



CAN TAŞPINAR

FLUTTER'DA STATE YÖNETİMİ VE PROVIDER

FLUTTER'DA STATE YÖNETİMİ VE PROVIDER

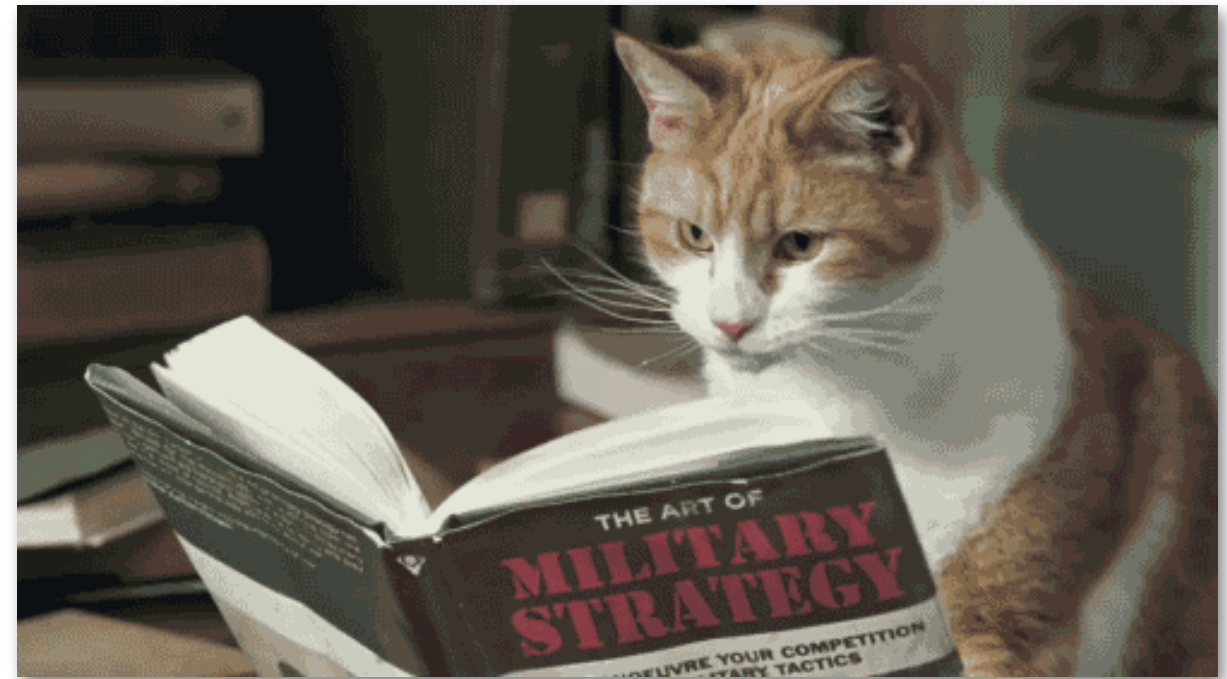
- ▶ State Yönetimi Nedir?
- ▶ Declarative Programming ve State Çeşitleri
- ▶ State Yönetimi Çözümleri - Provider
- ▶ Provider ile Alışveriş Sepeti Örneği

STATE NEDİR ?

- ▶ Uygulamanız çalışırken hafızada tutulan veriler - UI, Animasyonlar, Network, Business Logic
- ▶ Arayüzü oluşturmak ve güncellemek için ihtiyacınız olan veri
- ▶ Kendinizin yönettiği state - Kullanıcı Girişi, Uygulama Teması, Alışveriş Sepeti ...

STATE YÖNETİMİ NEDİR – NEDEN GEREKLİ ?

- ▶ Birşeyler değiştiğinde widget'ı nasıl güncellerim?
- ▶ Ekranlar ve Widget'lar arasında nasıl veri paylaşabilirim?
- ▶ Bir widget state'ni başka bir widget ile nasıl değiştirebilirim?



