

FLUTTER

Kartlar, Temalar ve Stiller

14. hafta • Düzen ve Gezinme

36 hafta, 72 bağımsız çalışan Flutter projesi, gerçek uygulama ekranları, uygulamalı dersler ve öğrenci geri bildiriminden sonra incelenebilen çözüm kaynakları.

Bu kitap güncel ders kaynaklarından otomatik üretilmiştir; özet ya da jenerik şablon değildir.

Öğrenme Haritası

Hafta	Ders	Projeler
14	Kartlar, Temalar ve Stiller	Haber Kartları • Okuma Listesi

14. Hafta — Kartlar, Temalar ve Stiller

Bu hafta ne yapacağız?

Okul haberlerini tutarlı kartlar hâlinde gösteren bir uygulama geliştireceğiz. Renk, yazı ve kart stillerini tek tek her widget'a yazmak yerine ThemeData üzerinden yöneteceğiz. Koyu tema ve haber kaydetme davranışı ekledikten sonra aynı tasarım ilkeleriyle özgün okuma listesi yapacağız.

Kazanımlar

- Tekrarlanan içeriği model ve yeniden kullanılabilir kart düzeniyle sunma
- Renk şemasını ThemeData ve ColorScheme üzerinden yönetme
- Açık/koyu temayı ThemeMode ile değiştirme
- Duruma göre kart rengi, simge ve buton görünümü seçme
- Metin stillerini tema üzerinden türetme

Uygulama 14 — Haber Kartları

1. Tema oluşturma

```
theme: ThemeData(  
  colorScheme: ColorScheme.fromSeed(seedColor: const Color(0xff00639b)),  
  useMaterial3: true,  
  cardTheme: const CardThemeData(elevation: 1, margin: EdgeInsets.zero),  
)  
darkTheme: ThemeData(  
  colorScheme: ColorScheme.fromSeed(  
    seedColor: const Color(0xff91cbff),  
    brightness: Brightness.dark,  
  ),  
  useMaterial3: true,  
)
```

Tema renkleri tek merkezden üretildiği için açık ve koyu yüzeylerde metin karışıklığı korunur.

2. Tema modunu değiştirme

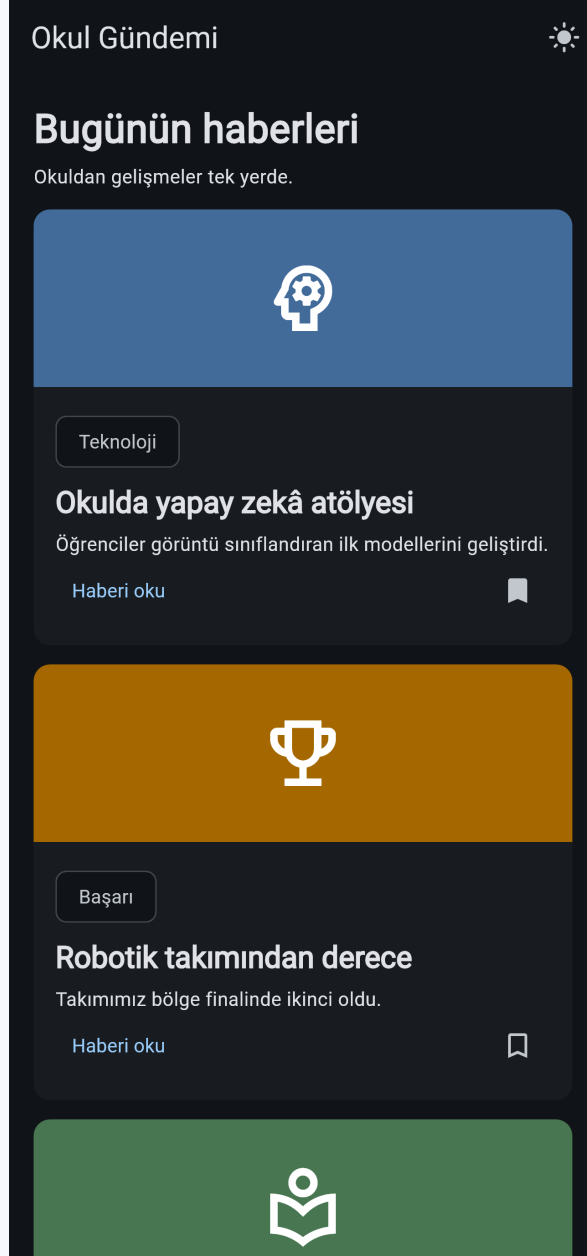
```
bool koyu = false;  
  
MaterialApp(  
  themeMode: koyu ? ThemeMode.dark : ThemeMode.light,  
)
```

