göre numaralı belgelerden oluşur. PEP, Python için önerilen yeni özellikleri tanımlayan ve geliştirme topluluğu için Python'un tasarım ve stil gibi yönlerini gösteren bir belgedir.

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

23

23

## Ön Hazırlık: Python 3 – Değişkenler

### PEP8'e göre Değişken İsimlendirme Düzeni (Variable Naming Conventions)

PEP 8, Python kodunun okunabilirliğini geliştirmek için mevcuttur.

Python'da yeniyseniz, yazdıktan birkaç gün veya hafta sonra bir kod parçasının ne yaptığını hatırlamak zor olabilir. PEP 8'i takip ederseniz, değişkenlerinizi iyi adlandırdığınızdan emin olabilirsiniz. Yeterince boşluk eklediğinizi bileceksiniz, böylece kodunuzdaki mantıksal adımları takip etmek daha kolay olacaktır. Ayrıca kodunuzu iyi yorumlamış olacaksınız. Tüm bunlar, kodunuzun daha okunabilir ve hatırlamasının daha kolay olduğu anlamına gelir. Yeni başlayan biri olarak, PEP 8 kurallarına uymak Python öğrenmeyi çok daha keyifli bir görev haline getirebilir.

Bir geliştirici olarak görev alıyorsanız, PEP 8'i takip etmek özellikle önemlidir. Net, okunabilir kod yazmak profesyonelliği gösterir.

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

24

24

## Ön Hazırlık: Python 3 – Değişkenler

### PEP8'e göre Değişken İsimlendirme Düzeni (Variable Naming Conventions)

Python kodunu yazarken birçok şeyi adlandırmanız gerekir: değişkenler, fonksiyonlar, sınıflar, paketler vb.

Mantıklı isimler seçmek size daha sonra zaman ve enerji tasarrufu sağlayacaktır. İsimlendirmeyi inceleyerek belirli bir değişkenin, fonksiyonun veya sınıfın neyi temsil ettiğini anlayabileceksiniz. Ayrıca, hata ayıklaması zor olan hatalara neden olabilecek uygunsuz adlar kullanmaktan da kaçınacaksınız.

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

25

25

## Ön Hazırlık: Python 3 – Değişkenler

### PEP8'e göre Değişken İsimlendirme Düzeni (Variable Naming Conventions)

Not: Yazı tipine bağlı olarak 1 ve 0 ile karıştırılabileceğinden asla l, O veya I tek harfli adları kullanmayın. Örneğin;

```
O = 2 # Bu, 2'yi sıfıra yeniden atamaya çalışıyormuşsunuz gibi görünebilir.
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

26

26

# Ön Hazırlık: Python 3 – Değişkenler

## PEP8'e göre Değişken İsmilendirme Düzeni (Variable Naming Conventions)

Değişkenleri adlandırırken, x gibi basit, tek harfli küçük harf adları seçmek isteyebilirsiniz. Ancak, bir matematiksel fonksiyonun argümanı olarak x'i kullanmadığınız sürece, x'in neyi temsil ettiği net değildir. Bir kişinin adını bir string olarak kaydettiğinizi ve adını farklı biçimde biçimlendirmek için string dilimlemeyi kullanmak istediğinizi düşünelim.

Önerilmez	Önerilen
<pre>x = 'Fatma GÜMÜŞ' y, z = x.split() print(z, y, sep=', ')</pre>	<pre>name = 'Fatma GÜMÜŞ' first_name, last_name = name.split() print(last_name, first_name, sep=', ')</pre>

27

# Ön Hazırlık: Python 3 – Değişkenler

## PEP8'e göre Değişken İsmilendirme Düzeni (Variable Naming Conventions)

Değişkenler	Değişkenin görevini ifade eden küçük harf, tek harf, kelime veya kelimeler kullanın. Okunabilirliği artırmak için sözcükleri alt çizgilerle ayırın.	<b>first_name</b>
-------------	---	-------------------

28

## Ön Hazırlık: Python 3 – Değişkenler

### PEP8'e göre Diğer İsimlendirme Düzeni (Other Naming Conventions)

Kaynak: <https://realpython.com/python-pep8/#why-we-need-pep-8>, Erişim: Haziran 2022.

Type	Naming Convention	Examples
Function	Use a lowercase word or words. Separate words by underscores to improve readability.	function, my_function
Variable	Use a lowercase single letter, word, or words. Separate words with underscores to improve readability.	x, var, my_variable
Class	Start each word with a capital letter. Do not separate words with underscores. This style is called camel case.	Model, MyClass
Method	Use a lowercase word or words. Separate words with underscores to improve readability.	class_method, method
Constant	Use an uppercase single letter, word, or words. Separate words with underscores to improve readability.	CONSTANT, MY_CONSTANT, MY_LONG_CONSTANT
Module	Use a short, lowercase word or words. Separate words with underscores to improve readability.	module.py, my_module.py
Package	Use a short, lowercase word or words. Do not separate words with underscores.	package, mypackage

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

29

29

## Ön Hazırlık: Python 3 – Değişkenler

Python, bir satırda birden çok değişkene değer atamanıza izin verir:

