What is Python?

As you can imagine, Python is a programming language (such as Java, C ++, R, Ruby, and so on). Like other programming languages, it allows you to control the machine in front of you, the computer.

The IT industry is booming with Data science applications using,

Artificial intelligence,

Deep learning and

Machine learning algorithms.

Python is the most widely used technology in this domain. With the new-age applications, demand for a Python developer has also increased.

According to the [Google search trends](https://trends.google.com/trends/explore?q=%2Fm%2F05z1_,%2Fm%2F07sbkfb,%2Fm%2F02p97,%2Fm%2F0jgqg), in 2021, Python is the most popular searched term, among all programming languages.

According to Tiobe.com's index of programming languages, Python is the fastest-growing language.

Python has become the second-most-popular language in GitHub, overtaking Java for the first time, according to GitHub’s 2020 State of the [Octoverse report](https://octoverse.github.com/) on the usage of the popular code-sharing site.

Programs can be developed very quickly with this language. In addition, the simple and clean syntax of the Python programming language has made it a preferred language by many programmers. It's easy to write programs and read a program written by others. So, it has been widely used - especially in Data Science - and has received lots of demands in recent years.

| Python Usage |
| --- |

Python nedir?

Tahmin edebileceğiniz gibi Python bir programlama dilidir (Java, C++, R, Ruby vb. gibi). Diğer programlama dilleri gibi önünüzdeki makineyi yani bilgisayarı kontrol etmenizi sağlar.  
  
BT endüstrisi, aşağıdakileri kullanan Veri bilimi uygulamalarıyla patlama yaşıyor,  
  
Yapay zeka,  
Derin öğrenme ve  
Makine öğrenimi algoritmaları.  
Python bu alanda en çok kullanılan teknolojidir. Yeni çağ uygulamalarıyla birlikte bir Python geliştiricisine olan talep de arttı.  
  
Google arama trendlerine göre 2021 yılında tüm programlama dilleri arasında en çok aranan terim Python oldu.  
  
Tiobe.com'un programlama dilleri endeksine göre Python en hızlı büyüyen dildir.

GitHub'ın popüler kod paylaşım sitesinin kullanımına ilişkin 2020 State of the Octoverse raporuna göre Python, GitHub'da Java'yı ilk kez geçerek ikinci en popüler dil haline geldi.

Programlar bu dil ile çok hızlı bir şekilde geliştirilebilir. Ayrıca Python programlama dilinin basit ve temiz söz dizimi, onu birçok programcı tarafından tercih edilen bir dil haline getirmiştir. Program yazmak ve başkaları tarafından yazılmış bir programı okumak kolaydır. Bu nedenle, özellikle Veri Biliminde yaygın olarak kullanılmaktadır ve son yıllarda çok sayıda talep almıştır.

The Programming Language of the Agile Era (Optional)

The contents in this section are developed by making use of the official brochure (Python Brochure Vol.I 2nd Edition, named: Case Studies & Success Stories), which was prepared by the Python Software Foundation (PSF) to introduce Python. You can access all the brochure [here](https://brochure.getpython.info/).

Programming with Python

Software quality is a vital ingredient to success in industry and science. Ubiquitous IT systems control the business processes of the global economy. Increasingly powerful computers and sophisticated algorithms provide the platform for new scientific discoveries. And global communication is inconceivable without intelligent software. In the race for customers, the pole position belongs to those who get to market faster than their competitors. Better and more creative solutions combined with the ability to respond instantly to new challenges drive the race. Writing secure and reliable programs in a fraction of the time normally required gets you first across the finish line.

The Programming Language of the Agile Era :

Agility is the hallmark of our times and Python is the programming language of the agile era. The Python universal programming language is the turbocharger of the IT department. Compared with other modern programming languages such as Java or C, Python achieves superior results in significantly shorter timescales for a number of different reasons.