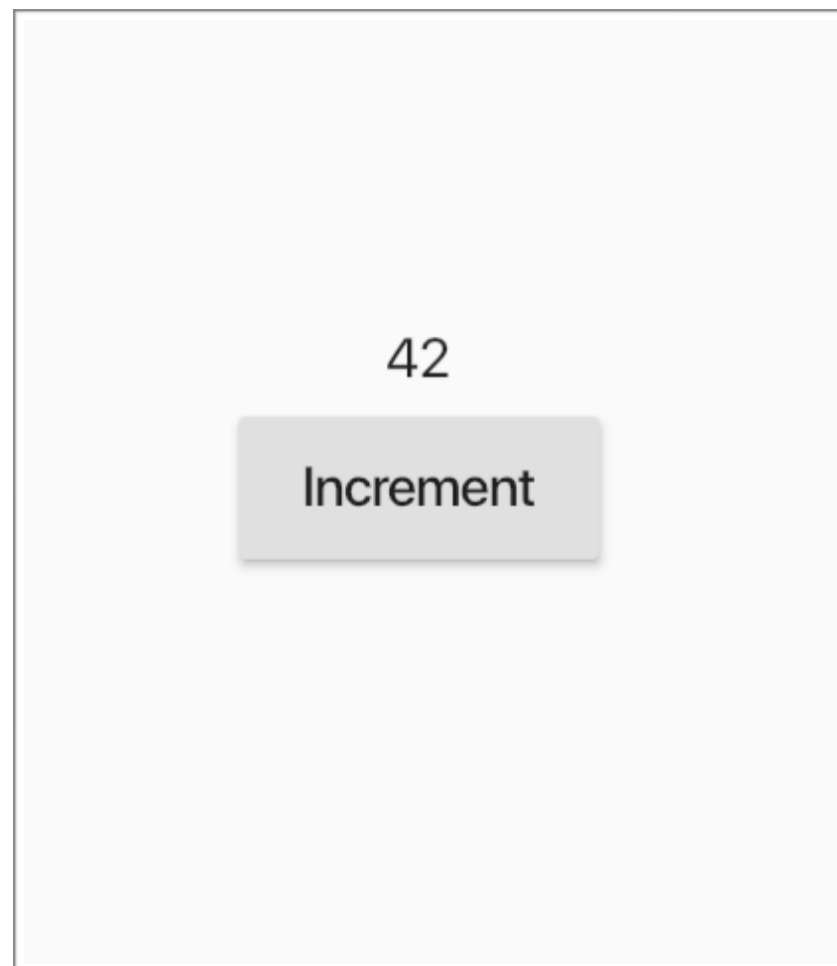
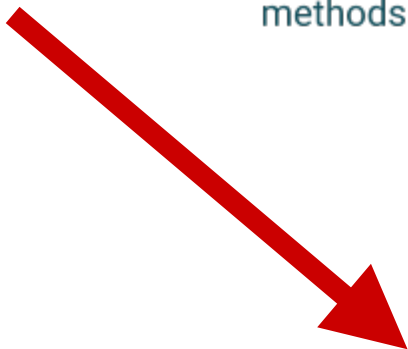
DECLARATIVE PROGRAMMING

UI = f(state)

The layout
on the screen

Your
build
methods

The application state



```
class CoolCounter extends StatefulWidget {  
  @override  
  _CoolCounterState createState() => _CoolCounterState();  
}  
  
class _CoolCounterState extends State<CoolCounter> {  
  int _counter = 0;  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      body: Center(  
        child: Column(  
          mainAxisAlignment: MainAxisAlignment.center,  
          children: <Widget>[  
            Text('${_counter}'),  
            RaisedButton(  
              child: Text("Increment"),  
              onPressed: () {  
                setState(() {  
                  _counter++;  
                });  
              },  
            ) // RaisedButton  
          ], // <Widget>[]  
        ), // Column  
      ), // Center  
    ); // Scaffold  
  }  
}
```