```
x, y, z = "Orange", "Banana", "Cherry"
print(x)
print(y)
print(z)
```

Orange  
Banana  
Cherry

**Not:** Değişken sayısının değer sayısıyla eşleştiğinden emin olun, aksi takdirde hata alırsınız.

Benzer şekilde, aynı değeri bir satırda birden çok değişkene atayabilirsiniz:

```
x = y = z = "Orange"
print(x)
print(y)
print(z)
```

Orange  
Orange  
Orange

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

30

30

## Ön Hazırlık: Python 3 – Değişkenler

Bir değerler koleksiyonunuz varsa (list, tuple vb.), Python değerleri değişkenlere çıkarmanıza izin verir. Buna «unpacking» denir.

```
fruits = ["apple", "banana", "cherry"]  
x, y, z = fruits  
  
print(x)  
print(y)  
print(z)
```

```
apple  
banana  
cherry
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

31

31

## Ön Hazırlık: Python 3 – Değişkenler

`print()` ifadesi genellikle değişkenlerin çıktısı için kullanılır. Python, metni ve değişkeni birleştirmek için `+` karakterini kullanır:

```
x = "awesome"  
print("Python is " + x)
```

```
Python is awesome
```

Başka bir değişkene bir değişken eklemek için `+` karakterini de kullanabilirsiniz:

```
x = "Python is "  
y = "awesome"  
z = x + y  
print(z)
```

```
Python is awesome
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

32

32

## Ön Hazırlık: Python 3 – Değişkenler

Sayılar için + karakteri matematiksel bir operatör olarak çalışır:

```
x = 5
y = 10
print(x + y)
```

15

Bir string ve bir sayıyı + operatörüyle birleştirmeye çalışırsanız, program hata verecektir:

```
x = 5
y = "John"
print(x + y)
```

TypeError: unsupported operand type(s) for +:

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

33

33

## Ön Hazırlık: Python 3 – Değişkenler

### Alıştırma Ödevi

Proje grubu için isim üretme programı `grup_turet.py` dosyası aşağıdaki gereksinimleri sağlamalıdır.

- #1. Programınız için bir selamlama oluşturun.
- #2. Kullanıcıya bir şehir adı sorun.
- #3. Kullanıcıya bir evcil hayvan adı sorun.
- #4. Şehir ve evcil hayvanın adını birleştirin ve onlara grup adlarını gösterin.
- #5. Kullanıcıdan girdi alırken, giriş imlecinin yeni bir satırda gösterildiğinden emin olun.

Örnek Çıktı:

```
Welcome to the Project Group Name Generator.
Which city you like visiting?
İstanbul
What is a pet name you like?
Boncuk
Your project group name could be İstanbul Boncuk
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

34

34

## Ön Hazırlık: Python 3 – Veri Tipleri

- Değişkenler farklı türlerde veri depolayabilir ve farklı türler farklı işlemler yapabilir.
- Python, varsayılan olarak aşağıdaki veri türlerine sahiptir:

