

Tasarım Örüntülerine Giriş

Tasarım Örüntüsü Nedir?

- Farklı ortamlarda/bağlamalarda karşımıza çıkan **benzer problemleri çözmek için uygulanan çözüm şablonlarıdır**
- Bu şablonlardaki **çözümün özü hep aynıdır**, ancak **şablonun uygulanışı** yapısal ve davranışsal olarak **her problem için farklılıklar** arz edebilir
- Bu şablonlar, zaman içerisinde benzer problemlere tekrar tekrar uygulanarak **evrilmiş ve olgunlaşmış, çalışırılığı** **sınanmış çözümlerdir**

Tasarım Örüntülerinin Çıkışı

- Tasarım örüntülerinin çıkışı **mimari ve antropolojiye** dayanır
- Christopher Alexander isimli mimar “**kalite nesnel veya subjektif bir olgu mudur?**” sorusuna cevap aramıştır
- Alexander'ın kanaati kaliteli/yaşanabilir mimarisel sistemleri teşhis etmek için **belirli nesnel kriterler söz konusudur**
- Benzer problemleri çözmek için inşa edilmiş yapılarda **kaliteyi belirleyen ortak nesnel özellikler vardır**

Tasarım Örüntülerinin Çıkışı

- Alexander, bu **ortak nesnel özelliklere** de **örüntü** (pattern) adını vermiştir
- Her bir örüntü öncelikle etrafımızda **sürekli olarak ortaya çıkan bir problemi** tanımlar
- Daha sonra da bu problem için **uygulanabilecek çözümün özünü** ortaya koyar
- Öyle ki bu **çözüm her seferinde farklı bir hal alabilir**

- Yazılım sistemlerini de **örüntüler etrafında tasarlamak** mümkün müdür?
- Bu konu üzerinde pek çok değişik çalışma yapılmıştır
- Ancak en çok ses getiren “**Design Patterns: Elements of Reusable Object-Oriented Software**” isimli kitap olmuştur

- Bu kitap örüntüler için **katalog** oluşturup, bunları **tasvir** etmeye çalışmıştır
- Kitapta **23 adet örüntü** katalog içerisine alınmıştır
- Yazarların bu örüntüleri **sıfırdan üretmediklerini** bilmek önemlidir
- Yaptıkları iş **değişik yazılım sistemlerini inceleyerek** bu örüntüleri tespit etmektir

- Örüntü tanımı **4 ana başlığa** sahiptir
 - Örüntünün ismi
 - Örüntünün amacı ve çözdüğü problem
 - Bu çözümün nasıl uygulanabileceği
 - Çözüm ile ortaya çıkan sonuçlar
- Hemen her tasarım problemi için örüntüler mevcuttur
- Örüntüler bir araya getirilerek **daha karmaşık problemleri çözmek** için de kullanılabilirler

Tasarım Örüntülerinin Faydaları

- Ekip içinde ve yazılım geliştiriciler arasında **ortak bir terminoloji** oluşmasını sağlar, ortak **bir bakış açısı** getirir
- Hazır çözümler probleme sıfırdan başlamayı, ve **olası hatalara düşmeyi önler**
- Çözümlerin **yeniden kullanılmasını** sağlar
- Diğer yazılım geliştiricilerin **deneyimlerinden faydalanmayı** sağlar

Tasarım Örüntülerinin Faydaları

- Bu örüntüler zaman içerisinde **evrilmiş ve olgunlaşmış** çözümlerdir
- Bu nedenle üzerlerinde **değişiklik yapmak daha kolay ve hızlıdır**
- Tasarım ve object oriented modelleme işlemine **üst perspektiften bakmayı** sağlar
- Bu sayede daha ilk aşamada **gereksiz detay ve ayrıntılar** içinde boğulmanın önüne geçilebilir

İletişim



www.harezmi.com.tr

www.java-egitimleri.com



info@harezmi.com.tr

info@java-egitimleri.com



[@HarezmiBilisim](https://twitter.com/HarezmiBilisim)

[@JavaEgitimleri](https://twitter.com/JavaEgitimleri)