

Nesneye Yönelimli Programlama – BLM 205

Hafta 3: Python Temelleri II



Fenerbahçe Üniversitesi

Öğretim Elemanları

Öğretim Üyesi: Dr. Vecdi Emre Levent

Ofis: 311

Email: emre.levent@fbu.edu.tr

Asistan: Arş. Gör. Uğur Özbalkan

Ofis: 307

Email: ugur.ozbalkan@fbu.edu.tr

Asistan: Arş. Gör. Ecenur Alioğulları

Ofis: 307

Email: ecenur.aliogullari@fbu.edu.tr

Ders Planı

- Python Temelleri II
 - String'ler
 - Matematik Operasyonları
 - JSON Operasyonları
 - Tarih Operasyonları
 - Scope
 - İstisna İdaresi
 - Random sayı üretimi
 - Moduller

Python Temelleri II

String'ler

Metnin içerisinde özel karakterler kullanımı

<code>\"</code>	" karakteri
<code>\'</code>	' karakteri
<code>\\</code>	\ karakteri
<code>\n</code>	Yeni satır
<code>\t</code>	Tab
<code>\b</code>	Sağdaki bir karakteri siler
<code>\ooo</code>	Octal değer (<code>\110 = H</code>)
<code>\xhh</code>	Hex değeri (<code>\x65 = A</code>)

Python Temelleri II

String'ler

Stringler için kullanılabilen tüm fonksiyonlar

Fonksiyon	Açıklama
capitalize()	İlk karakteri büyük harf yapar
casefold()	String'i küçük harflere dönüştürür
center()	Aldığı integer argüman ile, metnin sağ ve soluna boşluklar koyar
count()	Argüman olarak verilen string'i, ana string'in içerisinde kaç defa geçtiğini döndürür
encode()	Metni kodlayarak (örneğin UTF-8) geri döndürür
endswith()	String verilen argümandaki string ile bitiyorsa geriye doğru döndürür
expandtabs()	String'in içerisindeki tab'ların kaç boşluktan oluşacağını ayarlar
find()	Verilen argümanı, string'in içerisinde arar, bulunan pozisyonun index'ini döndürür.
format()	Verilen argümanları, string'in içerisine yerleştirir

Python Temelleri II

String'ler

Stringler için kullanılabilen diğer fonksiyonlar

Fonksiyon	Açıklama
index()	Find fonksiyonu gibi çalışır. Tek farkı bulunamadığı zaman, exception yaratır.
isalnum()	Tüm karakterler alfabetik ve sayıdan oluşuyorsa, doğru döndürür
isalpha()	Tüm karakterler alfabetik ise, doğru döndürür
isdecimal()	String, ondalıklı sayı ise, doğru döndürür
isdigit()	Tüm karakterler sayı ise, doğru döndürür
isidentifier()	Identifier, yani bir değişken ismi tanımına uyup uymadığı kontrolü yapılır.
islower()	Tüm karakterlerin küçük olup olmamasına bakılır
isprintable()	Tüm karakterlerin bastırılabilirliğini kontrol eder (Örn özel karakterler \n gibi)

Python Temelleri II

String'ler

Stringler için kullanılabilen diğer fonksiyonlar

Fonksiyon	Açıklama
isspace()	Tüm karakterlerin space olup olmasını kontrol eder
isupper()	Tüm karakterlerin büyük olup olmasını kontrol eder
join()	Kendisine verilen karakteri, dizinin elemanları arasına koyarak, tek bir string yapar
ljust()	Argüman olarak integer alır. Sağa yaslanmış olarak string'i geri döndürür.
lower()	Tüm karakterleri küçültür
lstrip()	String'in sol taraftaki boşluk karakterlerini siler

Python Temelleri II

String'ler

Stringler için kullanılabilen diğer fonksiyonlar

Fonksiyon	Açıklama
partition()	Bir string içindeki alt string aranır ve bulunduğunda, bulunan'dan öncesi, bulunan ve sonrası olarak 3 parça halinde geriye döndürülür.
replace()	String'in içindeki bir alt string'i başka bir string ile değiştirmek için kullanılır.
rfind()	String'in içinden bir değer aranır ve son bulunan index geri döndürülür
rjust()	String'in sağa yaslanmış halini döndürür.
rpartition()	Aranan alt string'in en son bulunan adresinden itibaren, öncesi, kendisi ve sonrasını döndürür
rstrip()	Sağ taraftaki boşluk karakterlerini silerek geriye döndürür

