

Python ile Temel Düzey Programlama Eğitimi

Murat Uğur KİRAZ
Online Python Eğitimi



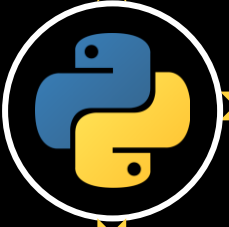
Murat Uğur KİRAZ

Siber Güvenlik Analisti / Developer

- Milli Savunma Üniversitesi Bilgisayar Mühendisliği Siber Güvenlik Programı – Tezli Yüksek Lisans
- 10 senelik tecrübe
- Web Uygulama Geliştirme ve Web Güvenliği



Tanışma



- **Eğitmenin Kendini Tanıtması**
- Katılımcıların Kendini Tanıtması
 - Adınız ve soyadınız
 - Çalıştığınız iş tanımı
 - Geçmiş iş tecrübesi
 - Bilgisayar Programlama ve Python ile ilgili Bildikleriniz
 - Eğitime katılma amacınız
 - Eğitimden beklentileriniz

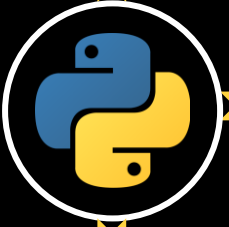
Python Nedir?



- Python, genel amaçlı programlama için kullanılan, yaygın olarak kullanılan, yorumlanan, nesne yönelimli ve dinamik anlambilime sahip üst düzey bir programlama dilidir.
- Monty Python's Flying Circus
- Guido van Rossum tarafından yaratıldı



Python Nedir?



- büyük rakiplerinki kadar güçlü, kolay ve sezgisel bir dil;
- açık kaynak, böylece herkes gelişimine katkıda bulunabilir;
- sade İngilizce kadar anlaşılır kod;
- Kısa geliştirme sürelerine izin vererek günlük görevlere uygundur.

Python Nedir?



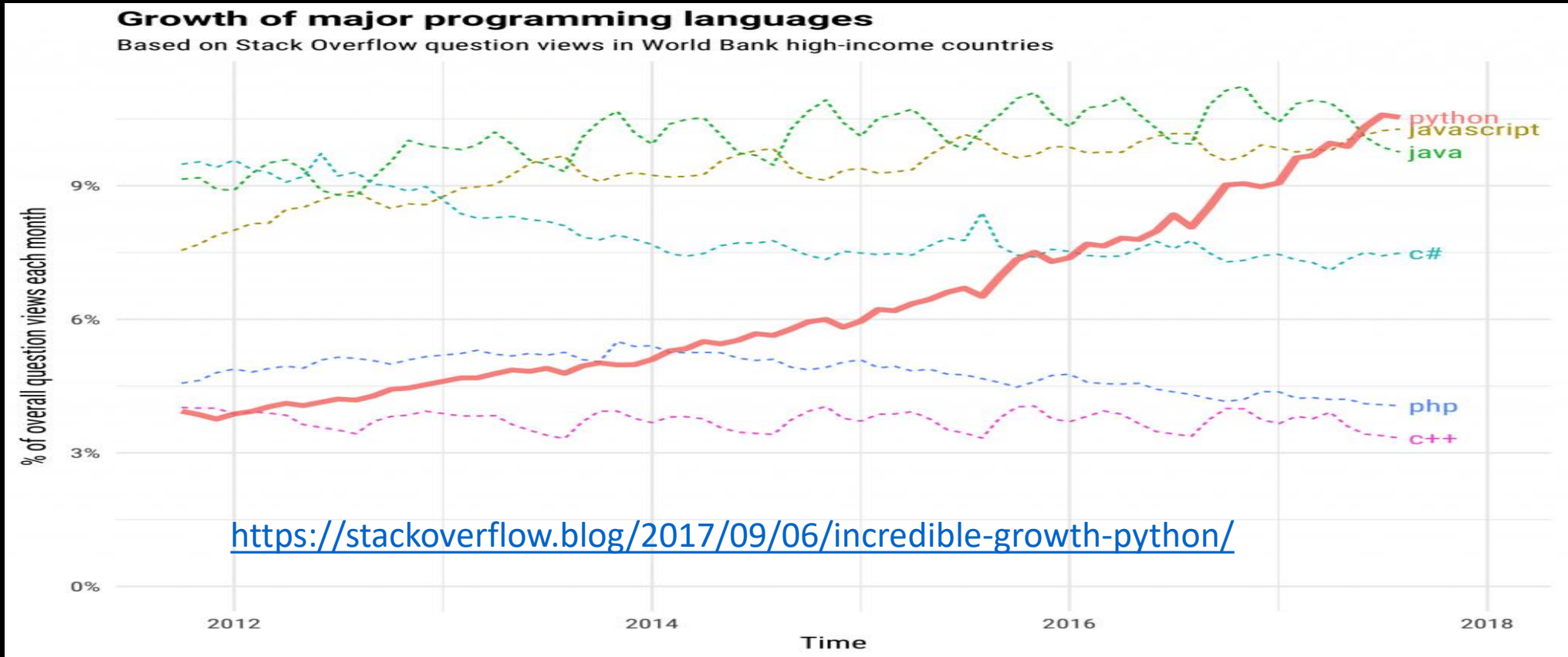
- Python;
 - Web uygulamaları,
 - Yazılım geliştirme,
 - Veri bilimi ve makine öğreniminde (ML) yaygın olarak kullanılan bir programlama dilidir.
- Geliştiriciler, etkili ve öğrenmesi kolay olduğu ve birçok farklı platformda çalıştırılabildiği için Python'ı kullanır.
- Python yazılımı ücretsiz olarak indirilebilir, her türlü sistemle iyi bir entegrasyon sağlar ve geliştirme hızını artırır.

