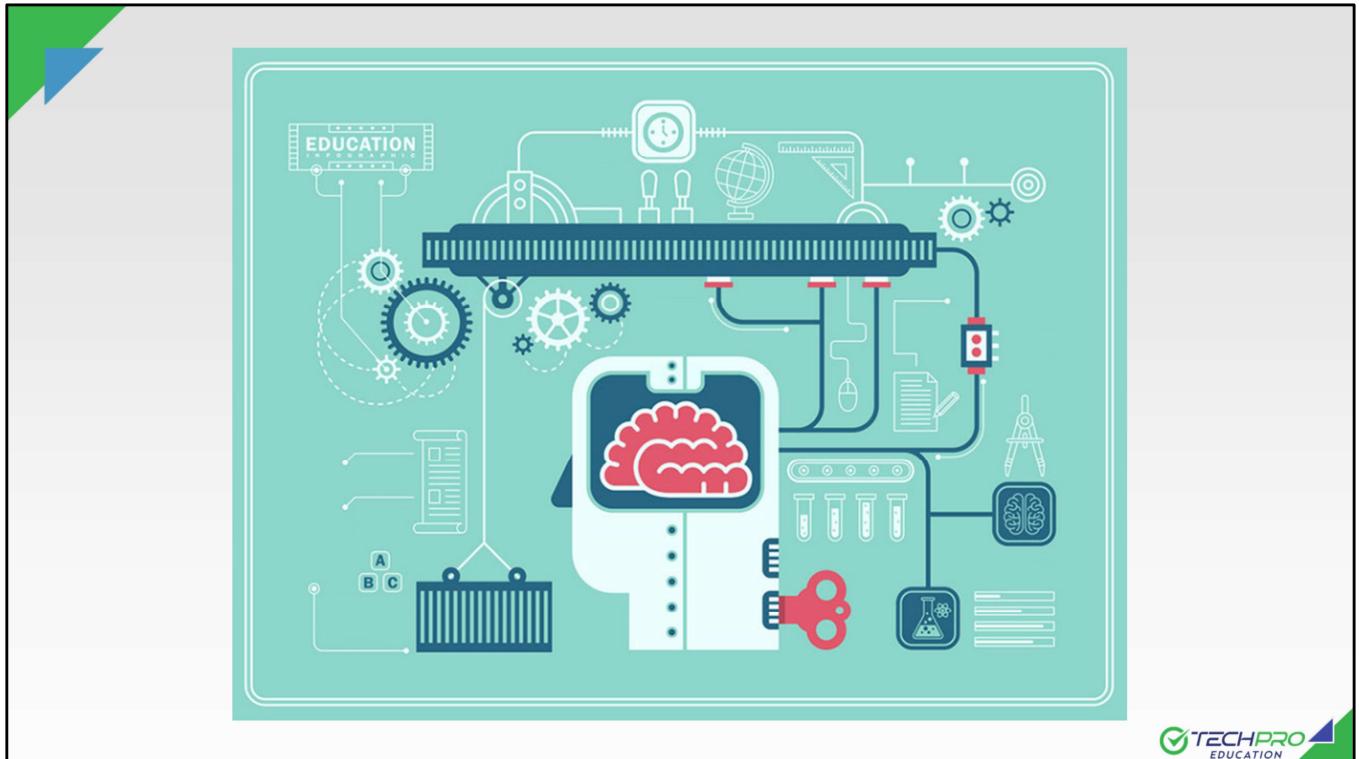




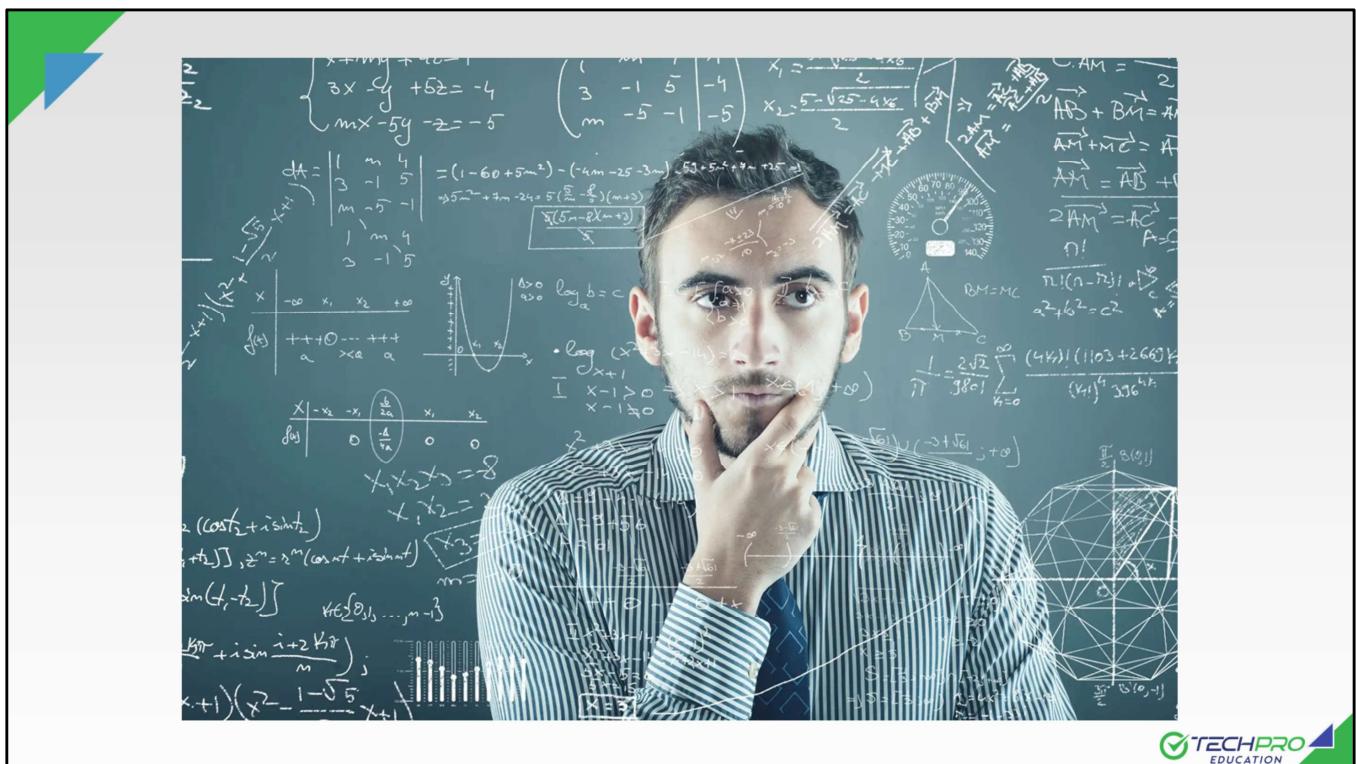
FLASK





Merhaba! Bugün size, veri bilimcileri olarak çeşitli veri analizleri ve modellemeler yapmış olduğunuz bilgi ve becerilerinizi nasıl daha geniş kitlelere ulaştıracığınızden bahsedeceğim.

Hikayemizin kahramanı, genç bir veri bilimcisi olan Ali'dir.





Bir gün, Ali'nin aklına bir fikir geldi. Veri bilimcilerin projelerini daha geniş bir kitleye nasıl ulaştıracaklarını araştırmaya başladı.