STATE ÇEŞİTLERİ

- ▶ Local State
- ▶ Global State

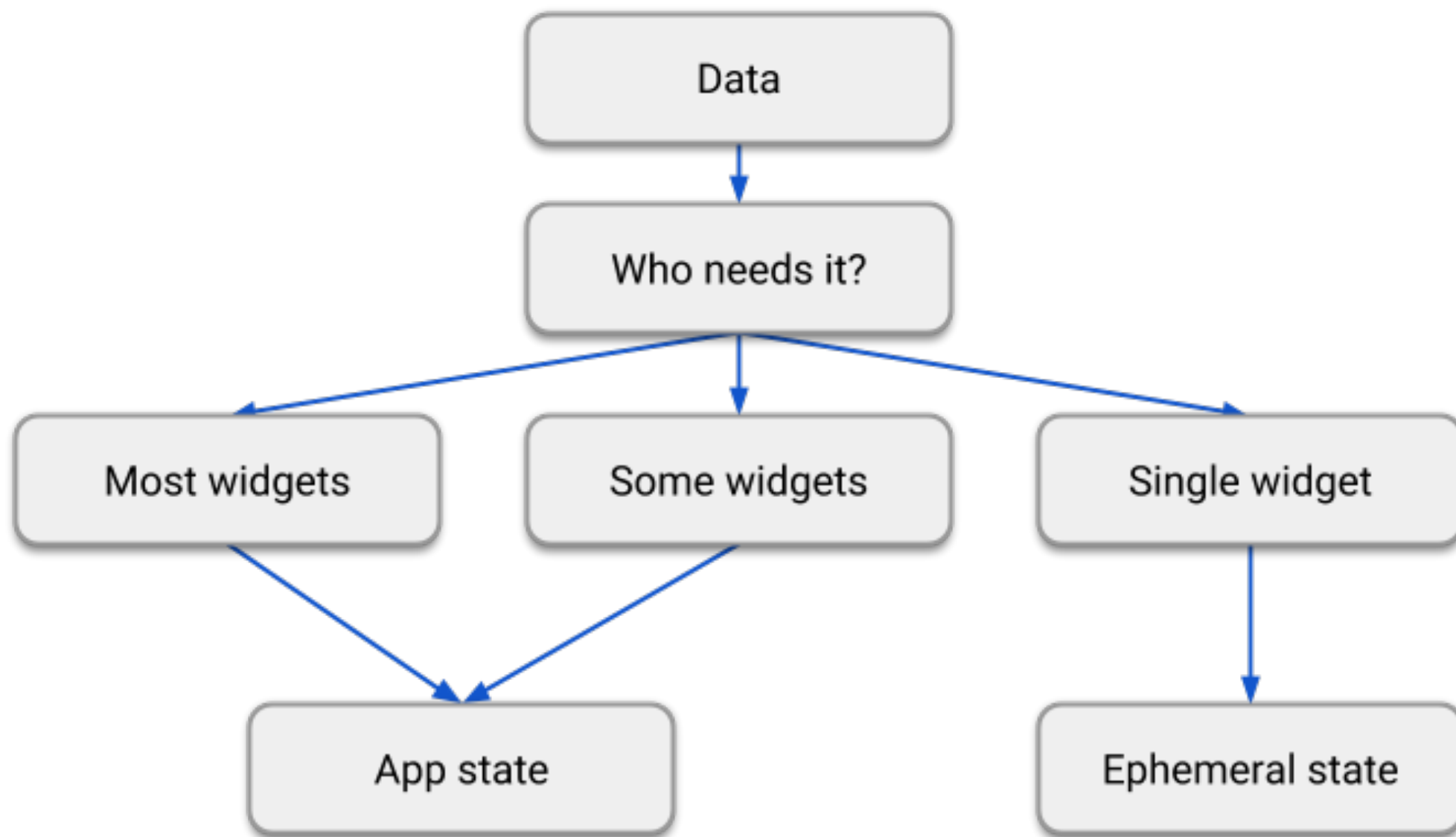
LOCAL STATE

- ▶ Tek bir widget'ta kullanılan veri
- ▶ Bu veriye başka widgetların'da erişmesi gerekiyor mu? (Hayır)
- ▶ Karışık yöntemlere ve ek araçlara gerek yok
- ▶ StatefulWidget ve setState() yeterli

