

Vue.js Eğitimi

Süre: 5 Gün

Eğitim Açıklaması

Vue.js Eğitimi, JavaScript ile kullanıcı arayüzleri geliştirmek isteyen katılımcılar için hazırlanmış 5 günlük bir programdır. Sanal Sınıf / Online formatındaki eğitimde DOM etkileşimleri, class ve style işlemleri, koşullar, listeler ve Vue instance yapısı ele alınır.

Program; Vue CLI kurulumu, component yapısı, componentler arası iletişim ve gelişmiş component kullanımını kapsar. Directive, filter ve mixin konuları, arayüz geliştirme yaklaşımının diğer bölümlerini oluşturur. Katılımcılardan temel JavaScript, HTML ve CSS bilgisi beklenir.

Axios ile HTTP istekleri, Vue Router ile routing, CRUD işlemleri ve Vuex ile state yönetimi müfredatın uygulama bölümlerinde işlenir. Auth işlemleri, NPM üzerinden component yayımlama, Translate API ile translator örneği ve deployment konuları da kapsamda yer alır.

Eğitim İçeriği

Vue.js Nedir?

- Vue.js Neden Kullanılmalıdır?
- Vue, React, Angular kıyaslaması
- Basit Vue uygulaması oluşturmak

Vue.js DOM Etkileşimleri

- Vue Template Yapısı
- Template Syntax ve Vue instance nasıl çalışır?
- Instance içerisindeki data property bilgilerine ulaşmak
- Attribute Bind
- Directive Nedir? Nasıl kullanılır?
- HTML kodlarını Template üzerinde göstermek
- Event Dinlemek
- \$event objesinden veri almak
- Event modifier
- Template içerisinde JavaScript kodu yazmak
- Two Way DataBinding

- Computed Property Kullanımı
- Watcher Kullanımı

Class ve Style İşlemleri

- Dinamik olarak CSS Class ekleme işlemi
- Dinamik olarak Stil kodları ekleme işlemi

Vue.js Koşul Kullanımı ve Listeler

- v-if, v-else-if, v-else kullanımı
- v-show kullanımı
- v-if ile v-show farkı
- v-for ile listeleri render etmek
- v-for ile object döndürmek
- v-for ile index numarasını almak
- Listelerde v-bind:key özelliği

Vue.js Instance Nedir?

- Vue Instance ile ilgili temel bilgiler
- Birden fazla Vue instance ile çalışmak
- Vue Instance'a dışarıdan ulaşmak
- Vue Data ve Method verilerini nasıl yönetir?
- \$el ve \$data objelerine daha yakından bakalım
- Template içinde \$refs kullanımı
- Vue LifeCycle

Vue CLI Kurulumu

Component Yapısına Giriş

- Component Neden Kullanılır?
- Kaç farklı component tanımlanabilir?
- Component içerisinde data metodu ile veri depolama
- Component' leri Lokal ve Global olarak kayıt ettirmek
- vue içerisindeki Root Component

Component ile Çalışmak

- Component yapısını daha iyi hale getirmek
- Component' leri isimlendirmek
- Component Style Scope İşlemleri

Componentler Arası İletişim

- İletişim Problemleri
- Parent => Child Arası iletişim
- Child => Parent Arası iletişim
- Child => Child Arası iletişim
- Event Bus Kullanımı
- Ödev - Pratik

Gelişmiş Component Kullanımı

- İçerik Göndermek
- Slot ile içerik göndermek
- Named Slots ile içerikleri ayırtmak
- Default Slot ve Slot Defaults
- Componentler arasında dinamik olarak Switch işlemi □ Dinamik Component'lerin davranışını anlayalım!
- Dinamik Component'leri canlı tutmak (keep alive)
- Dinamik Component'lerde LifeCycle Hooks
- Ödev - Pratik

Directive Kullanım

- Directive nasıl çalışır? Hook fonksiyonları
- Basit directive oluşturmak
- Directive'e veri göndermek
- Directive'e argüman göndermek
- Modifier ile directive değiştirmek
- Directive'lerin local olarak tanımlanması

Filter ve Mixins Kullanımı

- Local Filter Oluşturmak
- Global Filter ve birden fazla Filter'ı zincirleme olarak kullanmak
- Filter'a alternatif (Computed Property)
- Mixins Nedir? Mixins yapısını anlayalım
- Mixin Oluşturma ve Kullanma
- Mixin'ler nasıl birleştirilir?
- Global Mixin oluşturma
- Mixin ve Scope Kavramı

Vue.js İle HTTP İstekleri Atmak (Axios)

- Axios sınıfının yüklenmesi
- GET isteği
- POST isteği
- PATCH isteği
- DELETE isteği
- GET ile beraber Parametre gönderimi (istisnaî durum)
- Custom Axios Instance oluşturmak

Vue Router İle Routing Yapısı

- Router ve Route Nedir?
- Route tanımlamak
- Routing Metodları
- Lazy Routing
- Router Linkler ile Componentler arası gezinme
- Neredeyim ben? Aktif Linkleri Stillendirme
- JavaScript üzerinden routing işlemleri
- Route ile parametre gönderme ve okuma
- Redirect İşlemi

Örnek Uygulama (CRUD İşlemleri)

Vuex İle State Management

- Neden daha farklı bir State Yönetimine ihtiyacımız var?
- Central Store ve State Kavramları
- Vuex Yapılandırması ve Store Oluşturmak
- State Verilerine Ulaşım yöntemleri
- Getters Nedir? Nasıl Çalışır?
- mapGetters kullanımı
- Getters için spread operator kullanımı
- Mutations Nedir? Nasıl Çalışır?
- Mutations Kullanımı
- mapMutations Kullanımı
- Actions Nedir? Nasıl Çalışır?
- mapActions kullanımı
- Mutations ve Actions içerisinde parametre kullanımı
- Modül Sistemi

Örnek Uygulamalar

- Auth İşlemleri
- Component Hazırlayıp NPM üzerinden Yayınlamak
- Translate API ile Translator Yapımı

Deployment İşlemleri

- Vue Projesini Canlı Sisteme Almak

Ön Koşullar

Temel seviye JavaScript, HTML ve CSS bilgisi gerekmektedir.