

PYTHON



İÇERİK

- ✓ Giriş Çıkış İşlemleri
- ✓ Stringler
- ✓ Listeler

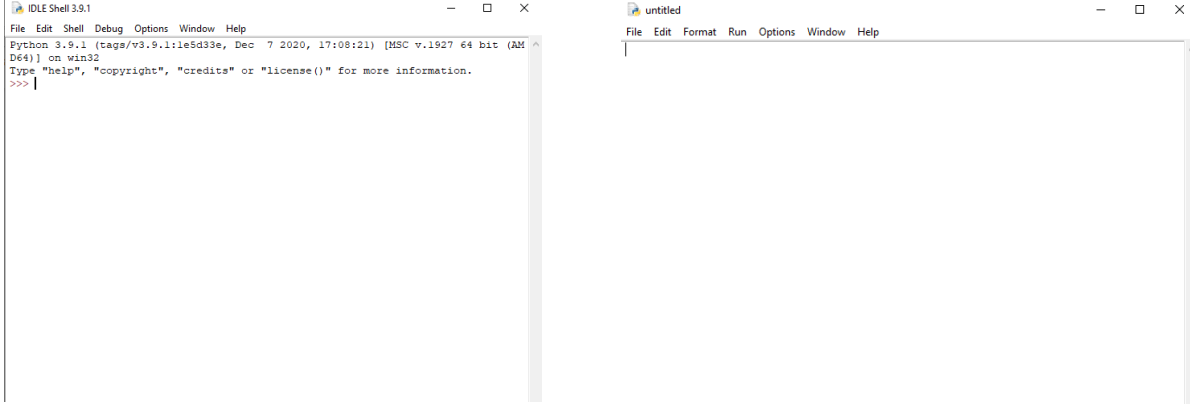
KAZANIMLAR

- ✓ Kullanıcıdan veri almayı öğrenir
- ✓ Kullanıcıdan alınan veriler ile işlem yapmayı öğrenir
- ✓ String ifadesini ve sözcüklerin hangi veri tipi ile tutulduğunu öğrenir.
- ✓ Liste kavramını ve yazım dizimini öğrenir



Ders Aşamaları

Bu dersin işlenişi için Python ara yüzü olarak kullandığımızı IDLE SHELL uygulaması açılır. Açılan bu pencere üzerinden **File** **→** **New File** seçeneği seçilir.



Açılan bu yeni sayfada kodlar yazıldıktan sonra **Run** seçeneğine tıklanır. Program çalıştırılır.

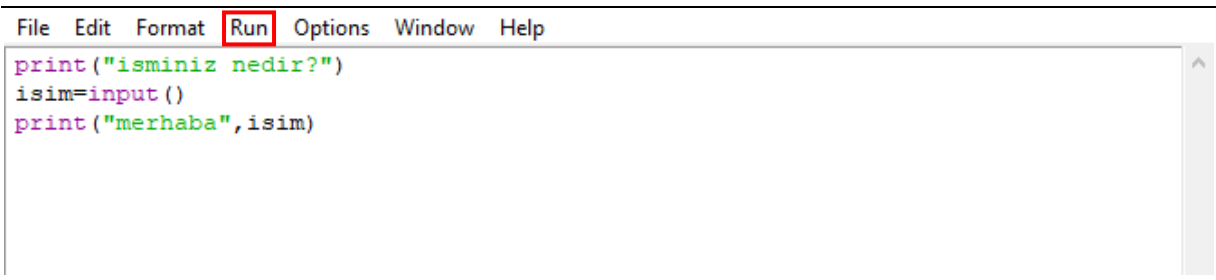
Giriş Çıkış İşlemleri

Kullanıcıdan veri almak

Python'da kullanıcı ile etkileşime geçmek için **Input ()** fonksiyonu kullanılır.

Input () fonksiyonu kullanım örnekleri:

Kullanıcıya ismini soran ve aldığı cevaba göre çıktı veren program



Kodu yazdıktan sonra **Run(F5)** sekmesine tıklanır ve program çalıştırılır.

Açılan yeni sekmeden istenilen veriler girilerek programın çıktısı alınır.



Python ile kullanıcıdan veriler alarak bir telefon rehberi programı oluşturalım

```
print('isminiz=')
isim=input()
print('soyisminiz=')
soyisim=input()
print("telefon numaranız=")
telno=input()
print(isim+' '+soyisim+' '+telno)
```

İki sayının toplamını kullanıcıdan alan ve aldığı verileri toplayan program

Burada dikkat edilmesi gereken nokta input() fonksiyonumuz ile veri alırken değişkenin tipini belirtmek gerekir. Örneğin bir sayısal veri alınacaksa değişken tipi int veya float olarak belirtilmelidir.

```
print ("İki Sayıyı Toplayan Program")
sayi1 = int(input("Birinci Sayıyı Giriniz:"))
sayi2 = int(input("İkinci Sayıyı Giriniz:"))
toplam = sayi1+sayi2
print(toplam)
```



Stringler

Bu bölümde daha önce bahsettiğimiz değişken türü olan “String”leri ve kullanım amaçlarını anlatacağız. Bir önceki dersimizde de string ifadesini ekrana yazdırmıştık. Merhaba dünya yazdırdığımız örneğimizde string veri tipini kullanmıştık. Ekrana yazdırmış olduğumuz bu cümle harflerden oluşan bir String ifadedir. Burada dikkat edilmesi gereken nokta Stringler sadece harflerden oluşmaz. Örneğin tırnak içerisinde “2021” yazarsak bu da string bir ifade olur. Python kullanım itibari ile bir, iki ve üç tırnak kullanımına izin vermektedir.

