



COMPREHENSIVE VUE.JS EĞİTİMİ 5 GÜN



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eğitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	5
Giriş ve Kurulum	5
Vue.js'deki temel kavramlar	5
Bileşenler ve Örnekler	5
Şablonlar	6
Bileşenlerle Çalışma.....	6
Olay işleme	6
Özel Etkinlikler	6
Kontrol Yapıları	7
CSS Bağlamaları	7
Formlarla Çalışma	7
Slots	7
Geçişler	8
Yönlendirmeye Giriş.....	8
Rota eşleştirme	8
Gelişmiş Yönlendirme	9
Vuex ile durum yönetimi	9
Temel kavramlar	9
Vuex ile çalışmak	9

Eđitim Hakkında

Comprehensive Vue.js eđitimi, web geliřtiricilerinin çok yönlü ve popüler bir JavaScript framework'ü olan Vue.js'yi derinlemesine anlamalarını sađlayan bir kurs programıdır. Günümüzde, web uygulamalarının performansı ve kullanıcı deneyimi kritik öneme sahiptir. Bu eđitim ile katılımcılar, Vue.js'nin karmařık yapılarını ve özelliklerini kullanarak web uygulamalarını daha etkili ve kullanıcı odaklı hale getirmeyi öğrenecekler.

Eđitim, özellikle Vue.js'nin temel konseptlerine ařına olan orta veya ileri düzey web geliřtiricilerine yöneliktir. Eđitimin amacı, Vue.js'in daha gelişmiş özelliklerini ve uygulama oluřturmada bu özelliklerin nasıl etkili bir şekilde kullanılabileceđini öğretmektir.

Eđitim boyunca, katılımcılar Vue.js component yapılarını ve bu componentler arasında nasıl veri akıřı sađlanacađını öğrenecekler. Özellikle, props, events ve slots gibi Vue.js özelliklerinin nasıl kullanılacađı üzerinde duruyoruz. Ayrıca, componentlerin yeniden kullanılabilirliđi, performans optimizasyonu ve ölçeklenebilirlik konuları da iřliyoruz.

Stil ve tasarım ađısından, eđitimde katılımcılara, componentlerin görünümünü ve davranıřlarını dinamik bir şekilde nasıl deđiřtirebilecekleri anlatılacak. CSS ve Vue.js'in kendi stil yönetim özellikleri ile nasıl daha çekici kullanıcı arayüzleri oluřturulabileceđi detaylı olarak ele alınacak.

Veri yönetimi, bir diđer önemli konudur. Katılımcılar, Vue.js ile birlikte çalıştırılabilen HTTP kütüphaneleri ile RESTful API'ler ve GraphQL gibi farklı veri kaynaklarından nasıl veri çekileceđini ve bu verilerin nasıl iřleneceđini öğrenecekler.

Vue.js ekosistemini derinlemesine inceliyoruz. Örneđin, Vuex ile durum yönetimi ve Vue Router ile uygulama içi yönlendirme gibi konular ele alınacak. Eklentiler ve Vue CLI gibi Vue.js ile gelen araçların nasıl kullanılacađı da bu eđitimde yer alacak.

İleri düzey JavaScript konseptleri ve modern JavaScript framework ve kütüphaneleriyle Vue.js entegrasyonu da kursun kapsamında olacak. Örneđin, TypeScript veya Nuxt.js gibi teknolojilerle nasıl daha sađlam uygulamaların geliştirilebileceđi üzerine vurgu yapıyoruz.

Kısacası, Comprehensive Vue.js eđitimi, web geliřtiricilere, Vue.js'in gücünden tam anlamıyla yararlanarak etkileyici, performanslı ve kullanıcı dostu web uygulamaları geliřtirmeleri için gereken beceri ve bilgiyi sađlar. Bu eđitim, katılımcıların kariyerlerini bir üst seviyeye taşıyarak, onları endüstride talep gören profesyoneller haline getirme potansiyeline sahiptir.

Neler Öğreneceksiniz?