Python'ın avantajları nelerdir?



- Python, okunabilir ve üretkendir.
- Python, yeniden kullanılabilir kodlar içerir.
- Python, diğer dillerle uyumludur.
- Python, destekleyici bir topluluğa sahiptir.
- Python, öğrenmesi kolaydır.
- Python, taşınabilirdir.

Neden Python?



Neden Python?



- Yorumlanan bir dil
- Kullanımı kolay dil
- Dinamik olarak yazılan bir dil
- Üst düzey bir dil
- Nesne odaklı bir dil

learn
Python
after C++



learn C++
after
Python

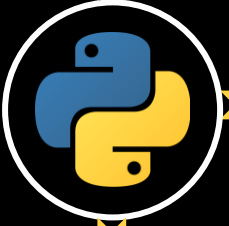


Python'un Rakipleri



- Perl
- Ruby

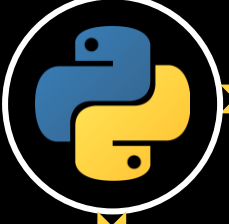
Neden Python Değil?



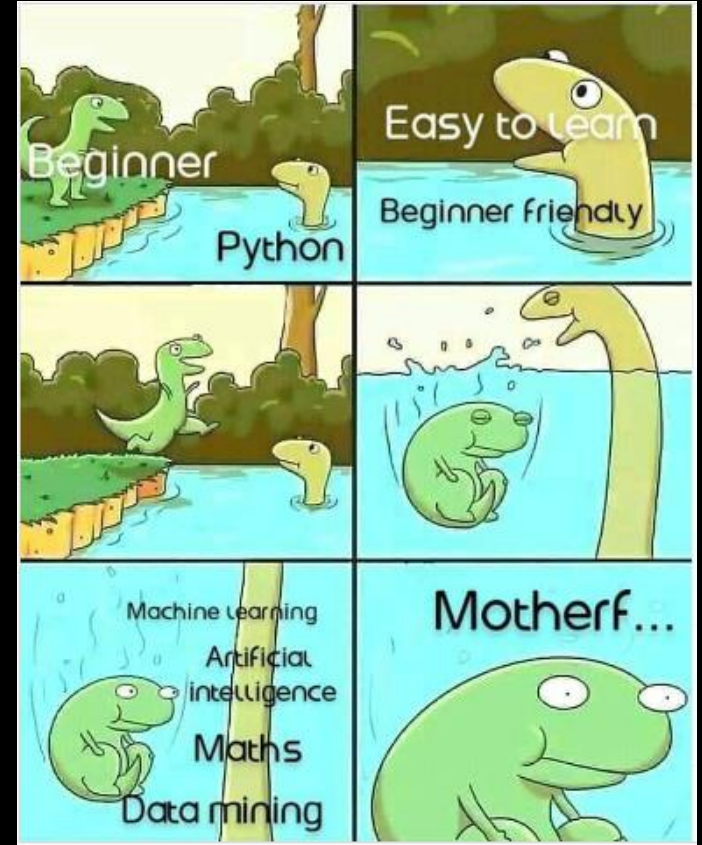
- RAM'i ve işlemciyi ekonomik kullanmaz!!!
- Diğer programlama dillerine göre çok enerji harcar.
- [Link](#)
- Düşük Seviyeli Programlama
- Mobil Cihaz Uygulamaları