Text Type: `str`

Numeric Types: `int`, `float`, `complex`

Sequence Types: `list`, `tuple`, `range`

Mapping Type: `dict`

Set Types: `set`, `frozenset`

Boolean Type: `bool`

Binary Types: `bytes`, `bytearray`, `memoryview`

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

35

35

## Ön Hazırlık: Python 3 – Veri Tipleri: String

### String

```
print(len("Hello"))
print("Hello"[0])
print("Hello"[-1])
```

```
5
H
o
```

	H	e	l	l	o
İndis	0	1	2	3	4
İndis	-5	-4	-3	-2	-1

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

36

36

## Ön Hazırlık: Python 3 – Veri Tipleri: String

- Diğer birçok programlama dilinde olduğu gibi, Python'daki string'ler de unicode karakterleri temsil eden byte dizileridir.
- Ancak Python'da bir karakter veri türü yoktur, tek bir karakter yalnızca 1 uzunluğunda bir string'tir.
- String öğelerine erişmek için köşeli parantezler kullanılabilir.

```
a = "Hello, World!"  
print(a[1])
```



- Bir dizinin uzunluğunu almak için `len()` fonksiyonunu kullanın.

```
a = "Hello, World!"  
print(len(a))
```



27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

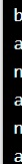
37

37

## Ön Hazırlık: Python 3 – Veri Tipleri: String

- String'ler dizi olduğundan, bir dizideki karakterler arasında `for` ifadesi ile döngü yapabiliriz.

```
for x in "banana":  
    print(x)
```



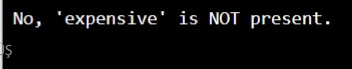
- Bir dizide belirli bir kelime öbeği veya karakterin olup olmadığını kontrol etmek için `in` anahtar sözcüğünü kullanabiliriz.

```
txt = "The best things in life are free!"  
print("free" in txt)
```



- Bir dizide belirli bir kelime öbeğinin veya karakterin OLMADIĞINI kontrol etmek için, `not in` anahtar sözcüğünü kullanabiliriz.

```
txt = "The best things in life are free!"  
if "expensive" not in txt:  
    print("No, 'expensive' is NOT present.")
```



27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

38

38




## Ön Hazırlık: Python 3 – Veri Tipleri: String

- Dilimleme (slicing) sözdizimini kullanarak bir dizi karakter döndürebilirsiniz. String'in bir bölümünü döndürmek için başlangıç dizinini ve bitiş dizinini iki nokta üst üste ile ayırarak belirtin.

- Örnek: 2. konumdan 5. konuma kadar olan karakterleri alın (dahil değildir): 

```
b = "Hello, World!"  
print(b[2:5])
```

 **Not: İlk karakterin indeksi 0'dır.**

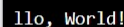
- Örnek: Başlangıç dizinini boş bırakarak, aralık ilk karakterden başlar:

```
b = "Hello, World!"  
print(b[:5])
```



- Örnek: Bitiş dizinini boş bırakarak, aralık sonuna kadar gidecektir:

```
b = "Hello, World!"  
print(b[2:])
```



27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

39

39

## Ön Hazırlık: Python 3 – Veri Tipleri: String

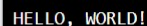
- String'in sonundan dilimi başlatmak için negatif indeksleri kullanın.
- Örnek: "o" dan (konum -5), "d" ye kadar (konum -2). Ancak "d" dahil değil:

```
b = "Hello, World!"  
print(b[-5:-2])
```



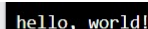
- upper()** metodu, string'i büyük harfle döndürür:

```
a = "Hello, World!"  
print(a.upper())
```



- lower()** metodu, string'i küçük harfle döndürür:

```
a = "Hello, World!"  
print(a.lower())
```



27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

40

40

## Ön Hazırlık: Python 3 – Veri Tipleri: String

- Boşluk (whitespace), asıl metinden önceki ve/veya sonraki boşluktur ve çoğu zaman bu boşluğu kaldırmak istersiniz. `strip()` metodu, başlangıçtaki veya sondaki tüm boşlukları kaldırır:

```
a = "    Hello, World! "
print(a.strip())
```

```
Hello, World!
```

- `replace()` metodu, bir string'i başka bir string ile değiştirir:

```
a = "Hello, World!"
print(a.replace("H", "J"))
```

```
Jello, World!
```