Vue.js Comprehensive Eđitimi süresi boyunca, katılımcılar řunları öğrenecekler:

- js framework'ünün temel kavramları ve fonksiyonları
- Componentler arasında veri ve etkileřimleri yapılandırma



- Component tasarımı ve stilini deęiřtirme
- Farklı veri kaynaklarından veri çekme ve yapılandırma
- js ile birlikte kullanılan dięer teknolojilerle etkileřim kurma
- Uygulamanın performansını ve kullanıcı dostuluęunu optimize etme teknikleri
- js framework'ünün tüm özelliklerini ve fonksiyonlarını uygulama imkanı.
- Vuex ve Router kütüphanelerinin nasıl kullanılacaęı
- js ile geliştirilen uygulamaların test edilmesi ve debug edilmesi
- js ile geliştirilen uygulamaların scalability ve maintainability sıkıntılarının nasıl çözüleceęi
- js ile geliştirilen uygulamaların yapılandırılması ve optimize edilmesi
- js ile geliştirilen uygulamaların farklı cihazlarda nasıl düzgün çalışacaęı
- js ile geliştirilen uygulamaların farklı API'lar ve veri kaynakları ile nasıl etkileřim kurulacaęı.

Ön Kořullar

Vue.js Comprehensive Eęitimi'ne katılmak için řu ön kořulların yerine getirilmesi gerekir:

- HTML, CSS ve JavaScript'in temel kavramlarının bilinmesi
- JavaScript dilleriyle ilgili temel programlama becerilerinin olması
- JavaScript ile geliştirilmiş uygulamaların nasıl çalıştıęının ve yapılandırılmasının bilinmesi
- CSS dilleri ve frameworkleri hakkında temel bilgi sahibi olmak
- Terminal ve Git kullanımının bilinmesi (isteęe baęlı)

Ek olarak, ařaęıdaki beceriler de Vue.js Comprehensive Eęitimi'ne katılmak için faydalı olacaktır:

- JavaScript dillerinde en az bir framework veya library kullanmış olmak
- HTTP protokolü ve RESTful API'lar hakkında bilgi sahibi olmak
- Veri yapıları ve algoritmalar hakkında temel bilgi sahibi olmak.

Vue.js Comprehensive Eęitimi, ileri seviye bir programlama eęitimidir ve katılımcıların öncelikle programlama dilleri ve frameworkleri hakkında iyi bir anlayıřa sahip olması beklenir. Eęitim sırasında, katılımcılar ileri seviye programlama konseptleri ve tekniklerini öęrenme fırsatı bulacaklar.

Kimler Katılnalı

Vue.js Comprehensive Eęitimi, ařaęıdaki kiřiler için uygundur:

- Front-end veya full-stack web geliřtiricileri
- JavaScript dilleri ile geliştirilmiş uygulamaların yapılandırılması, optimize edilmesi ve test edilmesi konularında ileri seviye bilgi sahibi olanlar
- js veya dięer JavaScript framework veya library kullanmış olanlar
- js ile geliştirilen uygulamaların scalability ve maintainability sıkıntılarının çözümlmesini öęrenmek isteyenler



- Front-end web geliştirme veya full-stack web geliştirme konularında kendilerini geliştirmek isteyenler.

Vue.js Comprehensive Eğitimi, başlangıç seviyesinde Vue.js kullanmak isteyenler için uygun değildir ve eğitim süresi boyunca katılımcılar ileri seviye programlama konseptleri ve teknikleri hakkında bilgi

Outline

Giriş ve Kurulum

- Tanıtımlar
- Kurulumu doğrulayın
- js ve npm
- IDE
- Git
- Sınıf deposu
- js nedir?
- Vue benim için ne yapıyor?

Vue.js'deki temel kavramlar

- Vue'nun bir sayfaya dahil edilmesi
- Reaktif oluşturma
- js örneği
- Örnekler ve bileşenler
- Vue CLI
- Vue devtools

Bileşenler ve Örnekler

- Vue örneği
- Veri özelliği ile örnek verileri
- Method özelliği ile eylemler
- Olaylara yöntemler eklemek
- Yaşam döngüsü yöntemleri
- Vue bileşenleri
- Vue örneğiyle farklılıklar
- Veri özelliğindeki değişiklikler
- Yöntemlerde değişiklikler
- Bileşenleri ne zaman kullanmalıyım, örneği ne zaman kullanmalıyım?
- Tek dosya bileşeni
- js'de test etme