For example, Python is a very lean programming language. Python programs are a great deal shorter than code written in other modern programming languages. As a result, both development times and maintenance costs are drastically reduced. Less code means fewer errors, meaning the cost of identifying and eliminating these errors is also reduced.

A comprehensive standard library and thousands of additional libraries in the Python Package Index provide developers with high-quality solutions that they can easily integrate into their applications to meet virtually any requirement.

In this way, Python frees up vast resources, which can be earmarked for more productive use elsewhere.

The Master Key for System Integration :

Python offers unique benefits for system integration. On the one hand, there are huge numbers of Python libraries, with which virtually any third-party system can be integrated. On the other hand, the libraries of many other programming languages can also be used in Python.

Once they have been programmed, Python applications can run on all operating systems for which a Python interpreter exists, significantly reducing the cost of operating-system-specific applications.

The Language that has Changed Everything :

For over 20 years, Python has been used successfully throughout the world as a programming language in industry, in the service sector, and also in research and science to meet a wide range of different requirements. In this time, the language has changed many things.

The Python programming language is easy to learn. It has blurred the boundaries between users and developers. Increasing numbers of scientists, engineers, financial experts, and others with little programming experience are using Python to solve specific complex technical problems.

Çevik Çağın Programlama Dili (Opsiyonel)

Bu bölümdeki içerikler Python Yazılım Vakfı (PSF) tarafından Python'u tanıtmak için hazırlanan resmi broşürden (Python Brochure Vol.I 2nd Edition, Case Studies & Success Stories) yararlanılarak geliştirilmiştir. Broşürün tamamına buradan ulaşabilirsiniz.  
  
Python ile Programlama  
Yazılım kalitesi, endüstride ve bilimde başarı için hayati bir bileşendir. Her yerde bulunan BT sistemleri, küresel ekonominin iş süreçlerini kontrol eder. Giderek daha güçlü bilgisayarlar ve karmaşık algoritmalar, yeni bilimsel keşifler için bir platform sağlar. Ve küresel iletişim, akıllı yazılım olmadan düşünülemez. Müşteri yarışında, pazara rakiplerinden daha hızlı girenler birinci sırada yer alıyor. Yeni zorluklara anında yanıt verme yeteneğiyle birleşen daha iyi ve daha yaratıcı çözümler, yarışı yönlendiriyor. Normalde gereken zamanın çok daha kısa bir sürede güvenli ve güvenilir programlar yazmak, bitiş çizgisini ilk geçmenizi sağlar.  
  
Çevik Çağın Programlama Dili:  
  
Çeviklik, zamanımızın ayırt edici özelliğidir ve Python, çevik çağın programlama dilidir. Python evrensel programlama dili, BT departmanının turboşarjıdır. Java veya C gibi diğer modern programlama dilleriyle karşılaştırıldığında Python, birkaç farklı nedenden dolayı önemli ölçüde daha kısa zaman ölçeklerinde üstün sonuçlar elde eder.  
  
Örneğin Python çok yalın bir programlama dilidir. Python programları, diğer modern programlama dillerinde yazılan kodlardan çok daha kısadır. Sonuç olarak, hem geliştirme süreleri hem de bakım maliyetleri önemli ölçüde azalır. Daha az kod, daha az hata anlamına gelir, yani bu hataları belirleme ve ortadan kaldırma maliyeti de azalır.  
  
Python Paket Dizini'ndeki kapsamlı bir standart kitaplık ve binlerce ek kitaplık, geliştiricilere hemen hemen her gereksinimi karşılamak için uygulamalarına kolayca entegre edebilecekleri yüksek kaliteli çözümler sunar.  
  
Bu şekilde Python, başka yerlerde daha verimli kullanım için ayrılabilecek geniş kaynakları serbest bırakır.  
  
Sistem Entegrasyonu için Ana Anahtar:  
  
Python, sistem entegrasyonu için benzersiz avantajlar sunar. Bir yandan, neredeyse tüm üçüncü taraf sistemlerin entegre edilebileceği çok sayıda Python kitaplığı vardır. Öte yandan, Python'da diğer birçok programlama dilinin kütüphaneleri de kullanılabilir.  
  
