
İleri Seviye HTML5

İleri Seviye HTML5 Geliştirme eğitimi, katılımcılara HTML5 becerilerini bir sonraki seviyeye taşımak için ihtiyaç duydukları her şeyi öğretir.

Hedefler:

- Bir cihazın pilinin şarj durumunu öğrenmeyi
- Dokunma duygusu ile son kullanıcıların dikkatini çekmeyi
- Web tarayıcısının dışında görünen son kullanıcı bildirimlerini göndermeyi
- Cihazın fiziksel yönüne dokunmayı
- Web sayfanızın bölümlerini son kullanıcı tarafından düzenlenebilir hale getirmeyi
- Web sayfasının bölümlerini, kullanıcıların tüm ekranını kaplayacak şekilde görüntülemeyi
- Ses tanıma ve konuşma sentezini uygulamayı
- Yeni medya unsurlarından (ses ve video) yararlanmayı
- Yeni <form> öğeleri ve nitelikleriyle çalışmayı
- WebStorage API ile çalışmayı
- Farklı bir etki alanından web sayfaları dahil olmak üzere web sayfalarından mesaj göndermeyi ve almayı
- HTML öğelerini verilerle ilişkilendirmeyi
- Web sayfasındaki herhangi bir nesneyi sürüklenebilir yapmayı
- XMLHttpRequest (XHR) düzey 2'nin yeni özelliklerini kullanmayı
- WebSocket özelliği ile istemciden sunucuya tam çift yönlü bağlantılar açmayı
- Sunucu ve istemci arasında tek yönlü kanallar oluşturmayı
- <canvas> öğesi ile bir 2B çizim yüzeyi oluşturmayı ve JavaScript ile tuval üzerinde "çizim" yapmayı
- Uygulamalarınızda coğrafi konum bilgilerini kullanmayı
- Web uygulamalarınızın performansını artırmak için web çalışanlarını kullanmayı

Topics:



- BatteryStatus API
 - BatteryStatus API neden kullanılıyor?
 - BatteryStatus API nasıl çalışır?
 - BatteryStatus API
 - BatteryStatus API yöntemleri
 - BatteryManager özellikleri
 - BatteryManager olayları
 - Tarayıcı desteği
- Titreşim API'si
 - Titreşim API'sini neden kullanmalısınız?
 - Titreşim API'si nasıl çalışır?
 - Titreşim yöntemleri
- Bildirimler API'si
 - Bildirimler API'sini neden kullanmalısınız?
 - Notifications API nasıl çalışır?
 - Bildirimler API'si
 - Bildirim nesnesi nitelikleri
 - Bildirim olayları
 - Bildirim API yöntemleri
- DeviceOrientation Event API
 - DeviceOrientation API neden kullanılır?

- DeviceOrientation Event API
- DeviceOrientation olayları
- DeviceOrientation Event API nasıl çalışır?
- DeviceOrientation
- DeviceMotion
- DeviceOrientation Event API
- Tam ekran API
 - Neden Tam Ekran API'sini kullanmalı?
 - Tam Ekran API
 - Tam ekran API yöntemleri
 - Tam ekran API özellikleri
 - Tam ekran API etkinlikleri
 - Tam ekran görünümüyle ilgili notlar
- Konuşma API'leri
 - Konuşma Tanıma ve Konuşma Sentezi
 - Konuşma sentezi
 - SpeechUtterance nesnesi
 - SpeechUtterance nesne özellikleri
 - SpeechUtterance nesne olayları
 - Konuşma tanıma
- HTML5 Medya öğeleri ve API

- Ses ve video
- Medya unsurları
- Yeni unsurlar ne için kullanılıyor?
- Yeni medya öğeleri için tarayıcı desteği
- <audio> öge öznitelikleri
- <video> öge öznitelikleri
- Tarayıcı desteği <video>
- Tarayıcı desteği Ogg / Theora video formatı *
- Tarayıcı desteği WebM / VP8 video formatı *
- Tarayıcı desteği MPEG-4 / H.264video formatı *
- Tarayıcı desteği ses codec'i
- Bu yeni unsurları ne zaman kullanırım?
- Video ve Ses API'leri
- Video / ses yöntemleri
- Video / ses nesnesi özellikleri
- Video / ses nesnesi olayları
- Formlar / Kısıtlama Doğrulama API'si
 - Yeni form giriş türleri
 - Yeni giriş türleri nelerdir?
 - Giriş öğeleri için yeni nitelikler
 - Yeni form öğeleri