- `split()` metodu, belirtilen ayırıcı arasındaki metnin liste öğeleri haline geldiği bir liste döndürür. `split()` yöntemi, ayırıcının örneklerini bulursa dizeyi alt dizelere böler:

```
a = "Hello, World!"
b = a.split(",")
print(b)
```

```
['Hello', ' World!']
```

- Diğer string metotları için bkz.: [https://www.w3schools.com/python/python\\_ref\\_string.asp](https://www.w3schools.com/python/python_ref_string.asp)

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

41

41

## Ön Hazırlık: Python 3 – Veri Tipleri: String

- İki string'i birleştirmek için `+` operatörünü kullanabilirsiniz.

```
a = "Hello"
b = "World"
c = a + b
print(c)
```

```
HelloWorld
```

- Aralarına boşluk eklemek için bir `" "` ekleyin:

```
a = "Hello"
b = "World"
c = a + " " + b
print(c)
```

```
Hello World
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

42

42

## Ön Hazırlık: Python 3 – String

- Python Değişkenleri bölümünde öğrendiğimiz gibi, string'leri ve sayıları birleştiremeyiz. Ancak, `format()` metodunu kullanarak string'leri ve sayıları birleştirebiliriz.

```
age = 36
txt = "My name is John, and I am {}"
print(txt.format(age))
```

```
My name is John, and I am 36
```

- `format()` metodu sınırsız sayıda argüman alabilir ve ilgili yer tutuculara `{}` yerleştirilir:

```
quantity = 3
itemno = 567
price = 49.95
myorder = "I want {} pieces of item {} for {} dollars."
print(myorder.format(quantity, itemno, price))
```

```
I want 3 pieces of item 567 for 49.95 dollars.
```

- Bağımsız değişkenlerin doğru yer tutuculara yerleştirildiğinden emin olmak için `{0}` dizin numaralarını kullanabilirsiniz:

```
quantity = 3
itemno = 567
price = 49.95
myorder = "I want to pay {2} dollars for {0} pieces of item {1}."
print(myorder.format(quantity, itemno, price))
```

```
I want to pay 49.95 dollars for 3 pieces of item 567.
```

Bkz. [https://www.w3schools.com/python/python\\_string\\_formatting.asp](https://www.w3schools.com/python/python_string_formatting.asp)

43

## Ön Hazırlık: Python 3 – String

- Bir dizide geçersiz karakterler eklemek için bir **escape karakteri** kullanın.
- Bir escape karakteri, ters eğik çizgi `\` ve ardından eklemek istediğiniz karakterdir.
- Örnek: Çift tırnak içine alınmış bir string içinde çift tırnak kullanırsanız bir hata alırsınız:

```
txt = "We are the so-called \"Vikings\" from the north."
```

```
File "demo_string_escape_error.py", line
txt = "We are the so-called \"Vikings\" f
^
SyntaxError: invalid syntax
```

- Bu sorunu çözmek için escape karakteri `\"` kullanın:

```
txt = "We are the so-called \"Vikings\" from the north."
print(txt)
```

```
We are the so-called "Vikings" from the north.
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

44

44

## Ön Hazırlık: Python 3 – String

- Python'da kullanılan diğer escape karakterleri:

\'	Single Quote
\\	Backslash
\n	New Line
\r	Carriage Return
\t	Tab
\b	Backspace
\f	Form Feed
\ooo	Octal value
\xhh	Hex value

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

45

45

## Ön Hazırlık: Python 3 – Veri Tipleri: Basit (primitive) Veri Tipleri

### String

Aşağıdaki kod parçalarında satırları ayrı ayrı çalıştırınız. Çıktıları inceleyiniz.

```
print(len("1357")) 4
print("1357"[0]) 1
print("1357"[-1]) 7
```

```
print(len(1357)) TypeError: object of type 'int' has no len()
print(1357[0]) TypeError: 'int' object is not subscriptable
print(1357[-1]) TypeError: 'int' object is not subscriptable
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

46

46

## Ön Hazırlık: Python 3 – Veri Tipleri: Basit (primitive) Veri Tipleri

### int

Tamsayı ifade eder.

Aşağıdaki kod parçalarında satırları ayrı ayrı çalıştırınız. Çıktıları inceleyiniz.