Programlandıktan sonra Python uygulamaları, bir Python yorumlayıcısının bulunduğu tüm işletim sistemlerinde çalışabilir ve işletim sistemine özgü uygulamaların maliyetini önemli ölçüde azaltır.  
  
Her Şeyi Değiştiren Dil:  
  
20 yılı aşkın bir süredir Python, dünya çapında endüstride, hizmet sektöründe ve ayrıca araştırma ve bilimde çok çeşitli farklı gereksinimleri karşılamak için bir programlama dili olarak başarıyla kullanılmaktadır. Bu süre zarfında, dil birçok şeyi değiştirdi.  
  
Python programlama dilini öğrenmesi kolaydır. Kullanıcılar ve geliştiriciler arasındaki sınırları bulanıklaştırdı. Artan sayıda bilim insanı, mühendis, finans uzmanı ve programlama deneyimi az olan diğerleri, belirli karmaşık teknik sorunları çözmek için Python kullanıyor.

Historical Development of Python

This programming language was developed in the early 90s by a Dutch programmer called Guido van Rossum. Most people think that this programming language is named after the python snake, assuming its name is Python.

However, contrary to the assumption, the name of this programming language does not come from the python snake. Guido van Rossum named this programming language inspired by the show of Monty Python’s Flying Circus from an English comedy group called The Monty Python.

| Rossum |
| --- |
| Guido van Rossum |

Even though this is the case, the logo of the Python programming language with a serpent type in many places has become almost a tradition.



This language has a huge group of developers around the world. If you have any problem, you can always ask other Python users/developers for help or find a suitable answer on several sites like [stackoverflow.com](https://stackoverflow.com/)

Python'un Tarihsel Gelişimi

Bu programlama dili 90'ların başında Guido van Rossum adlı Hollandalı bir programcı tarafından geliştirildi. Çoğu kişi, adının Python olduğunu varsayarak, bu programlama dilinin adını piton yılanından aldığını düşünür.  
  
Ancak sanılanın aksine bu programlama dilinin adı piton yılanından gelmiyor. Guido van Rossum, The Monty Python adlı bir İngiliz komedi grubundan Monty Python's Flying Circus şovundan esinlenerek bu programlama dili adını verdi.  
  
  
Guido van Rossum  
  
Durum böyle olsa da birçok yerde yılan türü olan Python programlama dilinin logosu adeta bir gelenek haline geldi.  
  
  
  
Bu dilin dünya çapında büyük bir geliştirici grubu var. Herhangi bir sorununuz varsa, her zaman diğer Python kullanıcılarından/geliştiricilerinden yardım isteyebilir veya stackoverflow.com gibi çeşitli sitelerde uygun bir yanıt bulabilirsiniz.

Review of Tools & Installations

The version of Python 3.X was released in 2008. It makes Python more readable and consistent than previous versions. Throughout this course, we will use the latest versions of Python 3.X (currently Python 3.9.7).

There are several Integrated Development Environments (IDE) that we can write and run Python codes on them. We prefer Anaconda Navigator (Anaconda3) package program which includes several IDEs options for you. You can optionally install Anaconda on your computer. Currently, Python 3.8 is available in the Anaconda package.

Among the IDEs in Anaconda, we prefer to use Jupyter Lab to write down the Python codes on it. The Jupyter Lab is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. Jupyter Notebook can also be used which is very similar to Jupyter Lab.

Apart from Anaconda, of course, Python Shell (IDLE) can also be used as a basic interpreter tool for Python.

Throughout this course;

We will provide you Playground module to write and run your code on each lesson page. Therefore, as we stated before you don't have to install Anaconda Package, it is optional.

Additionally, we will be using all Jupyter Notebook-based IDEs, including the [Google Colab](https://colab.research.google.com/) application, during in-class sessions. Actually, you don't necessarily need to install any tool to run your Python code at all. Google Colab allows you to "plug and play" Python on your browser.

