

## Stok Kontrolü Yanlış

- addToCart metodunda bir breakpoint koydum ve quantity değerini takip ettim.
- Stok 1 olduğunda bile "Yetersiz stok!" hatası alıyordum.

### Hata Kodu:

```
if (product.stock <= quantity) {
```

- <= operatörü yanlış kullanılmış. Ürün miktarı 1'e eşit olduğunda da Yetersiz stok! uyarısı verecek.

### Çözüm:

- if (product.stock < quantity) { olarak değiştirilmelidir.

## Stok Güncellenmiyor

- removeItem metoduna breakpoint ekleyerek, ürün silindiğinde product.stock değerinin değişip değişmediğini gözlemledim.

### Hata Kodu:

```
const product = products.find(p => p.id === productId);
```

```
if (product) {
```

```
    product.stock += 1;
```

```
}
```

- removeItem metodunda, silinen ürünün quantity değeri kadar ürünün stoğu arttırılmalıdır.

### Çözüm:

- product.stock += item.quantity;

## Yanlış İndirim Hesaplama

- applyDiscount metoduna breakpoint ekleyerek, this.total değerinin nasıl değiştiğini izledim.
- İndirim sonrası total değerinin 10 kat düştüğünü fark ettim.

### Hata Kodu:

```
if (this.discountApplied && this.total > 0) {
```

```
    this.total *= 0.1;
```

```
}
```

- İndirim uygulandığında total yanlış hesaplanıyor, çünkü 0.1 ile çarpıldığı için fiyat 10 kat düşüyor. %10 indirim için 0.9 ile çarpılması gerekir.

### Çözüm:

```
if (this.discountApplied && this.total > 0) {
```

```
    this.total *= 0.9;
```

```
}
```

## app.addToCart Metodu Bulunamıyor

### Hata Kodu:

```
<button onclick="app.addToCart(${product.id})">
```

- App sınıfının içinde addToCart metodu var ancak window.app değişkeni üzerinden erişilmiyor.

### Çözüm:

- window.app.addToCart(\${product.id}) olarak güncellenmeli.

## DOMContentLoaded içindeki this Referansı

### Hata Kodu:

```
document.addEventListener('DOMContentLoaded', () => {  
  this.renderProducts();  
  this.setupEventHandlers();  
});
```

- Arrow function içinde this bağlamı korunmadığı için App sınıfına referans veremiyor.

### Çözüm:

- initializeEventListeners() içinde const self = this; kullanarak self.renderProducts(); self.setupEventHandlers(); şeklinde düzeltilmeli.

## Sepetim

Laptop	2 adet	15000 TL
Toplam: 15000 TL		
<input type="text" value="İndirim Kodu"/>	<input type="button" value="İndirimi Uygula"/>	
Geçersiz indirim kodu! Geçersiz indirim kodu!		

## Hata Mesajları Üst Üste Yazılıyor

- showError fonksiyonuna breakpoint ekleyerek her çağrıldığında errorElement.textContent değerini izledim.
- Konsolda hata mesajlarının birikerek arttığını fark ettim.

### Hata Kodu:

```
errorElement.textContent += message + '\n';
```

- showError fonksiyonunda hata mesajı her defasında birikerek artıyor.

**Çözüm:** errorElement.textContent = message;

## Toplam Fiyat Hesaplaması Yanlış

### Hata Kodu:

```
this.total = this.items.reduce((sum, item) => {  
  return sum + item.price;  
}, 0);
```

- price doğrudan toplandığı için ürün adedi (quantity) dikkate alınmıyor.

### Çözüm:

```
this.total = this.items.reduce((sum, item) => {  
  return sum + item.price * item.quantity;  
}, 0);
```

## Ürün Ekleme Sonrası Stok Güncellenmiyor

### Hata Kodu:

- addItem fonksiyonunda stoktan düşme işlemi yapılmıyor.

### Çözüm:

```
product.stock -= quantity;
```

## Sil Butonuna Basınca Tüm Ürün Siliniyor

- removeItem metoduna breakpoint koyarak, her çağrıldığında sepetin nasıl güncellendiğini gözlemledim.

### Hata:

- removeItem metodunda bir üründen birden fazla eklenmişse tümü birden siliniyor.
- removeItem metodu adet azaltmalı ve yalnızca adet 0 olduğunda ürünü tamamen kaldırmalı.

## "Satın Al" Butonu Eksik

### Hata:

- Sepette ürünler listeleniyor ama bunları satın almak için bir buton yok.
- updateUI metoduna bir "Satın Al" butonu eklenmeli ve satın alma işlemi için yeni bir metod yazılmalı.

## Diğer Hatalar:

- Dosya isimlerine dikkat edilebilir js ve html dosya isimleri aynı olmamalı.
- Ayrıca CSS düzenlemeleri ve iyileştirmeleri sağlanıp sepetteki adet kısımlarının bulunduğu sütun gibi kısımların tek sütunda olması sağlanabilir.