	Energy
(c) C	1.00
(c) Rust	1.03
(c) C++	1.34
(c) Ada	1.70
(v) Java	1.98
(c) Pascal	2.14
(c) Chapel	2.18
(v) Lisp	2.27
(c) Ocaml	2.40
(c) Fortran	2.52
(c) Swift	2.79
(c) Haskell	3.10
(v) C#	3.14
(c) Go	3.23
(i) Dart	3.83
(v) F#	4.13
(i) JavaScript	4.45
(v) Racket	7.91
(i) TypeScript	21.50
(i) Hack	24.02
(i) PHP	29.30
(v) Erlang	42.23
(i) Lua	45.98
(i) Jruby	46.54
(i) Ruby	69.91
(i) Python	75.88
(i) Perl	79.58

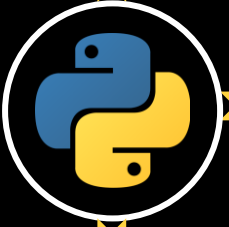
Python Nerelerde Kullanılır?



- Sunucu tarafında web geliştirme
- Python betikleriyle otomasyon
- Veri bilimi ve makine öğrenimi
- Yazılım geliştirme
- Yazılım testi otomasyonu



Sunucu tarafında web geliştirme



- Web sitelerinin arka uç işlevlerini içerir.
- Veritabanı işlemleri, diğer web siteleriyle iletişim ve veri güvenliği yer alır.
- Python, bu işlevler için önceden yazılmış kod içeren birçok kitaplık sunar.
- Python ayrıca web uygulamalarını oluşturmak için çeşitli çerçeveler sunar.
- Bu çerçeveler, geliştiricilerin web uygulamalarını daha hızlı ve daha kolay oluşturmalarına yardımcı olur.

Python betikleriyle otomasyon



- Çok sayıda dosyayı tek seferde yeniden adlandırma
- Bir dosyayı başka bir dosya türüne dönüştürme
- Bir metin dosyasındaki tekrarlayan kelimeleri silme
- Temel matematiksel işlemleri gerçekleştirme
- E-posta iletileri gönderme
- İçerik indirme
- Temel günlük analizi gerçekleştirme
- Birçok dosyada hata bulma

Veri bilimi ve makine öğrenimi



- Veri temizleme olarak bilinen, yanlış verileri düzeltme ve kaldırma
- Verileri ayıklama ve verilerin çeşitli özelliklerini seçme
- Veri etiketleme, yani veriler için anlamlı adlar ekleme
- Verilerden farklı öngörüler çıkarma
- Çizgi grafikler, çubuk grafikler, sütun grafikler ve dilim grafikler gibi grafikleri kullanarak verileri görselleştirme

Yazılım geliştirme



- Yazılım kodundaki hataları izleme
- Yazılımı otomatik olarak derleme
- Yazılım proje yönetimini gerçekleştirme
- Yazılım prototipleri geliştirme
- Grafik Kullanıcı Arabirimi (GUI) kitaplıklarını kullanarak masaüstü uygulamaları geliştirme
- Basit metin tabanlı oyunlardan daha karmaşık video oyunlarına kadar çeşitli oyunlar geliştirme

Yazılım testi otomasyonu



- Geliştiriciler, yazdıkları işlevleri test etmek için unittest, Robot ve PyUnit gibi Python birim testi çerçevelerini kullanır.
- Yazılım testinden sorumlu kişiler, çeşitli test senaryoları için test örnek olayları yazmak için Python'ı kullanır. Örneğin, Python'ı bir web uygulamasının, birden çok yazılım bileşeninin ve yeni özelliklerin kullanıcı arabirimini test etmek için kullanırlar.

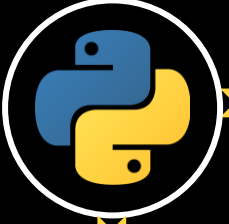
Python Türevleri



- Python2
- Python3
- Python veya Cpython
- Cython
- Jython
- PyPy ve RPython



Bu kurs size ne kazandıracak?



