Introduction to Spark Programming

Açıklama

Apache Spark'ın dağıtık bilgi işlem motoru hakkında temel bilgiler sunan bu eğitim geliştiriciler, veri analistleri, mimarlar, teknik yöneticiler ve Spark'ı uygulamalı olarak kullanma ihtiyacı duyan herkes için uygundur.

Bu eğitimde katılımcılar Spark mimarisi ve Spark'ın nasıl çalıştığıyla konusunda temel teknik bilgilere sahip olur. Spark'ın temel yapıtaşlarının (ör. RDD'ler ve dağıtık bilgi işlem motoru) yanı sıra daha basit ve daha yetenekli bir arayüz (ör. Spark SQL ve DataFrames) sağlayan daha üst düzey yapıları içerir. Akış verilerinin işlenmesi için Spark Streaming'in kullanımı gibi daha gelişmiş yetenekleri içerir ve Spark ML (makine öğrenimi) hakkında genel bilgiler sağlar. Eğitim son olarak olası performans sorunları ve optimizasyon için kullanılabilecek stratejilerin araştırılmasına da değinir.

Bu, birden fazla laboratuvarda işlenen çok uygulamalı bir eğitimdir. Katılımcılar Spark kabuğu (etkileşimli, geçici işleme için) ve Spark API'si kullanan programlar aracılığıyla Spark'la etkileşime geçeceklerdir.

Apache Spark dağıtık bilgi işlem motoru, büyük ölçekli veri kümelerinin işlenmesi ve analizi konusunda hızlı bir şekilde birincil araç haline geliyor. Hadoop gibi mevcut motorlara göre 10 ila 100 kat daha hızlı çalışma süresi hızları ve daha basit programlama modeli dahil olmak üzere birçok avantaja sahiptir. Bu eğitimi tamamladıktan sonra, Spark ile daha bilinçli ve üretken bir şekilde çalışmaya hazır olacaksınız.

Bu eğitimde neler öğreneceksiniz?

- Veri işlemede neden Spark'a ihtiyaç duyulduğu
- Spark mimarisi ve hesaplamaları küme düğümlerine nasıl dağıttığı
- Spark'ın yüklenmesi, kurulumu ve düzeni hakkında temel bilgiler
- Etkileşimli ve geçici işlemler için Spark kabuğunun kullanımı
- RDD'ler (Resilient Distributed Datasets), veri bölümlendirme, komut zinciri ve hesaplamalar konusunda bilgiler
- map(), filter(), reduce(), groupByKey(), join() ve benzeri RDD işlemleri hakkında

bilgi ve bunları kullanma
- Spark'ın veri önbelleği ve kullanımı
- Spark API'siyle bağımsız Spark programları yazma ve çalıştırma
- Yapılandırılmış verileri verimli bir şekilde işlemek için Spark SQL / DataFrames kullanımı
- Akış (gerçek zamanlı) verilerini işlemek için Spark Streaming'i kullanma
- Spark kullanılırken ortaya çıkan performans sonuçları ve optimizasyonları
- Spark ML ile ilgili bilgiler

Eğitim İçeriği

Apache Spark Nedir?

Apache Spark Mimarisi ve Kurulumları

Apache Spark RDD Yapısı

Apache Spark Proje Oluşturma

Veri Yükleme Aşamaları

Transformasyon Yapısı Map Methodu

Transformasyon Yapisi FilterMethodu

Flat Map Distinct Methodu

PairRDD ve GroupByKey Methodu

Lazy Evulation Kavrami

Action Methodu

SparkSQL Nedir?

SparkSQL Veri Okuma Aşamaları

Introduction to Spark Programming

SparkSQL StructType

SparkSQL Filter Yapisi

Spark SQL Group By Yapisi

Spark SQL API

Spark Tempview Globalview Kavramlari

Spark ve Hadoop(HDFS) Entegrasyonu

Spark ile Uçtan Uca Proje Geliştirme

Spark ile CallCenter Veri Analizi

Call Center Sonuclari MongoDB'ye Yazma

Spark Streaming Gerçek Zamanlı Veri Analizi Nedir?

Spark Streaming Çesitleri

Spark Streaming ile Anlik Mesaj Analizi

Spark Streaming Örneginin Mimarisi

Spark Streaming ile IoT Analizi

Streaming Complete ve Update Modları

Streaming Time Window (ZamanGruplari)

Streaming Time Window ile Mesaj Analizi

Streaming ile Kafka Entegrasyonu

Spark İle Makine Öğrenmesine Giriş – Spark MLLib?

Spark MLlib Kütüphanesi

Tahmin(Regresyon) Nedir?

Introduction to Spark Programming

Linear Regression(DogrusalRegresyon) Nedir?

Spark MLlib ile Linear Regression Uygulaması

Model Değerlendirme R2 YÖntemi

Naive Bayes Algoritması

Spark MLlib ile Naive Bayes Uygulaması

Naive Bayes ile örnek uygulama

Ön Koşullar

Programlama tecrübesine sahip olmak