

# Model Context Protocol (MCP) Eğitimi

Süre: 1 Gün

## Eğitim Açıklaması

### LLM Entegrasyonu ve MCP Server Geliştirme - 1 Günlük Teknik Program

**Model Context Protocol (MCP) Eğitimi**, büyük dil modeli (LLM) tabanlı uygulamaları API, dosya sistemi, veritabanı ve Git gibi harici araçlara ve veri kaynaklarına bağlamak isteyen teknik ekipler için hazırlanmıştır. Bir günlük online program; MCP mimarisini, uygulamalı MCP server geliştirmeyi, Claude Code ve Cursor entegrasyonlarını ve kurumsal güvenlik kontrollerini kapsar.

Eğitimin amacı, katılımcıların MCP istemcisi ile sunucusu arasındaki iletişimi anlaması, temel bir MCP server geliştirmesi ve tanımladığı araçları uyumlu yapay zeka uygulamalarında kullanabilmesidir. Program boyunca araç şemaları, çağrılarının işlenmesi, girdi ve çıktı doğrulaması ile yetki sınırlandırması uygulama senaryoları üzerinden ele alınır.

### Model Context Protocol (MCP) Nedir?

Model Context Protocol, yapay zeka uygulamalarının harici araçlar ve veri kaynaklarıyla iletişim kurmasına yardımcı olan açık bir protokoldür. Bir MCP server, belirli araçları ve kaynakları uyumlu istemcilere sunabilir. Böylece uygulamalar, tanımlanmış erişim sınırları içinde veri alabilir veya belirli işlemleri gerçekleştirebilir.

Eğitimde MCP'nin klasik REST API entegrasyonlarıyla ilişkisi de incelenir. REST API mevcut bir sistemin sunduğu işlemler için kullanılabilir; MCP ise bu işlemlerin yapay zeka uygulamalarına araç olarak nasıl sunulacağını düzenleyen bir iletişim yapısı sağlar.

### Eğitimin Çıktıları

- MCP mimarisini ve istemci-sunucu iletişimini teknik açıdan kavrama.
- Bir araç için tanım ve şema oluşturarak temel bir MCP server geliştirme.
- Girdi ve çıktı doğrulaması ile araç çağrılarını işleme.
- LLM tabanlı uygulamalar ile harici araçlar arasında entegrasyon kurma.
- Claude Code ve Cursor ortamlarında MCP araçlarını kullanma.
- Kurumsal MCP mimarisinde erişim yetkileri, kayıt tutma ve denetim ihtiyaçlarını değerlendirme.
- Güvenli ve denetlenebilir yapay zeka entegrasyonları için temel kontrolleri belirleme.

## MCP Eğitimi Kimler İçin Uygundur?

Program; LLM tabanlı uygulamalar geliştiren yazılım geliştiriciler, yapay zeka ekipleri, entegrasyon uzmanları ve çözüm mimarları için uygundur. Kurumsal araçları ve verileri yapay zeka uygulamalarına kontrollü biçimde açmayı planlayan teknik ekipler de eğitimden yararlanabilir.

## Uygulamalı MCP Server Geliştirme

Eğitimde araç tanımlama, şema oluşturma, handler fonksiyonu yazma ve yerel MCP server çalıştırma adımları incelenir. Claude Code ve Cursor bölümlerinde araçların bağlanması, çağırılması ve dönen sonuçların kullanılması ele alınır. Final uygulamasında katılımcılar basit bir MCP server geliştirir, bir araç üzerinden veri çeker ve elde edilen verilerle rapor oluşturur.

MCP bağlantılarını agent mimarisinin bir parçası olarak değerlendirmek isteyen ekipler, AI Agent Geliştirme ve Kullanma Eğitimi programını inceleyebilir. Diğer programlar için kurumsal yapay zeka eğitimleri kategorisine göz atabilirsiniz.

## Eğitim İçeriği

### 1. MCP Nedir? Teorik Temeller

- LLM tabanlı uygulamaların harici sistemlerle iletişim kurmak için neden entegrasyon bileşenlerine ihtiyaç duyduğu.
- Araç çağırma (tool calling) kavramı.
- MCP'nin ortaya çıkış motivasyonu ve standart bir bağlantı yapısına duyulan ihtiyaç.
- MCP ile klasik REST entegrasyonunun karşılaştırılması.
- Harici verinin modelin kullanımına sunulması (context injection).
- MCP istemcisi ve MCP sunucusu (client-server) mimarisi.

### 2. MCP Server Yazımı: Uygulamalı Geliştirme

- MCP aracı (tool) tanımlama ve şema (schema) oluşturma.
- Girdi ve çıktı doğrulaması (input ve output validation).
- Araç çağrılarını işleyen handler fonksiyonlarının yazılması.
- JSON-RPC iletişim mantığı.
- Yerel ortamda MCP server çalıştırma.

### 3. Claude Code ile MCP Kullanımı

- MCP araçlarını Claude Code ortamına bağlama.
- Araç çağırma ve dönen yanıtı işleme.
- Önce planlama, ardından araç çağırma yaklaşımı (plan-first ve tool-call).
- Bir araç üzerinden veri çekme ve analiz etme senaryosu.

## 4. Cursor ve MCP Entegrasyonu

- Agent mode üzerinden MCP araçlarını kullanma.
- Kod üretme sürecinde MCP çağrısı yapma.
- Dosya sistemi ve Git entegrasyonlarını değerlendirme.

## 5. Güvenlik ve Kurumsal Kontroller

- Araçların işlem ve erişim kapsamını belirleme (tool scoping).
- Erişim yetkilerini ihtiyaç duyulan işlemlerle sınırlandırma.
- Prompt injection risklerini değerlendirme.
- Araç çıktılarının doğrulanması (output validation).
- Denetim kayıtları ve log mekanizmaları.
- Kurum içi (on-prem) MCP server tasarımı.

## 6. Final Uygulama

- Basit bir MCP server geliştirme.
- Geliştirilen sunucuyu Claude Code'a bağlama.
- MCP aracı üzerinden veri çekme.
- Alınan verilerden rapor üretme.
- Entegrasyon üzerinde bir güvenlik senaryosu uygulama.