```
print(123)
```

123

```
print(123 + 987)
```

1110

```
print("123" + "987")
```

123987

```
print(-987)
```

-987

```
print(123 + -987)
```

-864

```
print(1_000_000)
```

1000000

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

47

47

## Ön Hazırlık: Python 3 – Veri Tipleri

**float:** Ondalık değer ifade eder.

Aşağıdaki kod parçalarında satırları ayrı ayrı çalıştırınız. Çıktıları inceleyiniz.

```
print(3.14159)
```

3.14159

```
print(-3.14159 + 1)
```

-2.14159

Bkz. Floating Point Arithmetics, <https://docs.python.org/3/tutorial/floatpoint.html> , Erişim: Haziran 2022.

### Boolean

Boolean değer ifade eder. Yalnız ilk harfler büyük olarak yazılır.

```
True
```

```
False
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

48

48

## Ön Hazırlık: Python 3 – Veri Tipleri: Boolean

- Programlamada genellikle bir ifadenin **True** yada **False** olduğunu bilmeniz gerekir.
- Python'da herhangi bir ifadeyi değerlendirebilir ve True veya False olmak üzere iki yanıtta birini alabilirsiniz.
- İki değeri karşılaştırdığınızda, ifade değerlendirilir ve Python Boolean yanıt döndürür:

```
print(10 > 9)
print(10 == 9)
print(10 < 9)
```

```
True
False
False
```

```
a = 200
b = 33

if b > a:
    print("b is greater than a")
else:
    print("b is not greater than a")
```

```
b is not greater than a
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

49

49

## Ön Hazırlık: Python 3 – Veri Tipleri: Boolean

- **bool()** metodu, herhangi bir değeri değerlendirmenize ve karşılığında size True veya False dönmeye olanak tanır. `print(bool("Hello"))`  
`print(bool(15))`

```
True
True
```

- Bir tür içeriğe sahipse, hemen hemen her değer True olarak değerlendirilir.
  - Boş dizeler dışında herhangi bir dize True'dur. `print(bool("abc"))`
  - 0 dışında herhangi bir sayı True'dur. `print(bool(123))`
  - Boş olanlar dışında herhangi bir **list**, **tuple**, **set** ve **dictionary** True'dur. `print(bool(["apple", "cherry", "banana"]))`
- Aslında **()**, **[]**, **{}**, **""**, **0** sayısı ve **None** değeri gibi boş değerler dışında False olarak değerlendirilen çok fazla değer yoktur. Ve elbette False değeri False olarak değerlendirilir.

```
print(bool(False))
print(bool(None))
print(bool(0))
print(bool(""))
print(bool(()))
print(bool([]))
print(bool({}))
```

```
False
False
False
False
False
False
False
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

50

50

## Ön Hazırlık: Python 3 – Veri Tipleri

- `type()` fonksiyonunu kullanarak herhangi bir nesnenin veri türünü alabilirsiniz:

```
x = 5
print(type(x))
```

```
<class 'int'>
```

- Python'da, bir değişkene değer atadığınızda veri türü ayarlanır:

ÖRNEK	VERİ TİPİ
x = "Hello World"	str
x = 20	int
x = 20.5	float
x = 1j	complex
x = ["apple", "banana", "cherry"]	list
x = ("apple", "banana", "cherry")	tuple
x = range(6)	range
x = {"name" : "John", "age" : 36}	dict
x = {"apple", "banana", "cherry"}	set
x = frozenset({"apple", "banana", "cherry"})	frozenset
x = True	bool
x = b"Hello"	bytes
x = bytearray(5)	bytearray
x = memoryview(bytes(5))	memoryview