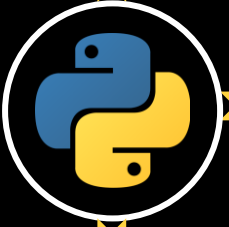
- Programlamanın temellerini öğreneceksiniz.
- Python temelli programlamalar yapabileceksiniz.
- Kod okuma becerileriniz gelişecek.
- Nesne tabanlı programlama temellerini aldığınız için C#, asp.net gibi dilleri kolay öğreneceksiniz.
- Machine learning, Artificial Intelligence gibi alanlarda çalışabileceksiniz.
- CV'nize 'Python Biliyorum' cümlesini ekleyeceksiniz.

Python kitaplıkları nelerdir?



- Kitaplık Nedir? => Koleksiyon!
- Python standart kitaplıklarla sunulur.
- 137.000'den fazla kitaplık!
 - Matplotlib
 - Pandas
 - NumPy
 - Requests
 - OpenCV-Python
 - Keras

Python çerçeveleri nelerdir?



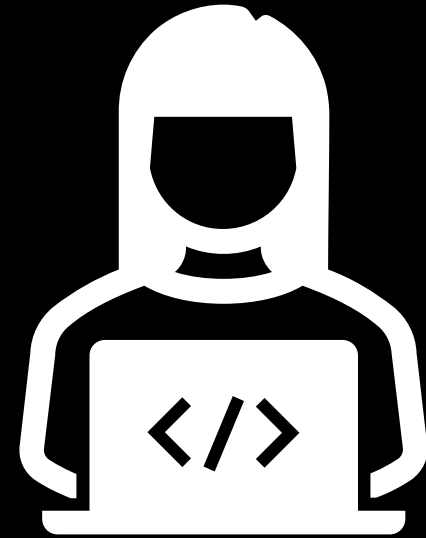
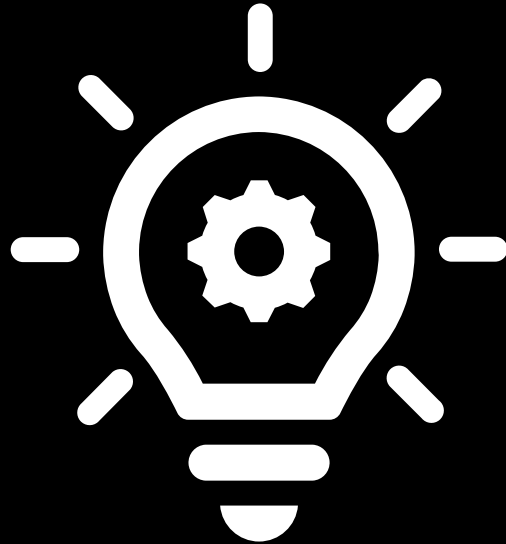
- Python çerçevesi, paketlerden ve modüllerden oluşan bir koleksiyondur.
- Modül, ilgili kodlardan oluşan bir kümeyken, paket ise modüllerden oluşan bir kümedir.
 - Django
 - Flask
 - TurboGears
 - Apache MXNet
 - PyTorch

Neler Öğreneceğiz?

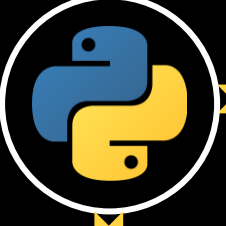


- Python'a Giriş
- Türler ve Değişkenler
- Temel İfadeler
- Fonksiyonlar
- Dosyalar ile Çalışmak
- Exceptions (İstisnalar)
- Modüller ve Paketler
- Sınıflar

Sizden Beklentilerimiz



Nasıl Çalışmalıyım?



- Dersten önce size sunulan dokümanları gözden geçirin.
- Ders bittikten sonra derste işlenen konuları kendiniz yapın.
- Dersten sonra size verilen ödev ve projeleri yapın.
- Aklınızdan geçen her şeyi koda dökmeye çalışın



Nasıl Çalışmalıyım?



- Algoritma yeteneğinizi geliştirin.
- Github, gitlab gibi repolara üye olun.
- Matematik bilginizi artırın. Kodlama=Matematik!
- Derste anlamadığınız yerleri mutlaka sorun, fakat kendi kendinize çalışırken yapamadığınız yerde internetten araştırma yapın. Kendinizi zorlayın. Hiçbir çözüm bulamadıysanız o zaman sorun.