Python Temelleri II

String'ler

Stringler için kullanılabilen diğer fonksiyonlar

Fonksiyon	Açıklama
split()	Verilen alt string'e göre, ana string'i elemanlara bölerek geriye dizi döndürür
splitlines()	\n newline karakteri ile string'i bölererek geriye dizi döndürür
startswith()	Verilen alt string, ana string'in başlangıcı ise, geriye doğru döndürür
strip()	Sağ ve sol boşluk karakterleri silerek geriye döndürür

Python Temelleri II

String'ler

Stringler için kullanılabilen diğer fonksiyonlar

Fonksiyon	Açıklama
swapcase()	Büyük harfleri küçük, küçükleri büyük yaparak döndürür
title()	Her bir kelimenin ilk harfini büyük yaparak döndürür
upper()	String'i büyük harflere dönüştürür
zfill()	String'in başlangıcına belirtilen integer değer kadar 0 ile doldurur.

Python Temelleri II

Matematik Operasyonları

Python'da matematik işlemlerinin yapılabilmesi için math isimli bir modül barındırmaktadır.

```
import math
```

Python Temelleri II

Matematik Operasyonları

Karekök alımı örneği, sqrt fonksiyonu

Örnek Kod Parçacığı

```
import math  
  
x = math.sqrt(64)  
  
print(x)
```

Çıktı

8.0

Python Temelleri II

Matematik Operasyonları

Ondalıklı sayıları yukarı yönde yuvarlamak için `ceil`, aşağı yönde yuvarlamak için `floor` fonksiyonu kullanılabilir.

Örnek Kod Parçacığı

```
import math

x = math.ceil(3.7)
y = math.floor(5.8)

print(x)
print(y)
```

Çıktı

```
4
5
```

Python Temelleri II

Matematik Operasyonları

Hazır tanımlı PI sayısı

Örnek Kod Parçasığı

```
import math  
  
x = math.pi  
  
print(x)
```

Çıktı

```
3.141592653589793
```

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.acos(x)</code>	Verilen X 'in arc kosinüs değerini hesaplar
<code>math.acosh(x)</code>	Verilen X 'in hiperbolik arc kosinüs değerini hesaplar
<code>math.asin(x)</code>	Verilen X 'in arc sinüs değerini hesaplar
<code>math.asinh(x)</code>	Verilen X 'in hiperbolik arc sinüs değerini hesaplar
<code>math.atan(x)</code>	Verilen X 'in arc tantajnt değerini hesaplar
<code>math.atan2(y, x)</code>	y/x 'in radyan cisinden arc tanjant değerini döndürür

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.atanh(x)</code>	Verilen X'in hiperbolik arc kosinüs değerini hesaplar
<code>math.ceil(x)</code>	Verilen X'i en yakın üst sayıya yuvarlar
<code>math.copysign(x, y)</code>	İlk argümanın değeri ile ikinci argümanın işaretini içeren sayıyı döndürür
<code>math.cos(x)</code>	Verilen X'in kosinüsünü döndürür
<code>math.cosh(x)</code>	Verilen X'in hiperbolik kosinüsünü döndürür

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.degrees(x)</code>	Radyan'dan dereceye dönüşüm yapar
<code>math.dist(p, q)</code>	Öklid uzaklığını verir
<code>math.exp(x)</code>	Euler sayısı e^x 'i döndürür
<code>math.fabs(x)</code>	Mutlak değer döndürür
<code>math.factorial()</code>	Faktöriyel döndürür

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.floor(x)</code>	Verilen sayıyı en yakın alt sayıya yuvarlar
<code>math.fmod(x, y)</code>	X'in y cinsinden modunu alır
<code>math.fsum(dizi)</code>	Verilen dizinin tüm elemanlarını toplar
<code>math.gcd(x,y)</code>	Verilen iki sayının EBOB'unu bulur

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.log(x, base)</code>	X'in verilen base tabanında değerini döndürür
<code>math.log10(x)</code>	X'i log10 tabanında değerini döndürür
<code>math.log2(x)</code>	X'in log2 tabanında değerini döndürür
<code>math.pow(x, y)</code>	X'in y kuvvetini alır.

Python Temelleri II

Matematik Operasyonları

Diğer Math modül fonksiyonları

Fonksiyon	Açıklama
<code>math.radians(x)</code>	Dereceden radyana dönüştürür
<code>math.sin(x)</code>	Sinus değerini döndürür
<code>math.sinh(x)</code>	Hiperbolik sinüs değerini döndürür
<code>math.sqrt(x)</code>	Karekök değerini döndürür
<code>math.tan(x)</code>	Tanjant değerini döndürür
<code>math.tanh(x)</code>	Hiperbolik tanjant değerini döndürür

Python Temelleri II

JSON Veri Değişimi Notasyonu

Veri alım ve gönderimini basitleştiren bir formattır.

