

PYTHON



İÇERİK

- ✓ **Python Nedir?**
- ✓ **Python Kurulumu Nasıl Yapılır?**
- ✓ **Python ile Basit Hesaplamalar**
- ✓ **Veri Tipleri**

KAZANIMLAR

- ✓ Programlama nedir ve neden gereklidir öğrenir
- ✓ Python nedir ve ne için kullanılır kavrar.
- ✓ Print komutunu öğrenir
- ✓ Basit hesaplamalar yapmayı öğrenir
- ✓ Veri tipi nedir ve veri tiplerin nelerdir öğrenir
- ✓ Değişken kavramını ve nasıl kullanılması gerektiğini öğrenir



Neden Programlama Öğrenmeliyiz?



Neden programlama öğreniyoruz. Buna gerçekten ihtiyacımız var mı ? Bu soruya her zamankinden daha güçlü bir şekilde “Evet” cevabını verdiğimiz günlerden geçiyoruz. Çünkü programlama yaratıcılığı destekler ve bireye problem çözme, karar verme becerisi kazandırır. Programlama bizlere yoktan bir şeyleri var etme olanağı sunar. Hem yeni dünya teknolojilerini yakalamak hem de var olan düzen içerisinde kendimizi daha ileriye taşımak adına programlama öğrenmek oldukça önemlidir.

Bugün izlediğiniz çizgi filmler ve oynadığınız sanal oyunları kendimiz tasarlamak ve orada var olmak aslında parmaklarımızın hemen ucunda. Bunu yapabilmek içinde algoritmik düşünme ve programlama bilme becerilerine sahip olmak gereklidir.

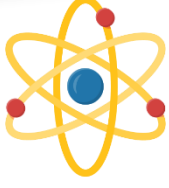
Evet şu an yeni bir dünya kurmak veya var olan dünyaya şekil vermek parmaklarımızın hemen ucunda biraz zorlu ama oldukça eğlenceli bu göreve hazır mısınız?

Haydi başlayalım...

Neden Python

Python ısırmayan son derece dost canlısı bir arkadaşımız. Girmiş olduğumuz bu yeni dünyada bize yardımcı olacak ve eminim sizlerde onu çok seveceksiniz. İstedğimiz komutları en kısa yoldan yerine getiriyor ve oldukça hamarat. Verdiğimiz işleri hızlı bir şekilde yerine getiriyor ve birçok farklı alanda çalışabiliyor. Aynı zamanda ona öğrettiğimiz her şeyi oldukça hızlı öğreniyor.

Python oldukça basit bir dil yapısına sahip olmakla beraber bizlere oyunlar ve grafikler tasarlama imkânında sunuyor. Çeşitli modüller ve bu modüllerin projelere kolay entegrasyonu ile bizleri büyük iş yüklerinden kurtarıyor.