Araştırmaları sonucunda, Flask adında bir Python web uygulama çerçevesini keşfetti.

Flask, veri bilimcilerinin verileri etkileşimli web uygulamalarına dönüşturmelerine ve çevrimiçi olarak paylaşmalarına olanak tanır.



Ali, Flask kullanarak veri analizlerini ve makine öğrenme modellerini interaktif bir web uygulaması haline getirmeye karar verdi.

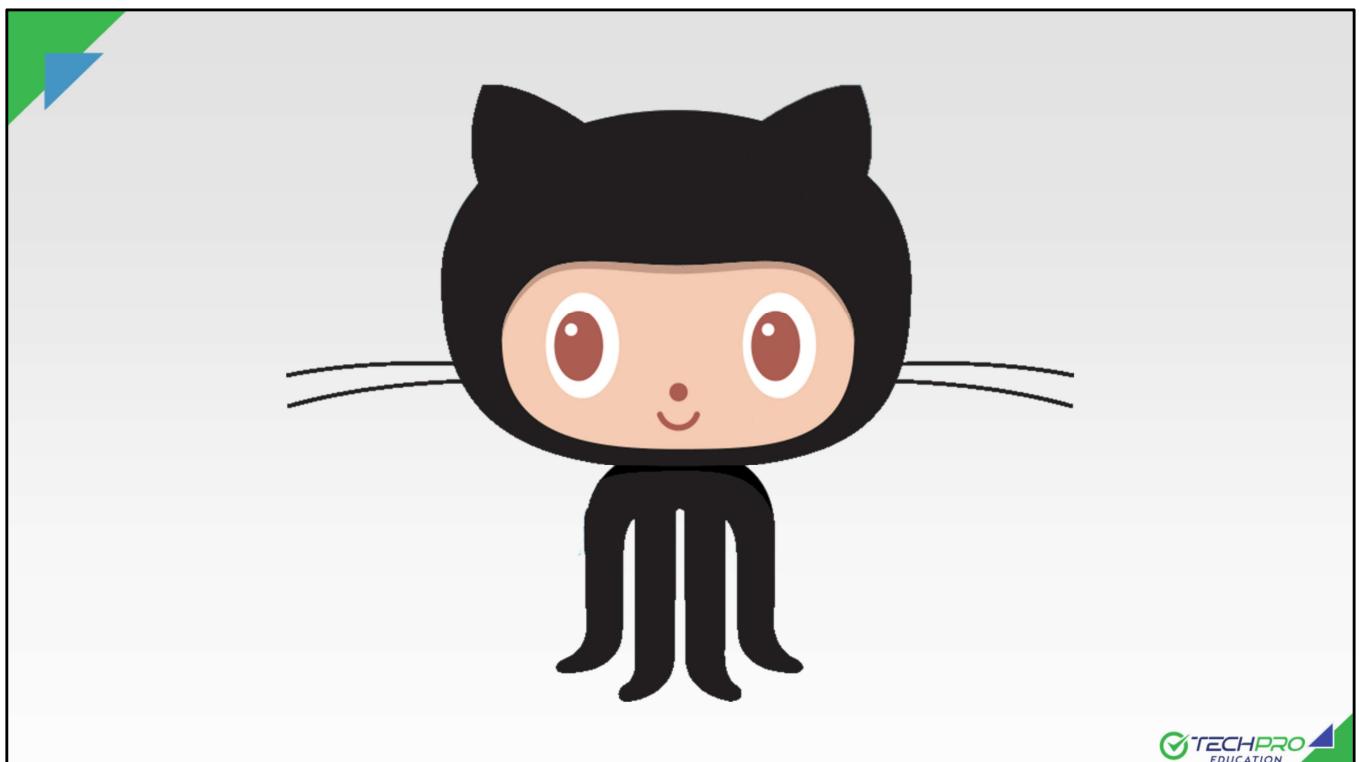
İlk olarak, basit bir web uygulaması oluşturdu ve bu uygulamaya verilerini görselleştirmek için grafikler ekledi.