İnsan'ın okuyabileceği bir metin formatında oluşturulur.

Kullanmak için

```
import json
```

Eklenmelidir

Python Temelleri II

JSON Veri Değişimi Notasyonu

JSON Formatı

```
{  
  "isim": "deneme",  
  "yas": 30,  
  "sehir": "Istanbul",  
  "adres": {  
    "cadde": "Test",  
    "mahalle": "Mahalle Test"  
  },  
  "isDurumu": "Yok"  
}
```

Python Temelleri II

JSON Veri Değişimi Notasyonu

Örnek Kod Parçasığı

```
import json

x = '{ "isim":"deneme", "yas":
30, "sehir":"Istanbul",
"adres":{" cadde" : "Test",
"mahalle" : "Mahalle Test" },
"isDurumu":"Yok" }'
```

```
y = json.loads(x)

print(y["isim"])
print(y["adres"]["cadde"])
```

Çıktı

```
deneme
Test
```

Python Temelleri II

JSON Veri Değişimi Notasyonu

Bitcoin JSON Veri yakalama örneği,

<https://api.coindesk.com/v1/bpi/currentprice.json>

Adresi json formatında Bitcoin fiyatları vermektedir

Python Temelleri II

JSON Veri Değişimi Notasyonu

Örnek çıktı:

```
{"time":{"updated":"Aug 15, 2020 12:34:00 UTC","updatedISO":"2020-08-15T12:34:00+00:00","updateduk":"Aug 15, 2020 at 13:34 BST"},"disclaimer":"This data was produced from the CoinDesk Bitcoin Price Index (USD). Non-USD currency data converted using hourly conversion rate from openexchangerates.org","chartName":"Bitcoin","bpi":{"USD":{"code":"USD","symbol":"$","rate":"11,824.8897","description":"United States Dollar","rate_float":11824.8897},"GBP":{"code":"GBP","symbol":"£","rate":"9,052.1541","description":"British Pound Sterling","rate_float":9052.1541},"EUR":{"code":"EUR","symbol":"€","rate":"10,012.3943","description":"Euro","rate_float":10012.3943}}}}
```

Python Temelleri II

JSON Veri Değişimi Notasyonu

JSON'u analiz etmek için bir görüntüleyici araç kullanılabilir

Örn.

<http://jsonviewer.stack.hu/>

Python Temelleri II

JSON Veri Değişimi Notasyonu

Örnek Kod Parçasığı

```
import urllib.request, json

with
urllib.request.urlopen("https://api.coindesk.com/v1/bpi/currentprice.json") as
url:
    data =
json.loads(url.read().decode())
    print(data["bpi"]["USD"]["rate"])
```

Çıktı

11,813.9635

Python Temelleri II

Tarih Operasyonları

Zaman ile ilgili verilerin elde edilmesi için datetime modülü kullanılabilir.

Örnek Kod Parçasığı

```
import datetime  
  
x = datetime.datetime(2020, 5, 17)  
  
print(x)  
Try it Yourself »
```

Çıktı

```
2020-08-15 15:45:13.879862
```

Python Temelleri II

Tarih Operasyonları

Zaman ile ilgili verilerin elde edilmesi için datetime modülü kullanılabilir.

Örnek Kod Parçasığı

```
import datetime

x = datetime.datetime.now()

print(str(x.day) + " " + str(x.month) +
      " " + str(x.year) )
```

Çıktı

15 8 2020

Python Temelleri II

Scope - Kapsam

Değişkenler tanımlı olduğu scope'ta kullanılabilirler. Scope aralığı dışında kullanımda hata verecektir.

Örnek Kod Parçacığı

```
def testFnc():  
    x = 300  
    print(x)
```

```
myfunc()  
print(x)
```

X fonksiyonda lokal olarak tanımlandı

X tanımsız olarak hata verecektir

Python Temelleri II

Scope - Kapsam

Global tanımlanmış değişkenler, her yerden erişilebilirler

Örnek Kod Parçasığı

```
def testFnc():  
    global x  
    x = 300  
    print(x)
```

```
myfunc()  
print(x)
```

X global tanımlandı

X bastırılırken hata vermeyecektir

Python Temelleri II

İstisna İdaresi

Kod çalışma esnasında karşılaşılan bir hatada, kodun çalışmayı durdurması yerine, hata duruma göre kod akışının devam ettirilmesinin sağlanması için kullanılır.