GLOBAL / APP / SHARED STATE

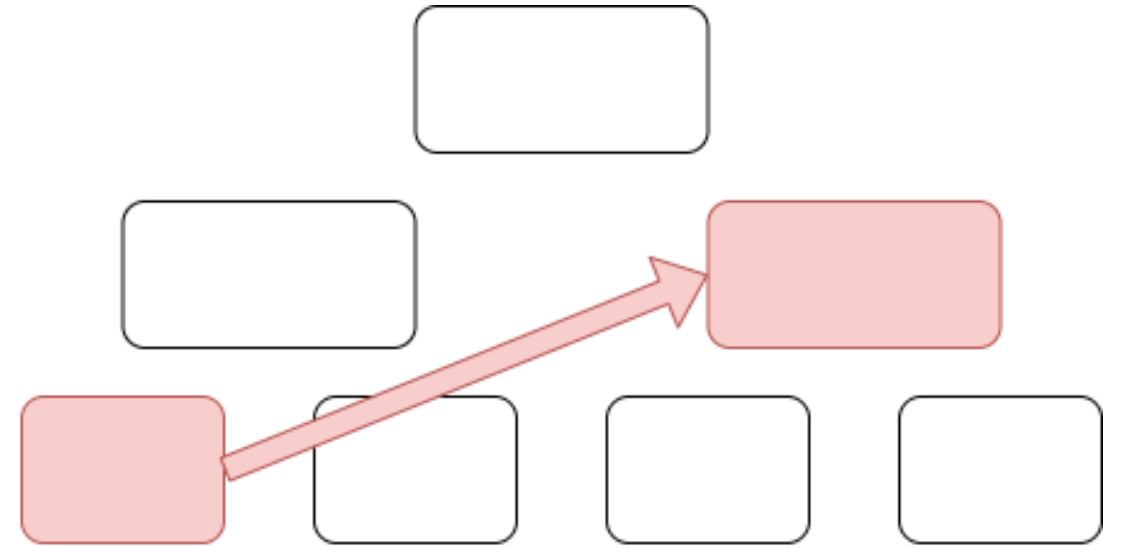
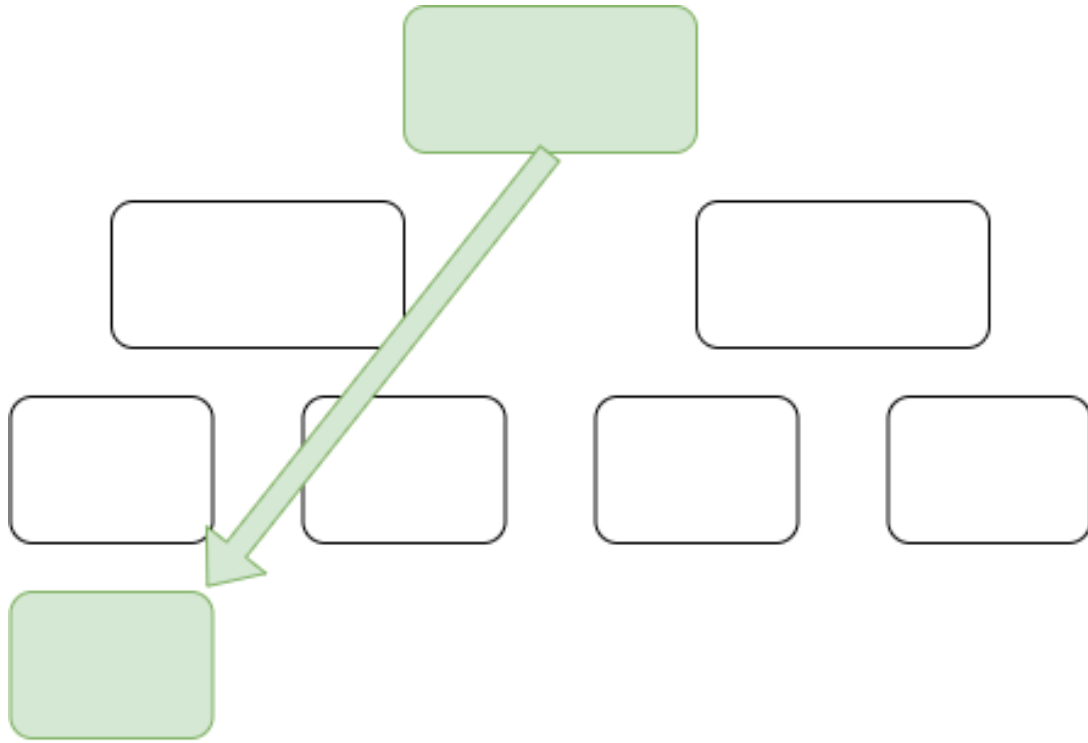
- ▶ Birden fazla widget'ta kullanılan veri
- ▶ Bu veriye farklı widget'lardan erişmek istiyor muyum? (Evet)
- ▶ Örnekler: Kullanıcı giriş bilgisi, Kullanıcı tercihleri, Bildirimler, Alışveriş Sepeti