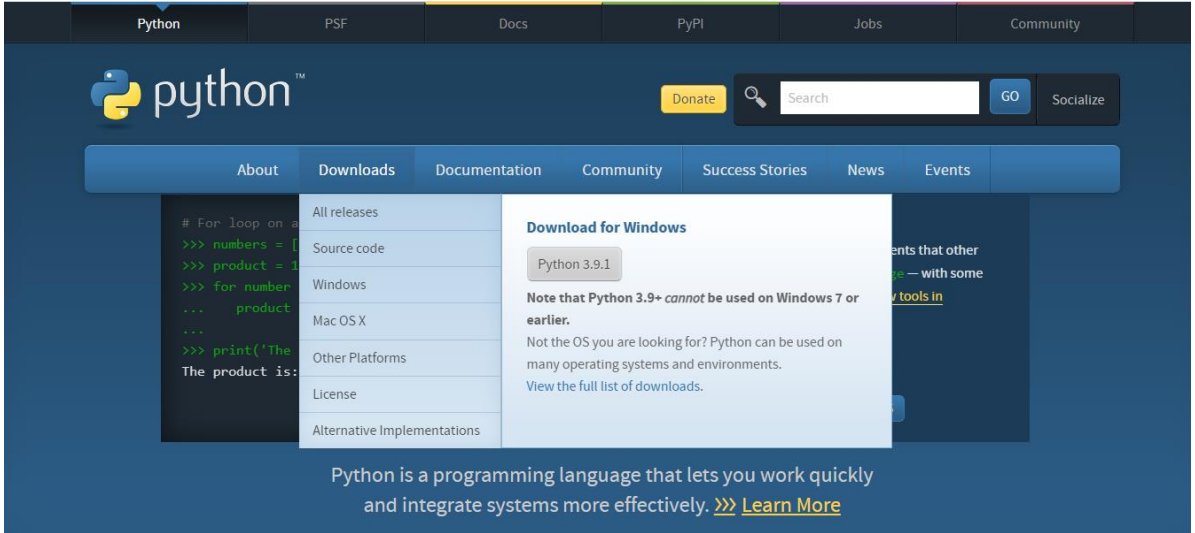


Nasıl Öğreneceğiz?

Burada çıkacağımız Python öğrenme serüvenini önünüze koyulan bir orkestra puzzle'ını yapmak gibi düşünebilirsiniz. Düşünün ki hayatınızda hiç orkestra görmediniz. Müzik aletlerini bilmiyorsunuz, ne yapardınız ? Öncelikle müzik aletlerini öğrenir, ardından da orkestra düzeni içerisindeki yerlerini keşfederdiniz. Sonunda da bütün parçaları birleştirerek puzzle'ı tamamlardınız. İşte burada da ilk bölümlerde Python ile yapabileceğimiz örnekleri inceleyeceğiz. Yani müzik aletlerimizi tanıyacağız. Ardından da örnek projelerde farklı müzik aletlerini beraber kullanarak tıpkı orkestra puzzle'ında olduğu gibi bir büyük resim ortaya çıkaracağız. Her bölümün sonunda da yılanımız size küçük sorular sorarak, dersleri iyi dinleyip dinlemediğinizi görecektir. Eğer soruları bilmezseniz ısırmayacak korkmayın, Ama sizden biraz daha tekrar yapmanızı isteyebilir.

Python Kurulumu

Python kurulumu son derece basit <https://www.python.org/> adresine giriyoruz.



Ardından Download Python 3.9.1 diyerek adımları izliyoruz.

Kurulumdan sonra IDLE dosyamıza tıklayarak Python derleyicimizi açıyoruz. Hemen ilk kodumuzu yazmaya başlayalım. Ekranı "Hello World " yazdıralım.



```
IDLE Shell 3.9.1
File Edit Shell Debug Options Window Help
Python 3.9.1 (tags/v3.9.1:1e5d33e, Dec 7 2020, 17:08:21) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('hello World')
hello World
>>> |
```

Tebrikler ilk python programınızı yazdınız. Burada kullandığımız print komutu adından da anlaşılacağı gibi ekrana yazı yazdırılmasını sağlar.

İlk uygulamamızda da görüldüğü gibi print komutu ile ekrana yazı yazdırmak için yapmamız gereken sadece parantez açmak ve tırnak koymak. Aynı zamanda print ile ekrana yazı yazdırmak için sadece çift tırnak("yazı") kullanılmak zorunda değildir. Aynı zamanda tek tırnak('yazı') ve üç tırnak("""yazı""") da kullanılabilir.

Python ile Basit Hesaplamalar



Normalde iki sayının çarpımını bulmamız istendiğinde bunu yapmak için ya hesap makinesi ya da kalem kağıt kullanırız. Peki bu hesaplamaları python ile yapmaya ne dersiniz. Haydi deneyelim.

IDLE Shell editörümüzü açtıktan sonra komut istemine şu denklemini giriniz.

```
>>> 8 * 3.57
28.56
```

Python'da bir çarpma hesaplaması girerken, çarpma işareti (×) yerine yıldız simgesini (*) kullanırsınız

Biraz daha zor bir denklem deneyelim:



Bir gün bir yolda giderken 20 tane altın buldunuz ve bu altın paraları klonlayabileceğiniz bir makineye sahipsiniz. Bu makine size günde sadece 10 tane altın klonlama hakkı sunuyor. Eğer bu işlemi 365 gün yapıyor olsaydınız. Bir yıl içerisinde ne kadar para elde ederdiniz.