You will have several Scratch exercises in accordance with Python codes. You can find an explanatory video below about how to solve the Scratch exercises.

Here is an alternative way to run your code. This alternative program is **Visual Studio Code (VS Code)** which is also available in the Anaconda Package. Although we prefer to use *Jupyter Lab* in Anaconda Package to run our codes, you can install and use the VS Code program either.

VS Code will be used in the **interactive complementary video lessons** you will receive throughout the course. You can watch the video below about the installation of Python and VS Code if you want.

Araçlar ve Kurulumların Gözden Geçirilmesi

Python 3.X sürümü 2008'de piyasaya sürüldü. Python'u önceki sürümlerden daha okunaklı ve tutarlı hale getiriyor. Bu kurs boyunca Python 3.X'in (şu anda Python 3.9.7) en son sürümlerini kullanacağız.  
  
Üzerine Python kodlarını yazıp çalıştırabileceğimiz birkaç Entegre Geliştirme Ortamı (IDE) vardır. Sizin için çeşitli IDE seçenekleri içeren Anaconda Navigator (Anaconda3) paket programını tercih ediyoruz. Anaconda'yı isteğe bağlı olarak bilgisayarınıza kurabilirsiniz. Şu anda Python 3.8, Anaconda paketinde mevcuttur.  
  
Anaconda'daki IDE'ler arasında Python kodlarını yazmak için Jupyter Lab'ı kullanmayı tercih ediyoruz. Jupyter Lab, canlı kod, denklemler, görselleştirmeler ve anlatım metni içeren belgeler oluşturmanıza ve paylaşmanıza olanak tanıyan açık kaynaklı bir web uygulamasıdır. Jupyter Lab'a çok benzeyen Jupyter Notebook da kullanılabilir.  
  
Anaconda dışında elbette Python Shell (IDLE), Python için temel bir yorumlayıcı aracı olarak da kullanılabilir.  
  
Bu kurs boyunca;  
  
Her ders sayfasında kodunuzu yazıp çalıştırmanız için size Playground modülü sağlayacağız. Bu nedenle daha önce de belirttiğimiz gibi Anaconda Paketi kurmanıza gerek yoktur, isteğe bağlıdır.  
  
Ayrıca, sınıf içi oturumlarda Google Colab uygulaması da dahil olmak üzere tüm Jupyter Notebook tabanlı IDE'leri kullanacağız. Aslında Python kodunuzu çalıştırmak için herhangi bir araç yüklemeniz gerekmez. Google Colab, tarayıcınızda Python'u 'tak ve çalıştır' yapmanızı sağlar.  
  
Python kodlarına uygun olarak birkaç Scratch alıştırmanız olacak. Aşağıda Scratch egzersizlerinin nasıl çözüleceği ile ilgili açıklayıcı bir video bulabilirsiniz.

İşte kodunuzu çalıştırmanın alternatif bir yolu. Bu alternatif program, Anaconda Paketinde de bulunan Visual Studio Code'dur (VS Code). Kodlarımızı çalıştırmak için Anaconda Paketinde Jupyter Lab kullanmayı tercih etsek de, VS Code programını da yükleyebilir ve kullanabilirsiniz.  
  
VS Code, kurs boyunca alacağınız etkileşimli tamamlayıcı video derslerinde kullanılacaktır. Dilerseniz Python ve VS Code kurulumu ile ilgili aşağıdaki videoyu izleyebilirsiniz.

FİRST STEP TO CODİNG

Introduction

In this lesson, we will show you how to write and develop a code outline with Python programming language. Although these initial codes are quite simple, they will give you great tips on what Python language is like.

Tanıtım  
Bu dersimizde size Python programlama dili ile nasıl kod taslağı yazıp geliştireceğinizi göstereceğiz. Bu ilk kodlar oldukça basit olsa da, Python dilinin nasıl olduğu konusunda size harika ipuçları verecekler.