AppBar düğmesi bu değeri setState ile değiştirir. Tema durumu MaterialApp seviyesinde tutulduğu için bütün ekran birlikte yenilenir.

3. Modelden kart üretme

```
class Haber {  
  Haber(this.baslik, this.ozet, this.kategori, this.simge, this.renk);  
  final String baslik, ozet, kategori;  
  final IconData simge;  
  final Color renk;  
  bool kayitli = false;  
}
```

Kaydet düğmesi kayıtlı durumuna göre boş veya dolu yer imi gösterir. Testte ilk haber kaydedildi ve koyu temaya geçildi.



Kaydedilmiş haberle koyu temadaki Okul Gündemi

Özgün proje — Okuma Listesi

Kitap kartları aynı yapıyı kullanır; yalnız model verileri değişir:

```
class Kitap {
  Kitap(this.ad, this.yazar, this.sayfa, this.okunan);
  final String ad, yazar;
  final int sayfa;
  int okunan;
}
```

İlerlemeyi göstermek

```
final bitti = kitap.okunan >= kitap.sayfa;
final oran = kitap.okunan / kitap.sayfa;

LinearProgressIndicator(value: oran.clamp(0, 1));
```

Tamamlanan kitapta kart, kapak simgesi ve ekleme düğmesi farklı görünür. Renkler sabit açık tonlardan değil, mevcut temanın `colorScheme` değerlerinden alınır:

```
color: bitti ? Theme.of(context).colorScheme.primaryContainer : null
```

Okuma olayını işlemek

```
onPressed: bitti
  ? null
  : () => setState(() {
    kitap.okunan = (kitap.okunan + 20).clamp(0, kitap.sayfa);
  })
```

Testte Şeker Portakalı 86 sayfadan 106 sayfaya ilerletildi, koyu temaya geçildi ve tamamlanan kitap sayısının 1/3 kaldığı doğrulandı.



Koyu temada ilerlemesi güncellenmiş Okuma Listesi

Sıra sende

1. Kolay: Haber kartlarına tarih bilgisi ekle.
2. Orta: Okuma listesini “Tümü / Devam / Tamamlandı” filtreleriyle süz.
3. İleri: Kullanıcının seçtiği vurgu renginden hem açık hem koyu ColorScheme üret ve düşük karışıklıklı renklerde erişilebilir alternatif uygula.

Çözümünü gönderdikten ve geri bildirim aldıktan sonra Okuma Listesi’nin gerçek Dart kodu ile çalışan proje paketi açılacaktır. İpucu tema ve durum rengini ayırmana yardım edecek; çözüm düğmesi eksiksiz `lib/main.dart` kodunu gösterecektir.

Kontrol listesi

- Ortak renk ve kart stilleri tema içinde tanımlandı.
- Koyu tema yalnız arka planı değil bütün renk şemasını değiştiriyor.

- Kartlar model listesinden üretiliyor.
- Tamamlanan ve devam eden içerik görsel olarak ayırt ediliyor.
- İki proje Flutter widget testinden geçti ve gerçek açık/koyu ekran çıktıları görsel olarak doğrulandı.