Daha sonra, makine öğrenme modelini bu uygulamaya entegre etti, böylece kullanıcılar verileri kendileri tahmin edebilirlerdi.



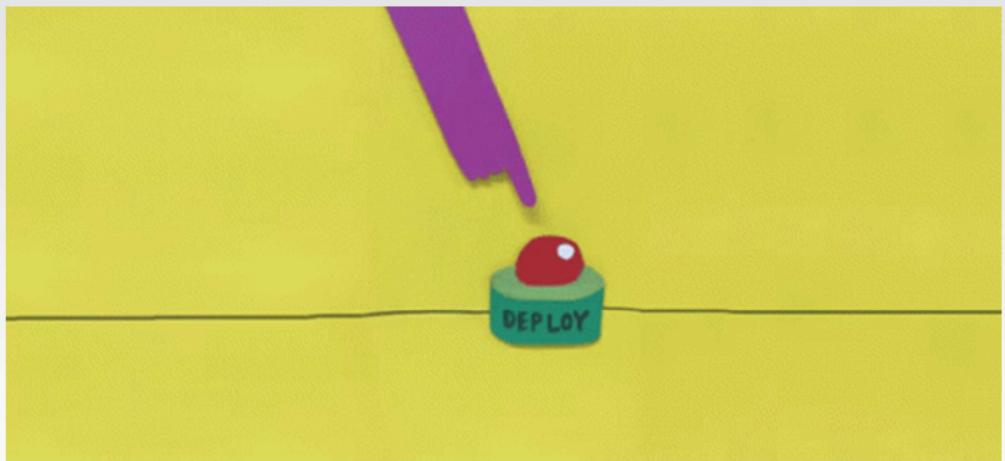
Ali'nin Flask uygulaması tamamlandığında, bu uygulamayı şirketin iç ağına yayınladı. Artık şirketteki herkes, verileri kolayca görselleştirebilir ve analiz edebilirdi.

Bu, kararlarını daha iyi desteklemelerine yardımcı oldu.



Ancak Ali'nin işi burada bitmedi. Kendi kişisel projelerini de Flask ile hayatı geçirdi ve bu projeleri GitHub ve benzeri platformlarda paylaştı.

Böylece, dünya genelindeki diğer veri bilimcileri ve ilgilenen kişiler, onun çalışmalarından öğrenip faydalananabilirlerdi.



Sonuç olarak, Flask kullanarak veri bilimcileri, projelerini daha erişilebilir ve etkileşimli hale getirebilirler.

Bu sayede, veri analizi ve makine öğrenme dünyasındaki katkıları daha geniş bir kitleye ulaşabilir ve bu değerli bilgiyi paylaşabilirler.

FLASK ile Deployment İçerik

- Framework Nedir? Flask Nedir?
- Flask ile Web Uygulamaları Neden Önemli?
- Veri Bilimi ve Flask Nasıl Birleştirilir?
- Bir Flask Uygulaması Hangi Adımlardan Oluşur?
- Deployment Nedir ve Neden Önemlidir?





FLASK ile Deployment İçerik

- Hangi Projeler Flask ile Yapılabilir?
- Başarılı Bir Flask Uygulaması Örneği
- Flask Eğitiminden Ne Bekliyorsunuz?



Framework Nedir? Flask Nedir?



Frame Work nedir: Bu, geliştiricilerin uygulama tasarımları, veritabanı entegrasyonu, güvenlik, oturum yönetimi, kullanıcı arayüzü, yönetim paneli ve diğer birçok konuda kod yazmaya başlamadan önce bu yapıları kullanmalarına olanak tanır.