Elbette bu işlemleri hesap makinesi veya kağıt üzerinde yapmak kolay ama gelin biz python ile yapalım.

```
>>> 10 * 365  
3650  
>>> 20 + 3650  
3670
```

Peki ya oluşturduğunuz altınları sakladığınız yeri küçük bir arkadaş görse ve her hafta içeri girip üç altın çalmaya başlarsa bir yıl sonunda ne kadar altınımız çalınmış olur?

```
>>> 3 * 52  
156  
>>> 3670 - 156  
3514
```

İlk olarak, 3 altını yılda 52 hafta ile çarpıyoruz. Sonuç 156'dır. Bu sayıyı toplam altın adedimizden (3670) çıkarıyoruz ve sonunda 3514 altınımızın kalacağını sonucunu elde ediyoruz.



Python Operatörler

Operatör	Fonksiyonu
+	Toplama
-	Çıkarma
*	Çarpma
/	Bölme
%	Kalan Bulma
**	Üs Alma
//	Kalansız Bölme

Aritmetik operatörler bizim Matematik işlemlerinde kullandığımız operatörler aslında. Matematik denildiğinde hangi işlemler aklımıza geliyor. “Toplama, çıkarma, çarpma, bölme, kalan bulma” değil mi? Ayrıca ilerleyen sınıflarda üs alma işlemini de görüyoruz. İşte bu işlemlerin aynılarını Python’da operatörler ile yapıyoruz. Şimdi bu operatörleri bir tabloda gösterelim.

Not: İşlem öncelik sıralarını kontrol etmek için parantez kullanılır.

Örneğin, aşağıdaki denklemde 30 ve 20 sayıları önce çarpılır ve ürününe 5 rakamı eklenir.

```
>>> 5 + 30 * 20
```

```
605
```

Önce 30 ile 5 i toplamak için aşağıdaki işlem yaptırılır

```
>>> (5 + 30) * 20
```

```
700
```

Bu denklemin sonucu 700 (605 değil) çünkü parantezler Python'a işlemi yapmasını söylüyor önce parantez içinde, sonra işlemi parantezlerin dışında yapın.



Bölme işlemi yapalım:

```
>>> ((5 + 30) * 20) / 10  
70.0
```



Unutmayalım: Çarpma ve bölme işlem önceliği olarak her zaman toplama ve çıkarmadan önce gelir. Burada önemli olan işlem önceliğinde toplama veya çıkarma istiyorsak parantez kullanmak. Bu durumda, Python önce en içteki parantezleri değerlendirir, sonra dış olanlar ve sonra son bölme operatörünü.

Veri Tipleri

Bu bölümde veri tipleri ve değişken kavramını öğreneceğiz. Burada şöyle küçük bir örnek verebiliriz. Şöyle düşünelim, evden çıktık markete alışveriş yapmaya gideceğiz ve yol oldukça kalabalık eğer yolda gördüğümüz kişiyi tanımiyorsanız hiçbir şey yapmazsınız. Uzaktan tanıyorsak selam veriniz. Yakın arkadaşımız ise durup sarılırız.

Aslında biz burada karşılaştığımız insanların “tipini” inceliyoruz. Ve buna göre karar veriyoruz.

İşte tıpkı bizim gibi programlarda verilerin tipine göre işlem yapar. Makineler bir işlem yapmadan önce verinin tipine bakar ve buna göre belleğinde yer ayırır, işlem yapar.

Değişken Nedir?

Şöyle ki, kitapçıya gidip alışveriş yapıp kitaplar aldığımızı ve bu kitapları eve geldiğimizde kitaplığımızın boş olan raflarına yerleştirdiğimizi düşünelim.

Programımızı yazarken elimizdeki kitaplar değişkenler, onları koyduğumuz kitaplık ise program gibidir. Program bizim değişkenlerimize göre onları hafızasına yerleştirir.

Peki neden değişkenlere ihtiyacımız var?

Bunun cevabı oldukça basit. Programdan eğer bir şeyler almak istiyorsak önce ona bir şeyler öğretmemiz lazım. Tıpkı günlük hayatta olduğu gibi. Örneğin ben size iki tane sayı söylüyorum ve bunları çarpmanızı istiyorum. Size “İlk sayı 3” dedim. Siz bunu aklınızın bir kenarında tutuyorsunuz. Ardından “İkinci sayı 5” dedim. Siz de aklınızda tuttuğunuz 3 ile 5 i çarparak 15 buldunuz. Harika ! Bir tane daha yapalım. Şimdi size “İlk sayı 4” dedim. Artık zihninizdeki ilk sayıyı 3 e değil 4 e eşitlediniz. Ardından “İkinci sayı 6” dedim. Bu sefer aklınızdaki ikinci sayı değişkenini değiştirmiş olduk.