LOCAL STATE VEYA GLOBAL STATE ?



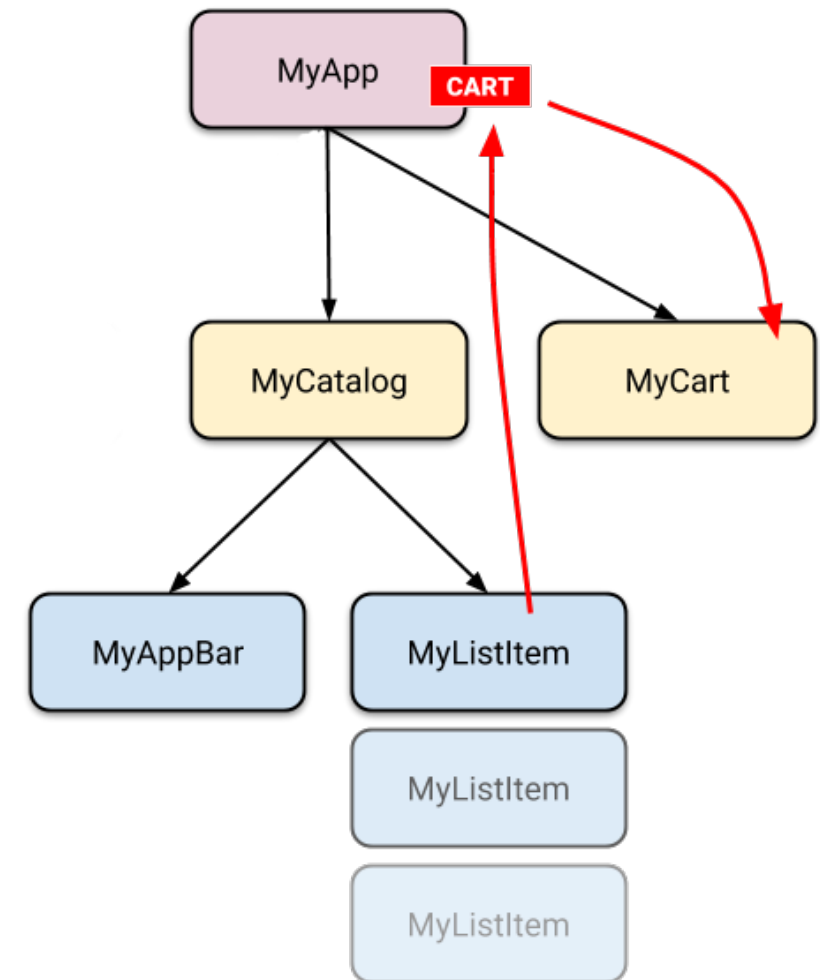
AİLE YADİGARİ

Aşağıdan yukarıya veri akışı

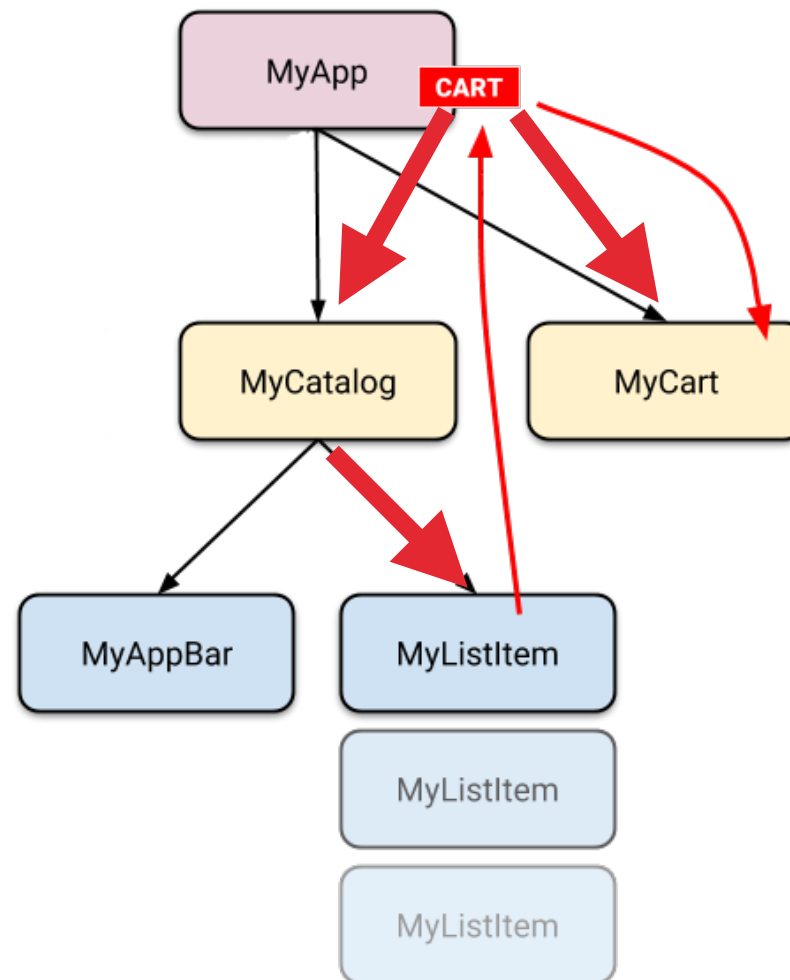


LIFTING STATE UP

- ▶ State'i onu kullanan widget'ların üstünde tutun
- ▶ State değiştiğinde, ilgili çocuk widget'lar arayüzü güncellemek için tekrar yaratılır
- ▶ Widget'lar düzenlenmez, yenisi ile değiştirilir
- ▶ Bir widget başka bir widget'ı direkt olarak değiştirmemeli



STATE'E GİDEN YOL





EN İYİ YÖNTEM?

- ▶ Uygulamanın yapısına ve ihtiyaçlarına bağlı
- ▶ Basit tutmaya çalışın
- ▶ Çok tartışılan bir konu, kavga çıkabilir

**“HANGİSİ DAHA AZ GARİP
GELİYORSA ONU YAPIN”**

Redux, Dan Abramov,

SEÇENEKLER ?