Örnek Kod Parçası

```
print(x)
```

X tanımsız olduğu için hata verecektir.

Python Temelleri II

İstisna İdaresi

Kod çalışma esnasında karşılaşılan bir hatada, kodun çalışmayı durdurması yerine, hata duruma göre kod akışının devam ettirilmesinin sağlanması için kullanılır.

`try:`

`## Kod parçacıkları`

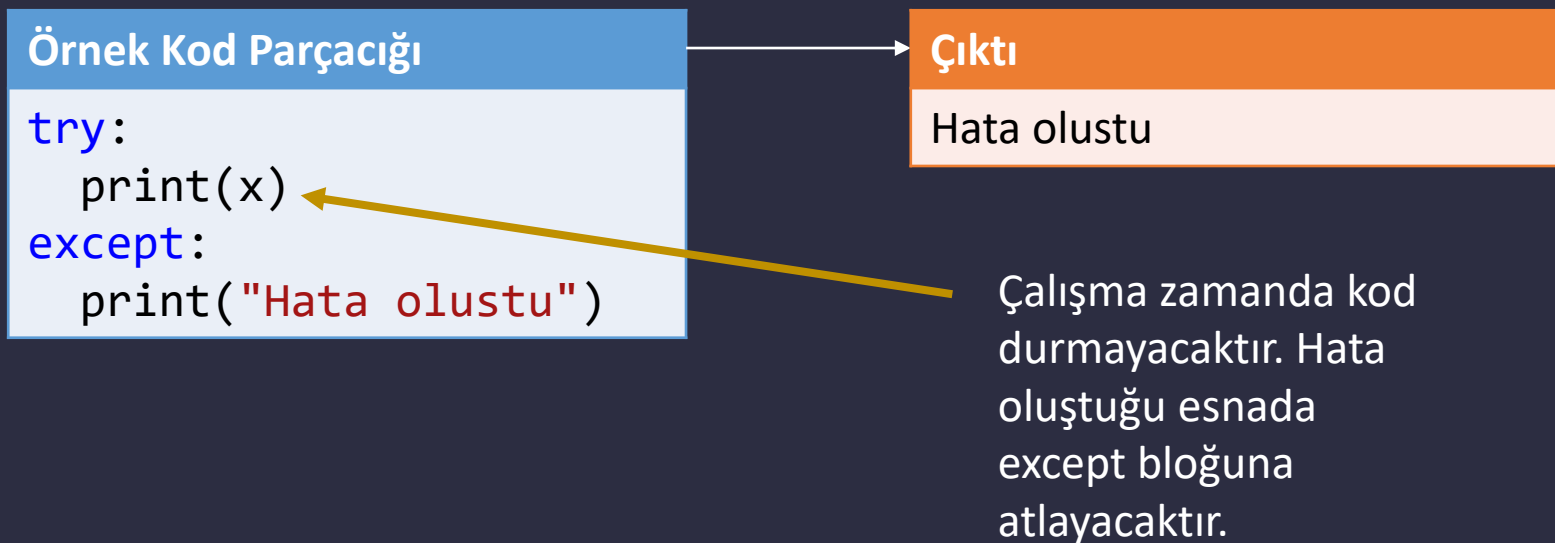
`except:`

`## Hata olduğu durumda koşacak kodlar`

Python Temelleri II

İstisna İdaresi

Kod çalışma esnasında karşılaşılan bir hatada, kodun çalışmayı durdurması yerine, hata duruma göre kod akışının devam ettirilmesinin sağlanması için kullanılır.



Python Temelleri II

Rasgele Sayı Üretimi

Random modülü kullanılarak üretilebilir

Örnek Kod Parçasığı

```
import random  
  
x = random.randint(3, 9)  
y = random.random()  
  
print(x)  
  
print(y)
```

Çıktı

```
4  
0.8546576843373211
```

3 ile 9 arasında rasgele tamsayı sayı üretecektir.
0-1 arasında rasgele sayı üretecektir.

Python Temelleri II

Modüller

Kod kütüphaneleridir.

Ana kodun bulunduğu python dosyasından ayrıca bir py uzantılı dosya oluşturularak tanımlanır.

Python Temelleri II

Modüller

yenimodul.py

```
def denemeFonksiyonu(arg):  
    print("test, " + arg)  
  
degiskenTest = 3
```

main.py

```
import yeniModul  
  
yeniModul.denemeFonksiyonu("Emre")  
print(yeniModul.degiskenTest)
```

Çıktı

```
test, Emre  
3
```