First Program for 'Being a Good Person'

Let's write our first program. We will print 'being a good person' on the screen:

input :

print('being a good person')

output :

being a good person

As you see, it prints what you have written between single quotes like: 'text'. Here is another single line of code:

input :

print('clarusway will change my life')

output :

clarusway will change my life

Scratch Time ! : Solve this example with [scratch](https://scratch.mit.edu/projects/341646035/editor/).

We wrote clarusway will change my life using print() function. For now, you do not need to understand how this code runs, just start to enjoy the coding looking like the English language.

If we make some statements at this point, print is the name of a function. A function is a piece of code that executes some logic for you, e.g. prints a text or square a number.

In some cases, a function is a subprogram we produce which can be reused within your programs. If the name of a function is followed by parentheses, it means that it was called to get the result.

'İyi İnsan Olmak' için İlk Program  
İlk programımızı yazalım. Ekrana 'iyi bir insan olmak' yazacağız:  
  
Giriş:  
  
Print('iyi bir insan olmak')  
Çıktı :  
  
Iyi bir insan olmak  
Gördüğünüz gibi, yazdıklarınızı tek tırnak işaretleri arasında yazdırıyor: 'metin'. İşte başka bir tek kod satırı:  
  
Giriş:  
  
Print('clarusway hayatımı değiştirecek')  
Çıktı :  
  
Clarusway hayatımı değiştirecek  
Kazıma Zamanı! : Bu örneği sıfırdan çözün.  
  
  
Print() fonksiyonunu kullanarak clarusway'in hayatımı değiştireceğini yazdık. Şimdilik bu kodun nasıl çalıştığını anlamanıza gerek yok, sadece İngilizce gibi görünen kodlamanın keyfini çıkarmaya başlayın.  
  
Bu noktada bazı açıklamalar yapacak olursak print bir fonksiyonun adıdır. İşlev, sizin için bir mantık yürüten bir kod parçasıdır, ör. Bir metin veya bir sayının karesini yazdırır.  
  
Bazı durumlarda fonksiyon, programlarınızda yeniden kullanılabilecek ürettiğimiz bir alt programdır. Bir fonksiyonun adını parantezler takip ediyorsa, sonucu almak için çağrıldığı anlamına gelir.

Matter of Quotes

As a Python programmer, you will deal with quotes a lot. As you know, a string text is surrounded by a pair of quotes.

There are basically two types of quotes we use. Single or double quotes. Both are the same but we should use them in the correct way:

Use double quotes if your string includes the single one:e.g. "It's my pleasure!"

Use single quotes if your string includes the double one:e.g. 'He said: "I am done" and fell down.'

Use triple quotes if your string is too long which composed of multiple lines : e.g. :

'''

...long string...

..long string..

'''

Here's an example that shows using double-quotes. Note that, the text (it's not a problem) below includes a single quote '.

input :

print("it's not a problem")

output :

it's not a problem

Here is a new single-line program too. In this case, be careful with double-quotes. It is the same as single quotes. Even triple quotes can be used for the same result. Try to guess output of this code : print('''it's not a problem using "triple" quotes''').

⚠️Avoid ! :

Do not mix two quotes style in one string.

print('We should have enough time for our family") This code gives an error because it starts with single quotes ends with double. See below:

input :

print('We should have enough time for our family")

output :

File "", line 1

print('We should have enough time for our family")

^

SyntaxError: EOL while scanning string literal

Let's do the same with numbers:

input :

print(572)

output :

572

input :

print(3.14)

output :

3.14

input :

print('3.14')

output :

3.14

💡Tips:

If you have noticed we used 👉🏻. not 👉🏻, for the decimal number: 3.14.

Surrounding the expression with quotes makes it in string type. We will see immutable types (int, string, tuple, etc.) in the next lessons.

We assumed that you have learned how to write simple Python code consisting of a single line which just prints a text. But, the real-world codes of Python contain a huge number of lines. So, now we will write codes that print multiple lines.