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

51

51

## Ön Hazırlık: Python 3 – Veri Tipleri

- Veri türünü kendiniz belirtmek için, constructor fonksiyonlar:

x = str("Hello World")	str
x = int(20)	int
x = float(20.5)	float
x = complex(1j)	complex
x = list(("apple", "banana", "cherry"))	list
x = tuple(("apple", "banana", "cherry"))	tuple
x = range(6)	range
x = dict(name="John", age=36)	dict
x = set(("apple", "banana", "cherry"))	set
x = frozenset(("apple", "banana", "cherry"))	frozenset
x = bool(5)	bool
x = bytes(5)	bytes
x = bytearray(5)	bytearray
x = memoryview(bytes(5))	memoryview

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

52

52

## Ön Hazırlık: Python 3 – Veri Tipleri

- Python'da üç sayısal tür vardır:

- `int`
- `float`
- `complex`

- Değer atadığınızda, sayısal türden değişkenler oluşturulur:

```
x = 1
y = 2.8
z = 1j

print(type(x))
print(type(y))
print(type(z))
```

```
<class 'int'>
<class 'float'>
<class 'complex'>
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

53

53

## Ön Hazırlık: Python 3 – Veri Tipleri

- `int`, pozitif veya negatif, ondalıksız, sınırsız uzunlukta bir tam sayıdır.
- `float`, bir veya daha fazla ondalık basamak içeren pozitif veya negatif bir sayıdır.
- `float`, 10'un kuvvetini belirtmek için "e" ile gösterilen bilimsel sayılar da kullanılabilir:

```
x = 35e3
y = 12E4
z = -87.7e100
```

```
print(type(x))
print(type(y))
print(type(z))
```

```
<class 'float'>
<class 'float'>
<class 'float'>
```

- Karmaşık sayılar sanal kısım "j" olarak yazılır:

```
x = 3+5j
y = 5j
z = -5j
```

```
print(type(x))
print(type(y))
print(type(z))
```

```
<class 'complex'>
<class 'complex'>
<class 'complex'>
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

54

54



## Ön Hazırlık: Python 3 – Veri Tipleri

- `int()`, `float()` ve `complex()` metotlarıyla bir tipten diğerine dönüştürebilirsiniz:

```
#int -> float
x = float(1)

#float -> int
y = int(2.8)

#int -> complex
z = complex(x)

print(x)
print(y)
print(z)

print(type(x))
print(type(y))
print(type(z))
```

```
1.0
2
(1+0j)
<class 'float'>
<class 'int'>
<class 'complex'>
```

27.06.2022

**Not: Karmaşık sayıları başka bir sayı türüne dönüştüremezsiniz.**

55

55

## Ön Hazırlık: Python 3 – Veri Tipleri

### Alıştırma Ödevi

Aşağıdaki kod parçasında hataları gideriniz.

```
num_char = len(input("What is your name"))
print("Your name has " + num_char + "characters!")
```

### Elde edilen çıktı

```
What is your nameFatma
```

```
TypeError: can only concatenate str (not "int") to str
```

### İstenen çıktı

```
What is your nameFatma
```

```
Your name has 5characters!
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

56

56

## Ön Hazırlık: Python 3 – Veri Tipleri

### Alıştırma Ödevi

Cevap:

```
num_char = len(input("What is your name"))  
print("Your name has " + str(num_char) + "characters!")
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

57

57

## Ön Hazırlık: Python 3 – Veri Tipleri

### Tip Dönüşümü

Aşağıdaki kod parçalarında satırları ayrı ayrı çalıştırınız. Çıktıları inceleyiniz.

```
a = 123  
print(type(a))  
  
a = str(123)  
print(type(a))  
  
a = float(123)  
print(type(a))  
  
print(123 + float("987.5"))  
  
print(str(123) + str(987.5))
```

```
<class 'int'>  
<class 'str'>  
<class 'float'>  
1110.5  
123987.5
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

58

58

## Ön Hazırlık: Python 3 – Veri Tipleri

### Alıştırma Ödevi

İki basamaklı bir sayının rakamlarını toplayan bir program yazın.

Örneğin, girdi 35 ise, çıktı  $3 + 5 = 8$  olmalıdır

**Uyarı!** 1-3. satırlardaki kodu değiştirmeyin. Programınız farklı girdiler için çalışmalıdır.

**İpucu:**

**two\_digit\_number** veri türünü bulmaya çalışın.

Abonelik hakkında öğrendiklerinizi bir düşünün.

Tip dönüştürmeyi düşünün.

