

Yapay Zeka Destekli Kodlama Eğitimi

Süre: 2 Gün

Kurumsal Yapay Zeka Destekli Kodlama Eğitimi Hakkında

Yapay Zeka Destekli Kodlama Eğitimi, yazılım ekiplerinin büyük dil modellerini (LLM) geliştirme süreçlerinde bilinçli ve kontrollü biçimde kullanmasına yönelik 2 günlük kurumsal eğitimidir. Program; token, prompt ve embedding gibi temel kavramlardan başlayarak kod üretimi, refactor, test oluşturma, basit RAG uygulaması ve IDE içindeki çalışma akışlarını ele alır.

Katılımcılar, yapay zekadan üretilen kodu değerlendirmeyi, ekip standartlarına uygun istemler hazırlamayı ve çıktıları test ederek doğrulamayı uygulamalarla öğrenir. Güvenlik, kişisel verilerin korunması, versiyonlama, maliyet yönetimi ve gözlemlenebilirlik konuları da üretim ortamına uygun geliştirme yaklaşımı içinde işlenir.

Eğitimde Model Context Protocol (MCP) ile araç ve servis entegrasyonuna giriş yapılır. Kurum içi servislerin LLM tabanlı araçlarla bağlanmasını daha ayrıntılı çalışmak isteyen ekipler, Model Context Protocol (MCP) Eğitimi içeriğini inceleyebilir.

Eğitim Sonunda Kazanılacak Yetkinlikler

- LLM, token, context window, embedding ve vektör kavramlarını yazılım geliştirme senaryolarında açıklamak.
- Kod üretimi, refactor ve test oluşturma için açık, değerlendirilebilir promptlar hazırlamak.
- Indexing, similarity search ve retrieval adımlarını içeren basit bir RAG akışını uygulamak.
- Yapay zeka araçlarını IDE ve Git tabanlı ekip süreçlerine yerleştirmek.
- Üretilen kodu test, kod incelemesi ve takım kurallarıyla doğrulamak.
- PII maskeleyme, model seçimi, caching ve batching gibi güvenlik ve verimlilik kararlarını değerlendirmek.

Kimler Katılmalı?

- Backend, frontend ve mobil uygulama geliştiricileri.
- Ürün ve prototip geliştirme ekipleri.
- IDE içinde LLM destekli geliştirme süreçleri oluşturmak isteyen yazılım ekipleri.
- Yapay zeka kullanımında ortak kod kalitesi ve inceleme standartları belirlemek isteyen teknik ekipler.

Eğitim İçeriği

LLM Temelleri ve Uygulamalı Kodlama

- LLM, token, embedding ve vektör kavramları.
- Context window ve token bütçesinin yönetimi.
- Rol, çıktı biçimi ve örneklerle prompt tasarımı; çıktıların değerlendirilmesi.
- Basit RAG akışı: indexing, similarity search ve retrieval.
- Fonksiyon ve kod parçacığı üretimi; mevcut kodun refactor edilmesi.
- Unit test ve test verisi üretimi; sonuçların doğrulanması.
- Güvenlik ve içerik filtreleme; kişisel verilerin (PII) maskelenmesi.
- Caching, batching ve model seçimiyle maliyet ve hız optimizasyonu.

RAG uygulamalarındaki vektör arama ve depolama altyapısını daha derinlemesine öğrenmek isteyen ekipler için Vektör Veritabanları Eğitimi ilgili bir devam programıdır.

Uygulama Günü: Promptlar, IDE Entegrasyonları ve Ekip Pratikleri

- Prompt kalıpları, örnekler, sık yapılan hatalar ve değerlendirme yöntemleri.
- VS Code, Cursor ve Windsurf gibi IDE ortamlarında yapay zeka destekli çalışma akışları.
- Takım kuralları, stil rehberleri ve tekrar kullanılabilir prompt şablonları.
- Kod üretimi, refactor ve test oluşturma için uygulamalı iş akışları.
- Caching, batching ve model seçiminin maliyet ile performansa etkisi.
- Pull request (PR) yorumları, prompt kalite kontrolü ve ekip içi kod inceleme süreçleri.
- **Proje:** Git deposuyla bağlantılı, tasarımdan koda uzanan uygulama çalışması.

IDE içindeki Cursor kullanımını ayrı bir programda ayrıntılandırmak isteyen ekipler Cursor ile AI Native Yazılım Geliştirme Eğitimi içeriğini inceleyebilir.

Eğitim Katılım Ön Koşulları

Katılımcıların programlama konusunda giriş düzeyinde bilgi sahibi olması gereklidir.