Örnek verecek olursak Düşünün ki bir ev yapmak istiyorsunuz. Framework, size evin temel bir iskeletini sağlar. İskelet, evin temel çerçevesidir ve neyin nereye yerleşeceğini belirler. Framework de yazılım geliştirirken aynı işi yapar.

Framework, geliştiricilerin yazılım projelerini daha hızlı ve düzenli bir şekilde yapmalarını sağlar.

Bir projeyi baştan sona yaparken birçok görev tekrar edebilir. Framework, bu tekrar eden görevleri otomatikleştirerek zaman ve çaba tasarrufu sağlar.

Örneğin, bir web sitesi oluşturmak istedığınızı düşünelim. Framework, size birçok ön tanımlı bileşen ve araçlar sunar. Bu bileşenler, web sitesinin tasarımını, kullanıcıların giriş yapması için form oluşturmayı, veritabanına bağlanmayı ve diğer görevleri kolaylaştırır.

Framework aynı zamanda geliştiriciler arasında işbirliğini de kolaylaştırır.

Örneğin, bir grup bir projede çalışıiyorsa, her öğrenci bir görev alabilir ve framework içindeki yapıya göre çalışabilir. Bu şekilde, herkes projeye uyumlu bir şekilde katkıda bulunabilir.

Framework'ler, geliştirme sürecinde hataları azaltır ve yazılımı daha güvenli hale getirir. Çünkü framework'ler, güvenlik önlemleri ve en iyi uygulamaları içerir. Bu da yazılımın daha güvenli ve sorunsuz çalışmasını sağlar.

Sonuç olarak, framework'ler, yazılım geliştirme sürecini daha hızlı, düzenli ve verimli hale getiren yapılar olarak düşünübilirsiniz. Framework'leri kullanarak, yazılım projelerinizi daha kolay yönetebilir, daha az hata yapabilir ve daha iyi bir şekilde işbirliği yapabilirsiniz.

- Django: Python tabanlı bir framework. Web uygulamaları geliştirmek için kullanılır.
- Ruby on Rails: Ruby tabanlı bir framework. MVC (Model-View-Controller) mimarisine dayalı web uygulamaları geliştirmek için kullanılır.
- Laravel: PHP tabanlı bir framework. Web uygulamaları geliştirmek için kullanılır.
- Angular: TypeScript tabanlı bir framework. Ön uç web uygulamaları geliştirmek için kullanılır.
- React: JavaScript tabanlı bir kütüphane. Ön uç web uygulamaları geliştirmek için kullanılır.

Flask, Python programlama dilinde geliştirilen ve web uygulamaları oluşturmak için kullanılan bir mikro web çerçevesidir.

Flask, minimal ve hafif bir yapıyla bilinir, bu nedenle temel özellikleri sunar, ancak projenize ihtiyaca göre özelleştirmenize olanak tanır.

Flask, web uygulamalarını hızlı bir şekilde oluşturmanıza ve geliştirmenize yardımcı olan bir dizi araç ve kütüphane içerir.

Framework Nedir? Flask Nedir?

- Minimal ve Hafif Yapı
- Jinja2 Şablon Motoru
- URL Yönlendirme
- WSGI Uyumlu
- Veri Tabanı Desteği
- Genişletilebilirlik
- Microframework Olması



Flask'in temel özellikleri şunlar içerir:

Minimal ve Hafif Yapı: Flask, temel işlevleri sunar, ancak gereksiz karmaşıklıklardan kaçınır. Bu, projelerinizi hızlı bir şekilde başlatmanıza olanak tanır.

Jinja2 Şablon Motoru: Jinja2 şablon motoru sayesinde, web sayfalarını dinamik olarak oluşturabilirsiniz. Bu, web uygulamanızın kullanıcı arayüzüni oluşturmanızı sağlar.

URL Yönlendirme: URL yönlendirmesi, web uygulamanızdaki farklı sayfaları ve işlevleri kolayca yönetmenizi sağlar.

WSGI Uyumlu: Flask, Python Web Server Gateway Interface (WSGI) ile uyumludur, bu da onu çeşitli web sunucularında çalıştırabilmenize olanak tanır.

Veri Tabanı Desteği: Flask, SQLAlchemy gibi popüler veritabanı kütüphaneleriyle entegrasyonu destekler, bu da veri tabanına erişim sağlar.

Genişletilebilirlik: Flask, eklentiler (extensions) kullanarak ek işlevselligi kolayca eklemenizi sağlar. Örneğin, oturum yönetimi, form işleme veya OAuth kimlik doğrulama gibi eklentileri kullanabilirsiniz.