Let's look at the example below. And this is our first multi-line code. The following code prints three lines:

input :

print('no pain')

print('no gain')

print(3.14)

output :

no pain

no gain

3.14

If you need an empty line, you can use only print() function. Let's try :

input :

print('first line')

print() # second line is empty

print('third line')

output :

first line

third line

Alıntı Konusu

Bir Python programcısı olarak alıntılarla çok uğraşacaksınız. Bildiğiniz gibi, bir dize metni bir çift tırnak içine alınır.  
  
Temel olarak kullandığımız iki tür alıntı vardır. Tek veya çift tırnak. İkisi de aynı ama biz bunları doğru şekilde kullanmalıyız:  
  
Dizeniz tek olanı içeriyorsa çift tırnak kullanın: ör. 'Benim için bir zevk!'  
  
Dizeniz çift olanı içeriyorsa tek tırnak kullanın: ör. 'İşim bitti' dedi ve düştü.'  
  
Dizeniz çok uzunsa ve birden çok satırdan oluşuyorsa üçlü tırnak kullanın: ör. :  
  
'''  
...uzun dize...  
..uzun dize..  
'''  
İşte çift tırnak kullanmayı gösteren bir örnek. Aşağıdaki metnin (sorun değil) tek bir alıntı içerdiğini unutmayın '.  
  
Giriş:  
  
Print('sorun değil')  
Çıktı :  
  
Problem değil  
Burada da yeni bir tek satırlık program var. Bu durumda, çift tırnaklara dikkat edin. Tek tırnak ile aynıdır. Aynı sonuç için üçlü tırnak bile kullanılabilir. Bu kodun çıktısını tahmin etmeye çalışın: print('''üçlü' tırnak kullanmak sorun değil''').  
  
⚠️ Kaçının ! :  
Bir dizede iki tırnak stilini karıştırmayın.  
Print('Ailemize yeterince zaman ayırmalıyız') Bu kod tek tırnak ile başlayıp çift ile bittiği için hata veriyor.Aşağıya bakınız:  
  
Giriş:  
  
Print('Ailemize yeterince zaman ayırmalıyız')  
Çıktı :  
  
Dosya '', satır 1  
Print('Ailemize yeterince zaman ayırmalıyız')  
  
  
SyntaxError: Dize değişmezi taranırken EOL  
Aynısını sayılarla da yapalım:  
  
Giriş:  
  
Yazdır(572)  
Çıktı :  
  
572  
Giriş:  
  
Yazdır(3.14)  
Çıktı :  
  
3.14  
Giriş:  
  
Yazdır('3.14')  
Çıktı :  
  
3.14  
💡İpuçları:  
Fark ettiyseniz 👉🏻 kullandık. 👉🏻 değil, ondalık sayı için: 3.14.  
İfadeyi tırnak işaretleri içine almak onu dize türünde yapar. İlerleyen derslerde değişmez türleri (int, string, tuple vb.) göreceğiz.  
Sadece bir metin basan tek bir satırdan oluşan basit Python kodunun nasıl yazılacağını öğrendiğinizi varsaydık. Ancak Python'un gerçek dünya kodları çok sayıda satır içerir. Şimdi birden çok satır basan kodlar yazacağız.  
  
Aşağıdaki örneğe bakalım. Ve bu bizim ilk çok satırlı kodumuz. Aşağıdaki kod üç satır yazdırır:  
  
Giriş:  
  
Print('acı yok')  
Print('kazanç yok')  
Yazdır(3.14)  
Çıktı :  
  
Acı yok  
Kazanç yok  
3.14  
Boş bir satıra ihtiyacınız varsa, yalnızca print() işlevini kullanabilirsiniz. Hadi deneyelim :  
  
Giriş:  
  
Print('ilk satır')  
Print() # ikinci satır boş  
Print('üçüncü satır')  
Çıktı :  
  
İlk satır  
  
Üçüncü satır