```
1 # Don't change the code below
2 two_digit_number = input("Type a two digit positive integer: ")
3 # Don't change the code above
4
5 #####
6 #Write your code below this line
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

59

59

## Ön Hazırlık: Python 3 – Veri Tipleri

### Alıştırma Ödevi (Çözüm)

```
1 # Don't change the code below
2 two_digit_number = input("Type a two digit positive integer: ")
3 # Don't change the code above
4
5 #####
6 #Write your code below this line
7 print(type(two_digit_number))
8 print(int(two_digit_number[0])+int(two_digit_number[1]))
```

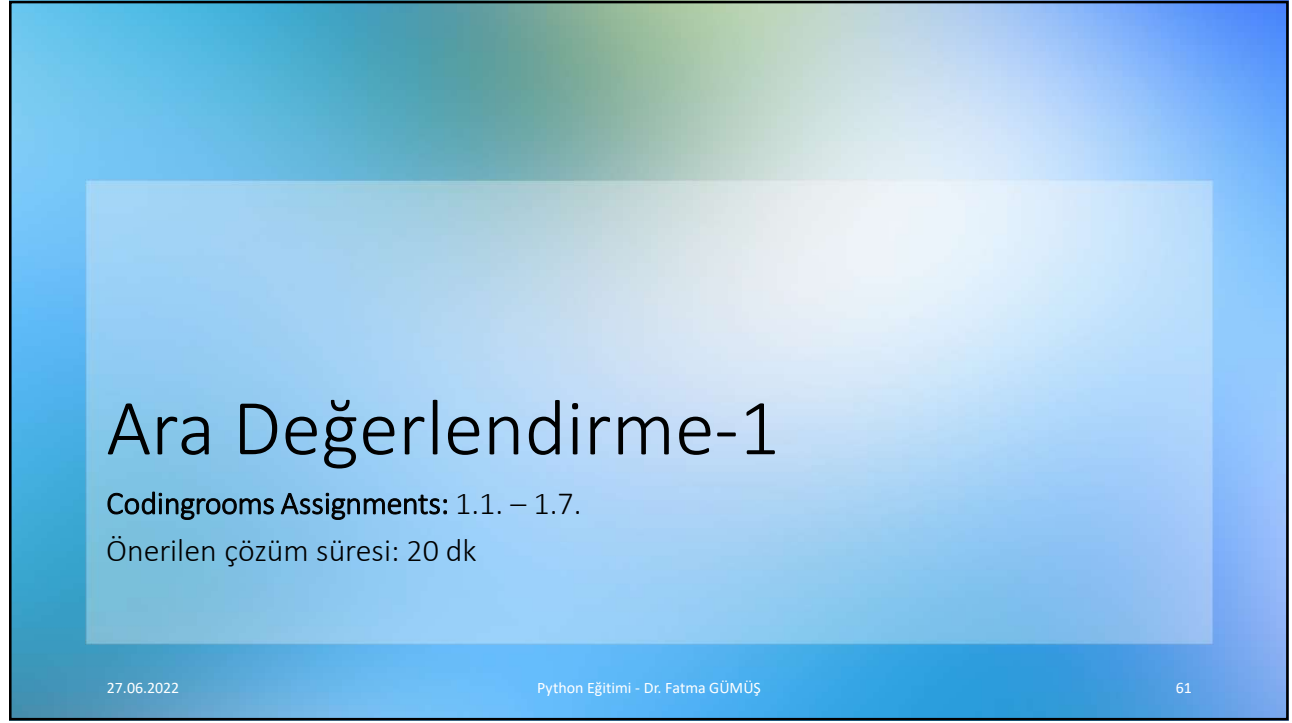
```
Type a two digit positive integer: 35
<class 'str'>
8
```

27.06.2022

Python Eğitimi - Dr. Fatma GÜMÜŞ

60

60



# Ara Değerlendirme-1

Codingrooms Assignments: 1.1. – 1.7.

Önerilen çözüm süresi: 20 dk

27.06.2022 Python Eğitimi - Dr. Fatma GÜMÜŞ 61

61