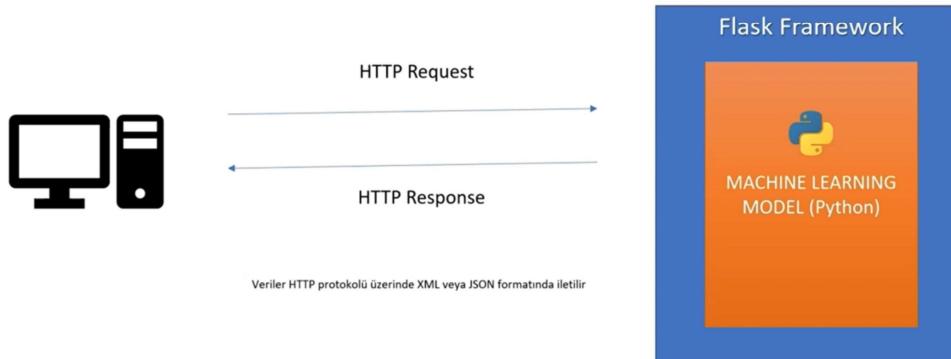
Microframework Olarak: Flask, bir mikro web çerçevesi olarak kabul edilir, bu da size

temel yapı taşlarını sunar, ancak diğer özellikleri projenizin gereksinimlerine göre eklemenize izin verir.

Flask, özellikle hızlı prototip oluşturma, RESTful API'ler geliştirme ve basit web uygulamaları oluşturma gibi kullanım senaryoları için popülerdir. Python diline aşina olanlar için Flask, web geliştirme becerilerini veri bilimi projeleri ile birleştirmek isteyenler için iyi bir seçenek olabilir.

Framework Nedir? Flask Nedir?

FLASK REST API ve Python ile Machine Learning Web API Tasarımı



REST: (REpresentational State Transfer)



Burada amacımız yaptığımz bir algoritmayı http protokolü üzerinden kullanıcıya sunacağz ama basit bir şekilde . Demem o ki örneğin bir kişi kan tahlillerini sizin hazırladığınız bu uygulama üzerinden ister mobil olsun ister web tabanlı olsun bunun içine girip prediction alacak.

Kısaca öğrendiğimiz ML modellerini deploy etmenin bir başka kolay yolu. Bunun için size öncelikle ihtiyacımız olan bir micro frame work olan flask i öğreneceğiz.

Bizim yazdığımız Python programları flask üzerinde çalışabiliyor. Flask inde bir özelşliği http protokolü üzerinde çalışması. HTTP işin içine girince ne oluyor biz dünyaya açılabiliyoruz.

Nasıl çalışır HTTP bir tane Client (klaynt) gönderir.

Falsk iniz bu değerlendirir buna bir cevap hazırlar.

Bu cevabı da HTTP Response objesi olarak geri Client (klaynt) gönderir.

Client (klaynt) da bunu görüntüler.

Veriler HTTP protokolü üzerinde XML veya JSON formatında iletiliyor. Burada JSON daha

popüler bunu kullanacağız.

JSON (JavaScript Object Notation-JavaScript Nesne Gösterimi(biçimi)'), veri alışverişi için kullanılan bir veri biçimidir. JSON, hafif ve okunabilir bir yapıya sahip olan metin tabanlı bir format olarak tasarlanmıştır. Genellikle web uygulamalarında veri aktarımı için kullanılır.

JSON'un temel amacı neydi, farklı platformlar arasında veri alışverişini kolaylaştırmaktır. Örneğin, bir web tarayıcısı bir sunucudan veri alırken veya bir mobil uygulama bir API'den veri çekerken, veriler genellikle JSON formatında iletilir.

Yani JSON üzerinden Yapay zeka istek requestimizi alacak. Predictionumuzun sonucunu da JSON formatında döndürücez Klaynta.

Bunlar Windows olur mac olur mobil olur ios Android olur hiç farketmez ama biz klayntımızı python dilince yazacağız.

Server da burada flask frame work üzerinden çalışan ML modelimiz.

Flask ile Web Uygulamaları Neden Önemli?

- Erişilebilirlik
- Veri Görselleştirmesi
- Kullanıcı Etkileşimi
- Çevrimiçi Veri Paylaşımı
- Gerçek Zamanlı Güncellemeler
- Kolay Dağıtım
- İş Analitiği ve Karar Destek
- Veri Güncellemesi ve Bakım Kolaylığı



Flask ile web uygulamaları geliştirmenin önemi birkaç açıdan öne çıkar:

Erişilebilirlik: Web uygulamaları, kullanıcıların çeşitli cihazlardan (bilgisayarlar, akıllı telefonlar, tabletler) ve farklı platformlardan (Windows, macOS, Linux) erişebileceği bir ortam sunar. Bu, projelerinizi daha geniş bir kitleye ulaştırmanıza olanak tanır.

Veri Görselleştirmesi: Web uygulamaları, verilerinizi grafikler, tablolar ve interaktif görsellerle sunmanıza olanak tanır. Bu, karmaşık verilerin daha anlaşılır hale gelmesini sağlar ve kullanıcıların verileri daha iyi anlamalarına yardımcı olur.

Kullanıcı Etkileşimi: Web uygulamaları, kullanıcıların verilerle etkileşimde bulunmasını sağlar. Kullanıcılar verileri sorgulayabilir, filtreleyebilir, analiz edebilir ve sonuçları kişiselleştirebilirler.

