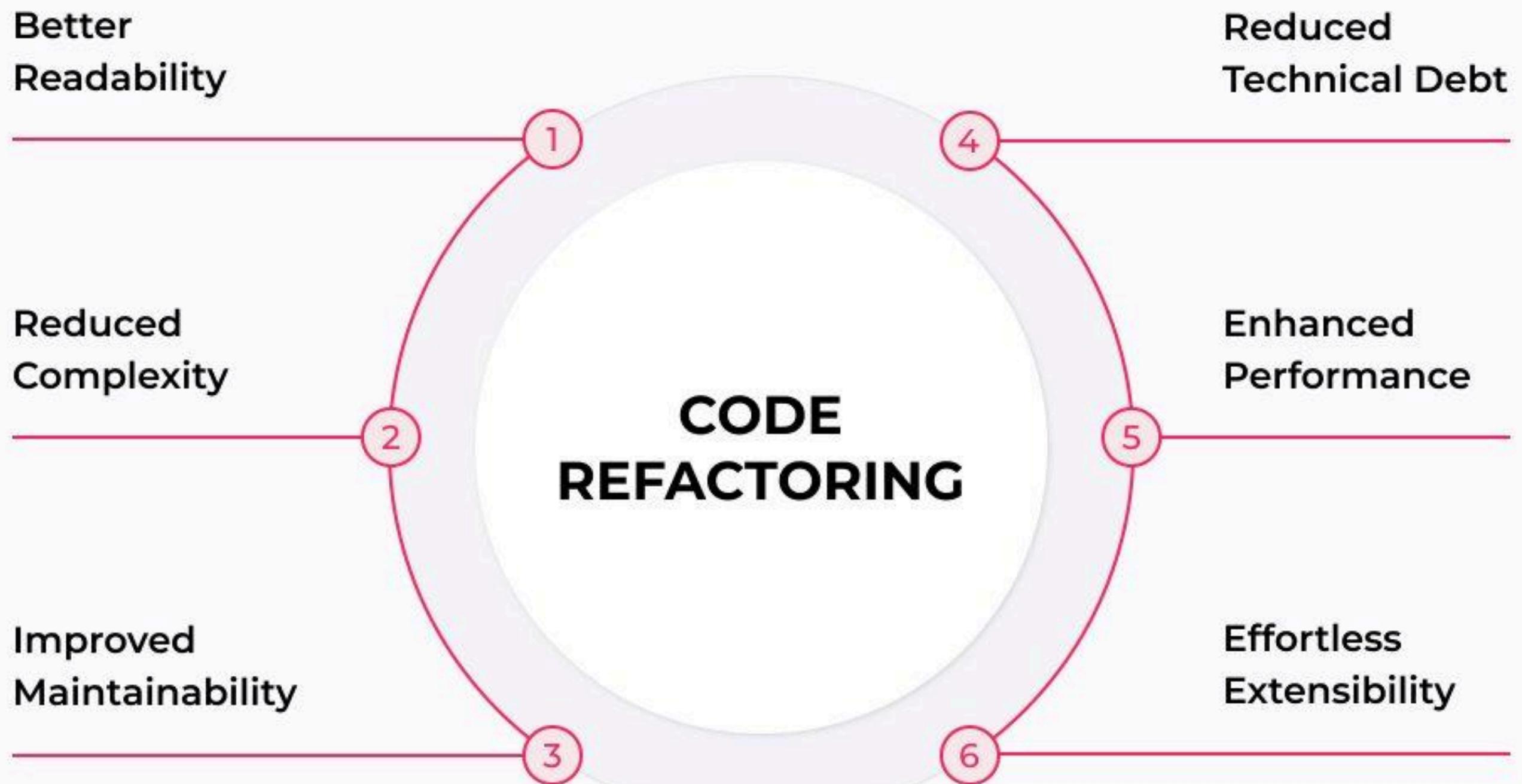


in/poyrazavsever



REFACTORING YAPIYOR MUSUNUZ?

Refactoring Nedir?





in/poyrazavsever

→|

Refactoring Nedir?

Refactoring, kodun davranışını değiştirmeden yapısını iyileştirme sürecidir.

Kod aynı işi yapar, ama artık daha temiz, düzenli ve okunabilir hale gelmiştir.

in/poyrazavsever

→|

```
● ● ●  
  
// Önce  
function hesapla(a, b, tip) {  
    if (tip === "topla") return a + b;  
    if (tip === "cikar") return a - b;  
}  
  
// Sonra  
const islemler = {  
    topla: (a, b) => a + b,  
    cikar: (a, b) => a - b  
};  
function hesapla(a, b, tip) {  
    return islemler[tip](a, b);  
}
```



in/poyrazavsever

→|

Refactoring'in Amacı

- Kodun bakımını kolaylaştırmak
- Hata riskini azaltmak
- Okunabilirliği artırmak
- Gelecekte geliştirmeyi hızlandırmak



in/poyrazavsever

→|

“Kod Çalışıyor” Yetmez

Bir kodun “çalışıyor” olması, onun iyi yazıldığı anlamına gelmez.

Zamanla büyüyen projelerde kötü yazılmış kod:

- Ekip hızını düşürür
- Hata riskini artırır
- Yeni özellikleri zorlaştırmır

in/poyrazavsever

→|

Refactoring Ne Zaman Gerekir?

1. Kod tekrarları arttıysa
2. Bir fonksiyon her şeyi yapıyorsa
3. Değişken isimleri karmaşıklaştıysa
4. Küçük değişiklik büyük hatalar çıkarıyorsa
5. “Buna dokunmayayım, bozulur” dediğin dosyalar varsa

Zamanı gelmiş demektir.

in/poyrazavsever

→|

Kötü Kodu Tanımak (Code Smell)

“Code smell” = kötü çalışan kod değil, kötü kokan kod.

Bir şeyler yanlış gidiyor demektir.

Örnek:

- Uzun fonksiyonlar
- Tekrarlayan kodlar
- Magic number'lar
- Zayıf isimlendirme

in/poyrazavsever



Magic Number Örneği

Bu küçük fark bile kodun kendini anlatmasını sağlar.



```
// Kötü
if (puan > 85) console.log("Başarılı");

// İyi
const BASARI_PUANI = 85;
if (puan > BASARI_PUANI)
console.log("Başarılı");
```

in/poyrazavsever

→|

Refactoring Türleri

1. Fonksiyon çıkarma (Extract Function)
2. Değişken isimlendirme (Rename Variable)
3. Tekrarlanan kodu ortaklaşdırma (DRY)
4. Sınıflara bölme (Split Class)
5. Uzun parametreleri sadeleştirme



in/poyrazavsever

→|

Refactoring Kültürü

Refactoring bir görev değil, bir alışkanlıktır.

Her commit öncesi küçük düzenlemeler yapmak, zamanla büyük fark yaratır.

Kodun çalışması yeterli değildir, anlaşılır ve geliştirilebilir olması gereklidir.



in/poyrazavsever

→|

Refactoring Kültürü

Refactoring = Aynı davranış, daha yüksek kalite.

Ben bir yazılımcı olarak gelişmemi yazdığım satırlar ile değil, **yeniden yazdığım satırların** kalitesi ile ölçüyorum.

Senin Refactoring Hikayen Ne?

Peki sen en son neyi refactor ettin?

Yorumlara yaz 



in/poyrazavsever