- Örnekleri test etme

Şablonlar

- Bir şablonda bir değişkeni işleme
- Bağlanma özellikleri
- Bağlayıcı olaylar
- Test şablonları
- Hesaplanan özelliklerle optimizasyon
- Hesaplanan mülk ve veri ne zaman kullanılır?
- Hesaplanan veriler ve yöntemler arasındaki etkileşimler
- Değişiklikleri izlemek
- Test saatleri

Bileşenlerle Çalışma

- Bileşenleri kaydetme
- Küresel ve yerel kayıt
- Modül sistemleri
- Ortak bileşenleri kaydetme
- Özellikleri
- Özellikleri bir bileşene aktarma
- Emlak türleri
- Özellikleri doğrulama
- Özellikleri test etme
- Öznitelik mirası

Olay işleme

- Satır içi ve yöntem işleyiciler
- Olay işlemeyi test etme
- Etkinlik değiştiriciler
- Klavye olaylarını alma

Özel Etkinlikler

- Özel etkinlik kavramları
- Özel olayları adlandırmak
- Özel etkinlikler üzerinden yerel etkinliklere bağlanma
- .Sync kullanma



Kontrol Yapıları

- V-if ile koşul ifadeleri
- v-başka
- v-başka-if
- V-show ile gizleme ve gösterme
- v-show vs v-if
- Koşulları test etme
- V-for ile döngüler
- Temel v-for
- Durum yönetimi ve değişiklikleri izleme
- Sıralama ve filtreleme
- Test döngüleri

CSS Bağlamaları

- CSS sınıflarını bağlama
- Nesne sözdizimi
- Dizi sözdizimi
- CSS'yi testlerle kontrol etme
- Satır içi CSS stillerini bağlama
- Nesne sözdizimi
- Dizi sözdizimi

Formlarla Çalışma

- V modelini kullanma
- Belirli form kontrolleri
- Metin alanları
- Onay kutuları ve radyo düğmeleri
- Seçimler / Açılır Menüler
- Çok satırlı metin alanları
- Form kontrollerini test etme
- V-modeli için değiştiriciler

Slots

- Slot nedir ve ne işe yarar?
- V-slot özelliği



- Slots arasında veri paylaşımı
- Kapsamlı yuvalar
- Yuvalarla içerik yönetim
- Adlandırılmış yuvalar
- Dinamik isimler
- Slot davranışını test etme

Geçişler

- js'deki değişiklikleri canlandırma
- Geçişleri listeleyin
- Giriş ve çıkış
- Liste öğeleri taşındığında (örneğin, sıralamada)
- Geçişlerin yeniden kullanılması
- Geçişlerin dinamik hale getirilmesi
- Durum geçişleri
- Ne dinlemeli
- İzleyicileri izlemek
- Dinamik durum geçişleri
- Geçişlerin bileşenlere dönüştürülmesi

Yönlendirmeye Giriş

- Yönlendirme nedir ve benim için ne yapar?
- Bir projeye yönlendirme ekleme
- CLI aracılığıyla
- Diğer yöntemler
- HTML güncellemeleri
- JS güncellemeleri
- Eşleşen rotalar
- Programlı gezinme
- Rota geçişleri

Rota eşleştirme

- Dinamik rotaları eşleştirme
- Parametre değişikliklerinde güncelleme
- Tümünü yakalama ve 404'ler
- İç içe rotalar
- Adlandırılmış rotalar



- Yönlendirmeler ve takma adlar
- Geçiş özellikleri

Gelişmiş Yönlendirme

- Navigasyon korumaları
- Veri alma ve yönlendirme
- Geç yükleme yolları
- Kaydırma davranışı

Vuex ile durum yönetimi

- Vuex nedir ve benim için ne yapar?
- Vuex Kurulumu
- CLI aracılığıyla
- Diğer yöntemler
- Vuex ve geliştirme araçları

Temel kavramlar

- Akı tarzı durum yönetimi
- Durum
- Getters
- Mutasyonlar
- Hareketler
- Modüller

Vuex ile çalışmak

- Uygulamanızı yapılandırma
- Form verilerini ve etkileşimleri işleme
- Vuex geçişleri
- Vuex'i Test Etmek