Çevrimiçi Veri Paylaşımı: Web uygulamaları, veri bilimi projelerinizi, sonuçlarınızı ve bulgularınızı çevrimiçi olarak paylaşmanıza olanak tanır. Bu, projelerinizin daha fazla kişiye fayda sağlamasına ve işbirliği yapmanızı olanak tanır.

Gerçek Zamanlı Güncellemeler: Web uygulamaları, kullanıcıların gerçek zamanlı olarak verilere erişmelerini ve güncellemeleri takip etmelerini sağlar. Bu özellik, dinamik ve canlı bir kullanıcı deneyimi sunar.

Kolay Dağıtım: Flask gibi web çerçeveleri, web uygulamalarını hızlı bir şekilde dağıtmانıza olanak tanır. Bu, projelerinizi hızla kullanılabılır hale getirerek zaman tasarrufu sağlar.

İş Analitiği ve Karar Destek: Web uygulamaları, iş analistleri ve yöneticiler için karar verme süreçlerini destekler. Veri analizi sonuçlarını hızlıca görüntüleyerek, iş stratejilerini

belirlemekte yardımcı olabilir.

Veri Güncellemesi ve Bakım Kolaylığı: Web uygulamaları, veri güncellemelerini ve bakımı kolaylaştırır. Veri bilimi projeleri, yeni verilerle sürekli olarak güncellenmesi gerekebilir, bu nedenle web uygulamaları bu süreci daha yönetilebilir hale getirir.

Sonuç olarak, Flask ile web uygulamaları geliştirmek, veri bilimi projelerinizi daha kullanılır, erişilebilir ve etkileşimli hale getirerek sonuçlarınızı daha geniş bir kitleye ulaştırmanızı ve verileri daha iyi anlamانıza yardımcı olur. Bu nedenle, veri bilimi alanında çalışan birçok profesyonel ve organizasyon, Flask ve benzeri web çerçevelerini kullanarak veri tabanlı projelerini daha da güçlendirir.

Veri Bilimi ve Flask Nasıl Birleştirilir?

- Web Uygulaması Arayüzü Oluşturma
- Makine Öğrenme Model Dağıtıımı
- Veri Görselleştirmesi
- İnteraktif Analiz Araçları
- Dashboard Oluşturma



Veri bilimi ve Flask, veri odaklı projeleri daha geniş bir kitleye ulaşımak ve veri analizi sonuçlarını etkileşimli hale getirmek için mükemmel bir şekilde birleştirilebilir. İşte veri bilimi ve Flask'in nasıl birleştirileceği konusunda bazı ana noktalar:

Web Uygulaması Arayüzü Oluşturma: Flask, veri bilimcilerine, veri analizi sonuçlarını web uygulamaları üzerinden görselleştirmek ve paylaşmak için bir platform sunar. Veri tabloları, grafikler, haritalar ve diğer görsellerle verileri daha anlaşılır hale getirebilirsiniz.

Makine Öğrenme Model Dağıtıımı: Flask, eğitilmiş makine öğrenme modellerini web uygulamalarına entegre etmenizi sağlar. Bu, kullanıcıların modelleri gerçek zamanlı olarak kullanarak tahminlerde bulunmasına olanak tanır.

Veri Görselleştirmesi: Flask ile, veri analizi sonuçlarınızı görsel olarak anlatmak için çeşitli grafik kütüphanelerini (örneğin, Matplotlib veya Plotly) kullanabilirsiniz. Kullanıcılar verilere grafikler üzerinden erişebilir ve verileri daha iyi anlayabilir.

İnteraktif Analiz Araçları: Flask ile, kullanıcıların verileri filtreleme, sıralama veya özelleştirme gibi interaktif analiz işlemleri yapmalarını sağlayabilirsınız. Bu, kullanıcıların veriler üzerinde daha fazla kontrol sahibi olmalarına yardımcı olur.

Dashboard Oluşturma: Flask, veri analizi sonuçlarını bir kontrol paneli (dashboard) şeklinde sunmanıza olanak tanır. Kullanıcılar bu panelleri kullanarak farklı verilere erişebilir ve analiz sonuçlarını kolayca takip edebilir.

Veri Bilimi ve Flask Nasıl Birleştirilir?

- Veri Girişi ve Formlar
- Veritabanı Integrasyonu
- API Geliştirme
- Deployment ve Erişilebilirlik



Veri Girişi ve Formlar: Flask ile, kullanıcıların veri girişi yapabileceği ve bu verileri analiz süreçlerine dahil edebileceği web formları oluşturabilirsiniz.

Veritabanı Integrasyonu: Veri bilimi projeleri genellikle büyük veri setleriyle çalışır. Flask, SQLAlchemy veya MongoDB gibi veritabanı sistemleriyle entegre edilerek büyük veri setlerini işlemek ve verileri depolamak için kullanılabilir.

