

LangChain ile Agent ve Tool Kullanımı

LangChain’de çeşitli **agent türleri** vardır ve her biri farklı kullanım senaryoları için optimize edilmiştir. İşte başlıca agent türleri ve açıklamaları:

1. ZERO_SHOT_REACT_DESCRIPTION (En yaygın)

- **Kullanımı kolay** ve çoğu temel kullanım için yeterlidir.
- Kullanıcıdan gelen girdiye göre hangi aracı kullanacağını **kendisi belirler**.
- Tool’lara yazılmış açıklamaları (“description”) kullanarak karar verir.

✦ **Kullanım örneği:**

```
agent_type=AgentType.ZERO_SHOT_REACT_DESCRIPTION
```

2. STRUCTURED_CHAT_ZERO_SHOT_REACT_DESCRIPTION

- Zero-shot ile aynı mantıkta çalışır ama **structured (yapılandırılmış)** mesaj formatı kullanır.
- Tool aramalarında daha tutarlı sonuçlar alınabilir.
- OpenAI GPT-4 gibi modellerle **daha iyi sonuçlar** verir.

✦ **Kullanım örneği:**

```
agent_type=AgentType.STRUCTURED_CHAT_ZERO_SHOT_REACT_DESCRIPTION
```

3. REACT_DOCSTORE

- Belge tabanlı aramalarda etkilidir.
- Bir doküman deposu (örneğin Wikipedia gibi) üzerinden arama yapıp bilgi getirir.

✦ **Örnek kullanım:**

```
agent_type=AgentType.REACT_DOCSTORE
```

4. SELF_ASK_WITH_SEARCH

- Önce soruyu parçalar, sonra arama yapar.
- **Kompleks ve çok adımlı sorgular** için daha uygundur.
- SerpAPI gibi arama motorları ile birlikte kullanılır.

5. OPENAI_FUNCTIONS

- OpenAI’nin functions yapısıyla entegre çalışan bir agent türüdür.
- Daha kontrollü ve **araç çağrısı odaklıdır**.
- ChatGPT function-calling mantığıyla çalışır.

🔗 Kullanımı biraz daha gelişmiş konularda tercih edilir.

🔍 6. Conversational ReAct Agent (ChatConversationAgent)

- **Kullanım:** Chat geçmişi dikkate alır, çok adımlı konuşmalarda kullanılır
- **Durum:** LangChain 0.1 sonrası daha az tercih edilir çünkü StructuredChat bu işlevi üstlenmiştir
- **Not:** UI uygulamaları için faydalı olabilir

Agent Türü 🌱	Kullanım Amacı 📌	Avantajları ✅	Sınırlamaları ⚠️	Tool Gereksinimi 🛠️	Örnek Soru 💬
ZERO_SHOT_REACT_DESCRIPTION	Tool açıklamalarına göre otomatik karar	Basit, hızlı ve esnek	Karmaşık sorgularda yetersiz kalabilir	Açıklaması olan herhangi bir tool	"2023 Nobel Barış Ödülü'nü kim aldı?"
SELF_ASK_WITH_SEARCH	Zincirleme düşünme, web arama	Multi-hop reasoning yeteneği	Tool adı Intermediate Answer olmalı	Sadece tek tool destekler	"Kız Kulesi nerede ve yakınındaki oteller?"
STRUCTURED_CHAT_ZERO_SHOT_REACT_DESCRIPTION	Form / UI destekli yapılar, net argümanlar	JSON destekli, kontrollü çağrılar	Tool şemaları gerekiyor	Tool'lar yapılandırılmış olmalı	"Ankara'da 3 gün yağmur yağacak mı?"
OPENAI_FUNCTIONS	Function-calling API ile dış servis bağlantısı	En stabil ve güvenli tool çalıştırma	Sadece OpenAI modelleriyle çalışır	Python fonksiyonu + açıklama	"İstanbul'da hava durumu nedir?"
REACT_DOCSTORE	Docstore (Wikipedia gibi) üzerinde bilgi sorgulama	Wikipedia tarzı içeriklerde hızlıdır	Deprecated olabilir, modül sorunlu	DocstoreExplorer kullanımı gerekir	"Türkiye'nin komşuları kimlerdir?"
CHAT_CONVERSATIONAL_REACT_DESCRIPTION	Chat geçmişine dayalı çok adımlı konuşmalar	Konuşma bağlamı takibi sağlar	Yeni sürümlerde StructuredChat tercih ediliyor	ConversationBufferMemory gerekir	"Fransa nerede?" → "Peki başkentinde ne meşhur?"

🎯 Hangi Durumda Hangi Agent Seçilmeli?

Kullanım Durumu	Önerilen Agent
Hızlı prototip, genel kullanım	ZERO_SHOT_REACT_DESCRIPTION
Adım adım bilgi toplama	SELF_ASK_WITH_SEARCH
Tool parametrelerini açıkça kontrol etme	STRUCTURED_CHAT_ZERO_SHOT_REACT_DESCRIPTION
Dış API'lerle güvenli ve kararlı tool kullanımı	OPENAI_FUNCTIONS
Wikipedia benzeri kaynaklardan bilgi çekme	REACT_DOCSTORE (opsiyonel, sorunlu olabilir)
Chat geçmişiyle bağlamlı yanıt üretme	CHAT_CONVERSATIONAL_REACT_DESCRIPTION