- Yeni unsurları ne zaman kullanırım?
 - Yer tutucu özelliđi
 - Otomatik tamamlama özelliđi
 - Otomatik odaklama özelliđi
 - Liste özelliđi ve veri listesi öđesi
 - Yazım denetimi özelliđi
 - HTML5 form dođrulama
 - Varsayılan dođrulamanın gerçekleşmesini önleme
 - Kısıtlama Dođrulama API'si
 - Form özellikleri
 - Form kontrol yöntemleri
 - Form geçerliliđi ve geçerliliđi kontrol eder
- WebStorage API
 - WebStorage API için tarayıcı desteđi
 - WebStorage API
 - Depolama Nesnesi
 - Ne zaman
 - Çerezler, oturum depolaması ve yerel depolama
- Web Mesajlaşma API'si
 - Web mesajlaşmasının postMessage () yöntemi
 - Etki alanları arasında güven tesis etmek

- Köken
- Kaynaklar arası kaynak paylaşımı nedir?
- CORS desteği ve sunucu
- Bağlantı noktaları ve kanal mesajlaşma
- MessagePort yöntemleri
- Message olay nesnesi özellikleri
- Veri Kümesi API'si
 - veri- * sözdizimi kuralları
 - Veri kümesi API'si
- Sürükle ve Bırak API
 - Sürükle ve bırak nedir?
 - Sürükle ve bırak ne için kullanılır?
 - Sürükle ve bırak için tarayıcı desteği
 - Sürükleyip bırakmayı anlama
 - Sürüklenabilir nitelik
 - Veri "yükü" ile sürükleyip bırakın
 - Olayları sürükleyin
 - Olayları bırak
- XMLHttpRequest Seviye 2 API'si
 - Hedefler
 - XMLHttpRequest nesnesi

- XHR API
- XHR ne için kullanılır?
- XHR seviye 2'nin faydaları
- XHR nesnesi: readyState özelliği
- XHR nesnesi
- XHR nesnesi: Olaylar
- XHR Nesnesi: Yöntemler
- XHR nesnesi: Özellikler
- Web Yuvaları API'si
 - HTTP
 - WebSocket özellikleri
 - WebSocket olay işleyicileri
 - WebSocket yöntemleri
 - WebSocket API ile bir WebSocket sunucusu kullanma
- Sunucu tarafından gönderilen Etkinlikler API'si
 - Sunucu mesajları
 - EventSource Oluşturma
 - EventSource özellikleri
 - EventSource olayları
 - Sunucu tarafından gönderilen olayların avantajları
 - Sunucu tarafından gönderilen olayların dezavantajları

- Sunucu tarafından gönderilen olayların özeti
- Canvas API
 - Yeni unsurlar neler?
 - Canvas API'yi neden kullanmalısınız?
 - Tuval hızlı özeti
 - Yeni Canvas API için kullanımlar
 - Canvas Demoları
- Coğrafi Konum API'si
 - Geolocation API ne için kullanılır?
 - Coğrafi Konum için tarayıcı desteği
 - Coğrafi Konum API'si
 - Coğrafi konum nesne yöntemleri
 - Nesne özelliklerini konumlandırma
 - Konum arayüz özellikleri
- Web Çalışanları API'si
 - Web Çalışanları tarafından kullanılabilen özellikler
 - Neden Web Çalışanları kullanılıyor?
 - Web Çalışanları ne zaman kullanılmalıdır?
 - Web Çalışanları nasıl çalışır?
 - Web Çalışanı API'si
 - AbstractWorker arabirim özellikleri

- WorkerGlobalScope nesnesi
- WorkerGlobalScope nesne özellikleri
- WorkerGlobalScope nesne yöntemleri
- WorkerGlobalScope etkinlikleri
- Web Çalışanları için tarayıcı desteği
- Web Çalışanı Türleri
- Özel çalışan
- Paylaşılan Çalışan
- Paylaşılan Çalışanlar Yaratmak