API Geliştirme: Flask, RESTful API'ler oluşturmaya olanak tanır. Bu, veri analizi sonuçlarını başka uygulamalar veya servislerle paylaşmanızı sağlar.

Deployment ve Erişilebilirlik: Flask ile geliştirilen web uygulamaları, projelerinizi daha fazla kişiye ulaştırmanızı sağlar. Çevrimiçi olarak dağıtılan uygulamalar, kullanıcıların herhangi bir cihazdan ve herhangi bir yerden erişebileceğiniz şekilde sunulabilir.

Veri bilimi ve Flask'in birleştirilmesi, veri bilimcilerine ve analistlere projelerini daha etkili bir şekilde paylaşma ve sonuçlarını daha geniş bir kitleye aktarma fırsatı sunar. Bu da iş dünyasında ve akademik dünyada veri analizi ve makine öğrenme projelerinin daha fazla kullanılmasını ve değer bulmasını sağlar.

Bir Flask Uygulaması Hangi Adımlardan Oluşur?

- Çevre Oluşturma ve Kurulum
- Proje Dizinini Oluşturma
- Uygulama Oluşturma
- Ana Uygulama ve Rotaların Tanımlanması
- HTML Şablonları ve Görseller
- Deployment



Flask uygulaması geliştirme süreci genellikle aşağıdaki adımları içerir:

Çevre Oluşturma ve Kurulum: Flask uygulamasına başlamadan önce, Python geliştirme ortamınızı (IDE) kurmalısınız. Ayrıca Flask kütüphanesini kurmalısınız. Flask'ı pip veya conda gibi Python paket yöneticileri kullanarak yükleyebilirsiniz.

Proje Dizinini Oluşturma: Flask uygulamanız için bir proje dizini oluşturun. Bu dizin, uygulamanızın dosyalarını içerecektir.

Uygulama Oluşturma: Flask uygulamasını oluşturun. Genellikle ana uygulama dosyanız (örneğin, app.py) bu adımla oluşturulur.

Ana Uygulama ve Rotaların Tanımlanması: Ana uygulama dosyanızda Flask uygulamanızı başlatın ve URL rotalarını (routes) tanımlayın. Her URL rotası, kullanıcının belirli bir URL'ye eriştiğinde hangi işlevin çağrılacağını belirtir.

HTML Şablonları ve Görseller: HTML şablonlarını ve statik görselleri projenize ekleyin. Şablonlar, web sayfalarınızın nasıl görüneceğini tanımlar.

Deployment: Flask uygulamanızı çevrimiçi sunucuya (örneğin, Heroku, AWS, veya diğer bir platform) dağıtn. Uygulamanızı genel kullanıcırlara açık hale getirin.

Deployment Nedir ve Neden Önemlidir?

- Erişilebilirlik
- Gerçek Dünya Testi
- Performans ve Ölçeklenebilirlik Değerlendirmesi
- Güvenlik
- Kullanıcı Geri Bildirimleri
- Yaygın Kullanım
- Yayın ve İzlenme



Deployment, yazılım veya uygulamanın geliştirme aşamasından üretim veya kullanılabilir bir duruma getirilmesi işlemidir. Deployment, geliştiricilerin veya yazılım ekiplerinin çalışmalarını gerçek dünya kullanıcıları tarafından erişilebilir hale getirme sürecini ifade eder. Deployment işlemi, yazılımın sorunsuz çalışmasını sağlamak ve kullanıcılarla sunulabilecek durumda olmasını amaçlar.

Deployment'in önemli nedenleri şunlar olabilir:

Erişilebilirlik: Deployment, yazılımı veya uygulamayı kullanıcıların erişebileceği bir duruma getirir. Bu, yazılımın veya uygulamanın amacına uygun şekilde kullanılabilmesi için önemlidir.

Gerçek Dünya Testi: Üretim ortamında çalışan yazılım, gerçek dünya koşullarında test edilebilir. Bu, geliştirme sırasında belirlenemeyen hataların tespit edilmesine yardımcı olabilir.

Performans ve Ölçeklenebilirlik Değerlendirmesi: Deployment, yazılımın performansını ve ölçeklenebilirliğini değerlendirmenize olanak tanır. Bu, kullanıcı sayısı arttıkça veya yüksek trafik dönemlerinde nasıl davranışını anlamınızı sağlar.

Güvenlik: Deployment, güvenlik açıkları veya hataları için bir test fırsatı sunar. Güvenlik açıkları, yazılımın üretim ortamında tespit edilip giderilebilir.

Kullanıcı Geri Bildirimleri: Üretim sürümü kullanıcılar tarafından kullanıldığında, geri bildirim almak ve kullanıcıların ihtiyaçlarına yanıt vermek daha kolaydır. Bu, yazılımı sürekli olarak geliştirmenize yardımcı olur.

Yaygın Kullanım: Deployment, yazılımın veya uygulamanın daha geniş bir kullanıcı kitlesi

tarafından kullanılmasını sağlar. Bu, projenizin veya ürünün başarısı için kritik öneme sahip olabilir.