Çift Tırnak Bir değişkene bir dize yazmak için, değişken adı ve ardından eşittir işareti ve dize ile yapılır:

```
python="python öğreniyorum"  
print(python)
```

Üç tırnak işareti kullanarak bir değişkene çok satırlı bir dize atayabilirsiniz:

```
python="""Python bir programlama dilidir.  
Python ile çeşitli uygulamalar geliştirebiliriz, ve farklı çalışmalar yapabiliriz.  
Örneğin python ile oyun yapabilir, web sitesi geliştirebiliriz."""  
Print(python)
```

Bir dizenin uzunluğunu öğrenmek için **len()** fonksiyonu kullanılır.

```
python="python öğrenelim"  
print(len(python))
```



Yazdıracağımız değerin tamamını büyük harfle yazdırmak için upper() fonksiyonu kullanılır.

```
python="python öğreniyorum"  
print(python.upper())
```

Yazının tamamını küçük harf ile yazdırmak için ise lower() fonksiyonu kullanılır.

```
python="PYTHON ÖĞRENİYORUM"  
print(python.lower())
```

Sıra Sizde

len () komutu ile aşağıdaki dizeninin uzunluğunu bulunuz. Eksik kod satırını tamamlayınız.

```
x = "Merhaba Dünya"  
print( )
```

Cevap: len(x)



Listeler

Listeler, tek bir değişkende birden çok öğeyi depolamak için kullanılır.

Bir alışveriş programı yazdığınızı düşünelim. Alışveriş sepetine eklemeler yapıyorsunuz. Elmayı eklediniz, yoğurdu eklediniz, peçeteyi eklediniz. Liste böyle uzayıp gidiyor. Şimdi bütün bunları ayrı ayrı değişkenler olarak kaydetmek kolay olur yoksa “Alışveriş Sepetim” şeklinde bir liste tanımlayıp içine koymak mı? Elbette liste kullanmak işinizi kolaylaştıracaktır. İşte bunun için de Listeler değişkenleri ile sizi tanıştırmak istiyorum.

Elemanları köşeli parantez içerisine koyar ve virgülle ayırırız. Örnek bir liste verelim.

```
listem=["elma","yoğurt","ekmek"]  
print(listem)
```

şimdi listemize yeni bir eleman atayalım ve bu eleman sayısal bir değer olsun.

```
listem=["elma","yoğurt","ekmek",34]  
print(listem)
```

Bu değişkenimiz tırnaklar içerisinde değil. Ayrıca bir sayı. Dolayısıyla integer türünde. Listemiz sorunsuz çalıştı. Burada da gördük ki listenin içine hem String hem de Integer ifadeler aynı anda eklenebilir.

Şimdi len() fonksyonu ile listemizde kaç eleman olduğuna bakalım.

```
listem=["elma","yoğurt","ekmek",34]  
print(len(listem))
```



Listeleri alfabetik ve sayısal olarak sıralayalım. Bunun için `sort()` fonksiyonu kullanılır.

```
listem=["elma","yoğurt","ekmek","armut","kivi"]  
listem.sort()  
print(listem)
```

Sayıları sıralayalım

```
sayılar = [100, 50, 65, 82, 23]  
sayılar.sort()  
print(sayılar)
```

Python'da iki veya daha fazla listeyi birleştirmenin veya birleştirmenin birkaç yolu vardır.

En kolay yollardan biri `+` operatörü kullanmaktır.

```
list1 = ["a", "b", "c"]  
list2 = [1, 2, 3]  
list3 = list1 + list2  
print(list3)
```



Liste öğelerine bulmak için indis(sıra) numarasını yazdırmak yeterlidir.

```
meyveler = ["elma", "muz", "kiraz"]  
print(meyveler[1])
```

Belirli bir öğenin değerini değiştirmek için indeks(indis) numarasına bakılır. Örneğin muz yerine karpuz yazalım.

```
meyveler = ["elma", "muz", "kiraz"]  
meyveler[1]="karpuz"  
print(meyveler)
```

Sıra Sizde

Şimdi meyveler listemizdeki ilk elemanı yazdırmak için boşluğu tamamlayın.

```
meyveler = ["elma", "muz", "kiraz"]  
print( )
```

Cevap meyveler[0]

Neler Öğrendik

Bu dersimizde python ile nasıl dışardan veri alacağımızı kullanıcı ile python arasında nasıl etkileşim kurabileceğimizi öğrendik. Kullanıcıdan aldığımız veriler ile işlem yapmayı öğrendik.

String ifadesinin çoklu kullanım çeşitlerini ve string fonksiyonuna ait parametrelili kullanmayı öğrendik.

Liste kavramını, liste ile değişken arasında ki farkı öğrendik. Listeleme fonksiyonun kullanımını ve listeleme ile çeşitli araştırmalar yaptık.