Adı üstünde “değişiyor” değil mi ? İşte tıpkı zihninizin işleyişi gibi programlamada da bir işleyiş var.

Haydi şimdi bu işlemi python ile yapalım.

```
>>> ilksayi=3
>>> ikincisayi=5
>>> ilksayi*ikincisayi
15
```

Burada kısaca şunu yaptık ilksayi isminde bir değişken tanımladık ve değerini 3 olarak belirledik. Ardından ikincisayi isminde bir değişken tanımladık ve değerini 5 olarak belirledik. Sonrasında ise bu iki değişkeni ilksayi * ikincisayi komutu ile çarparak sonuç olan 15’i ekrana yansıttık.



Şimdi birde değişkenlere değer atayarak sadece değişkeni çağırıp çıktı alalım:



Öncelikle sepetimizde 200 tane elma olduğunu varsayalım. Elma isimli bir değişken oluşturalım

```
>>> elma=200
>>> print(elma)
200
```

Şimdi sepetimizdeki 200 tane armut olduğunu ve bunun için elmadan aynı adeti alalım:

```
>>> elma=200
>>> armut=elma
>>> print(armut)
200
```

Artık değişkenlerimizi oluşturduktan sonra bu değişkenlerin tipi hakkında bilgi alalım. Bahsettiğimiz gibi her değişkenin bir veri tipi var. Bu veri tiplerinin ne olduğu öğrenmek için İngilizcede tip anlamına gelen “type” ifadesini kullanacağız.

Bir önceki örnekte yaptığımız ilksayi ve ikincisayi değişkenlerimizin tipini belirleyelim:

```
>>> ilksayi=3
>>> type(ilksayi)
<class 'int'>
```



Programda da gördüğümüz veri tipimiz integer yani tamsayılar olarak adlandırdığımız sayı kümesi.

Haydi şimdi bir ondalıklı sayı yazalım ve veri tipimizin ne olacağını görelim:

```
>>> ilksayi=4.8
>>> type(ilksayi)
<class 'float'>
```

Gördüğünüz gibi bu sefer class int değil class float yazdı. Yani veri tipimiz değişti. Bu şekilde 6 tane ana veri tipi çeşidimiz var bunlar:

1. Numbers(Sayılar)
2. String(Karakter Dizileri)
3. List(Listeler)
4. Tuple(Demetler)
5. Dictionary(Sözlükler)
6. Kumeler

Burada int ve float ifadesinin olmadığını söylediğinizi duyar gibiyim. Bu veri tiplerimiz var burada mevcut çünkü onlar Numbers(Sayılar) değişkenimiz içerisinde yer alıyor.

Ek Örnek: Dersin ilk başında yaptığımız örnekteki hikayeyi hatırlayalım bu hikaye içerisinde bulduğumuz 20 altın vardı ve biz bu altınları kopyalama cihazımız ile 365 gün çoğaltıyorduk küçük bir problemimiz vardı ama her hafta 3 altınımız çalınıyordu. Gelin şimdi bu kodu değişkenler aracılığı ile oluşturalım.

Öncelikle değişkenlerimi ve içerisindeki değerleri tanımlamalıyım:

```
>>> altın=20
>>> kopya=10
>>> kayıp=3
```



Değişkenlerimi ve içlerindeki değerleri tanımladıktan sonra şimdi de denklemimizi kuralım:

```
>>> altın+kopya*365-kayıp*52
```

```
3514
```



Sıra Send'e

Değişkenlerin veri tiplerini belirleyebilirsin...

Neler Öğrendik

Bu dersimizde basit denklemlerin nasıl python programında yazıldığını öğrendik. Python aritmetik operatörleri kullanırken parantez kullanmanın ve işlem önceliğinin öneminden bahsettik. Değişkenlerin nasıl oluşturulacağını ve değişkenlere nasıl değere ataması yapılacağını öğrendik. Değişkenlerin tiplerini belirlemeyi ve bu veri tiplerine uygun değerler atamayı öğrendik.