Yedekleme ve Kurtarma: Üretim ortamında çalışan yazılımın yedekleri alınabilir ve felaket kurtarma senaryolarına hazırlıklı olunabilir.

Yayın ve İzlenme: Üretim sürümü kullanıcılara sunulurken, yazılımın veya uygulamanın yaylanması ve izlenmesi gerekebilir. Bu, kullanıcıların yazılımı veya uygulamayı güncellemleri için bilgilendirmenize olanak tanır.

Deployment işlemi, yazılım geliştirme sürecinin önemli bir parçasıdır ve kullanıcıların yazılımınızı veya uygulamanızı kullanabilmesini sağlar. Doğru bir deployment süreci, yazılımınızın güvenilirlik, performans ve kullanılabilirlik açısından başarılı olmasını sağlar. Bu nedenle, geliştiriciler ve yazılım ekipleri genellikle deployment sürecine büyük önem verirler.

Hangi Projeler Flask ile Yapılabilir?

- Blog ve İçerik Yönetim Sistemi (CMS)
- Kişisel Portfolyo ve Web Sitesi
- Veri Görselleştirme ve Analiz Aracı
- Oyun Pano ve Skor Takipçisi
- RESTful API
- Sosyal Medya Analizi Aracı
- E-Ticaret Uygulamaları



Flask, çeşitli projeleri geliştirmek için kullanılabilir ve genellikle hafif ve esnek yapısı nedeniyle çeşitli kullanım senaryolarına uygun bir web çerçevesi olarak kabul edilir. İşte Flask ile yapılabilecek bazı projelerin örnekleri:

Blog ve İçerik Yönetim Sistemi (CMS): Flask, kendi blogunuzu veya özelleştirilebilir bir içerik yönetim sistemi oluşturmak için kullanılabilir. Kullanıcıların makaleler eklemelerine, düzenlemelerine ve paylaşmalarına izin veren bir blog oluşturabilirsiniz.

Kişisel Portfolyo ve Web Sitesi: Flask, kişisel portfolyo veya web sitesi oluşturmak için kullanılabilir. Özelleştirilmiş bir şekilde tasarılanmış sayfaları ve projeleri sergileyebilirsiniz.

Veri Görselleştirme ve Analiz Aracı: Flask ile veri bilimi projelerinizin sonuçlarını görsel olarak sunan interaktif bir veri analiz aracı oluşturabilirsiniz. Grafikler, tablolar ve görsel analiz araçları ekleyebilirsiniz.

Oyun Pano ve Skor Takipçisi: Flask kullanarak bir oyun pano veya skor takipçisi oluşturabilirsiniz. Kullanıcılar skorlarını kaydedebilir ve rekabet edebilirler.

RESTful API: Flask, RESTful API'ler oluşturmak için kullanışlıdır. Bu API'ler, farklı uygulamalar veya cihazlar arasında veri alışverişi yapmanızı olanak tanır.

Sosyal Medya Analizi Aracı: Flask ile sosyal medya verilerini analiz eden bir araç oluşturabilirsiniz. Kullanıcılar, hashtag analizi, etkileşim takibi veya kullanıcı davranışları analizi gibi işlevleri kullanabilirler.

E-Ticaret Uygulamaları: Flask, özelleştirilebilir e-ticaret platformları oluşturmak için kullanılabilir. Ürün listeleme, sepet yönetimi ve ödeme işlemleri gibi temel e-ticaret

işlevsellliğini ekleyebilirsiniz.

Öğrenme Yönetim Sistemi (LMS): Flask ile öğrenme yönetim sistemi oluşturabilirsiniz.

Kursları listeleme, öğrenci kaydı ve ders içeriği sunumu gibi özellikleri içerebilir.

Görev Yöneticisi ve Zaman Yönetimi Araçları: Flask, görevlerinizi ve projelerinizi yönetmek için kullanabileceğiniz özelleştirilebilir bir görev yönetici veya zaman yönetimi aracı oluşturmanıza olanak tanır.

Hava Durumu ve Hesaplama Araçları: Flask ile hava durumu tahmini sağlayan bir uygulama veya hesaplama aracı geliştirebilirsiniz. Kullanıcılar, belirli bölgelerin hava durumu bilgilerini veya matematiksel hesaplamaları yapabilirler.

Flask, hafif yapısı ve esnekliği sayesinde pek çok farklı projenin temeli olabilir. Hangi projeyi oluşturacağınız, ilgi alanlarınıza, becerilerinize ve ihtiyaçlarınıza bağlı olarak değişecektir. Flask kullanarak özgün ve yaratıcı projeler geliştirmek için hayal gücünüzü kullanabilirsiniz.

Başarılı Bir Flask Uygulaması Örneği

Anket

Adınız:

Memnuniyet Seviyeniz: Çok Memnun

Anket Sonuçları

Adı Memnuniyet Seviyesi

Ahmet Çok Memnun

Ahmet Çok Memnun



Flask Eğitiminden Ne Bekliyorsunuz?