Scoped Model

RxDart

Redux

BLoC

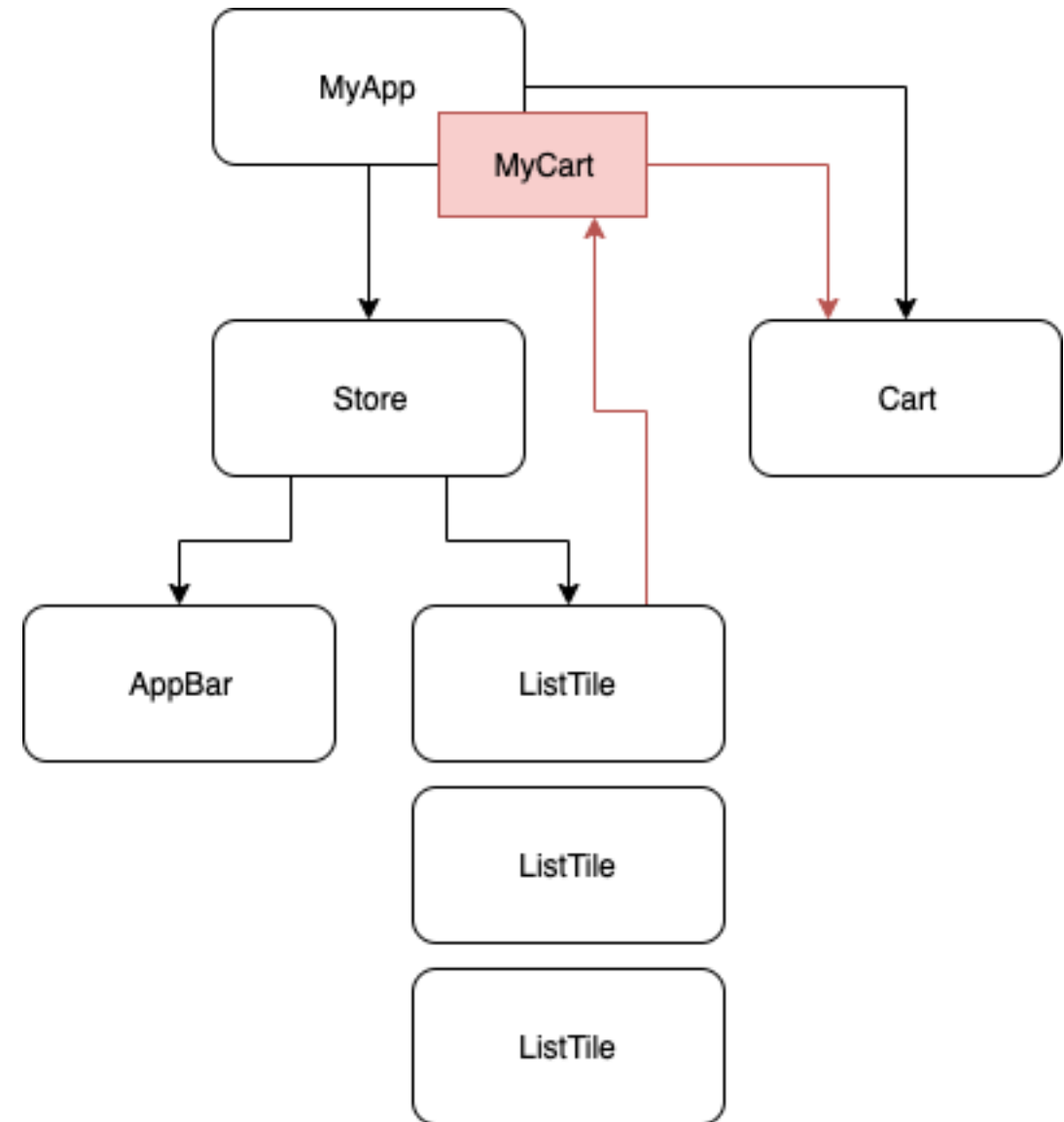
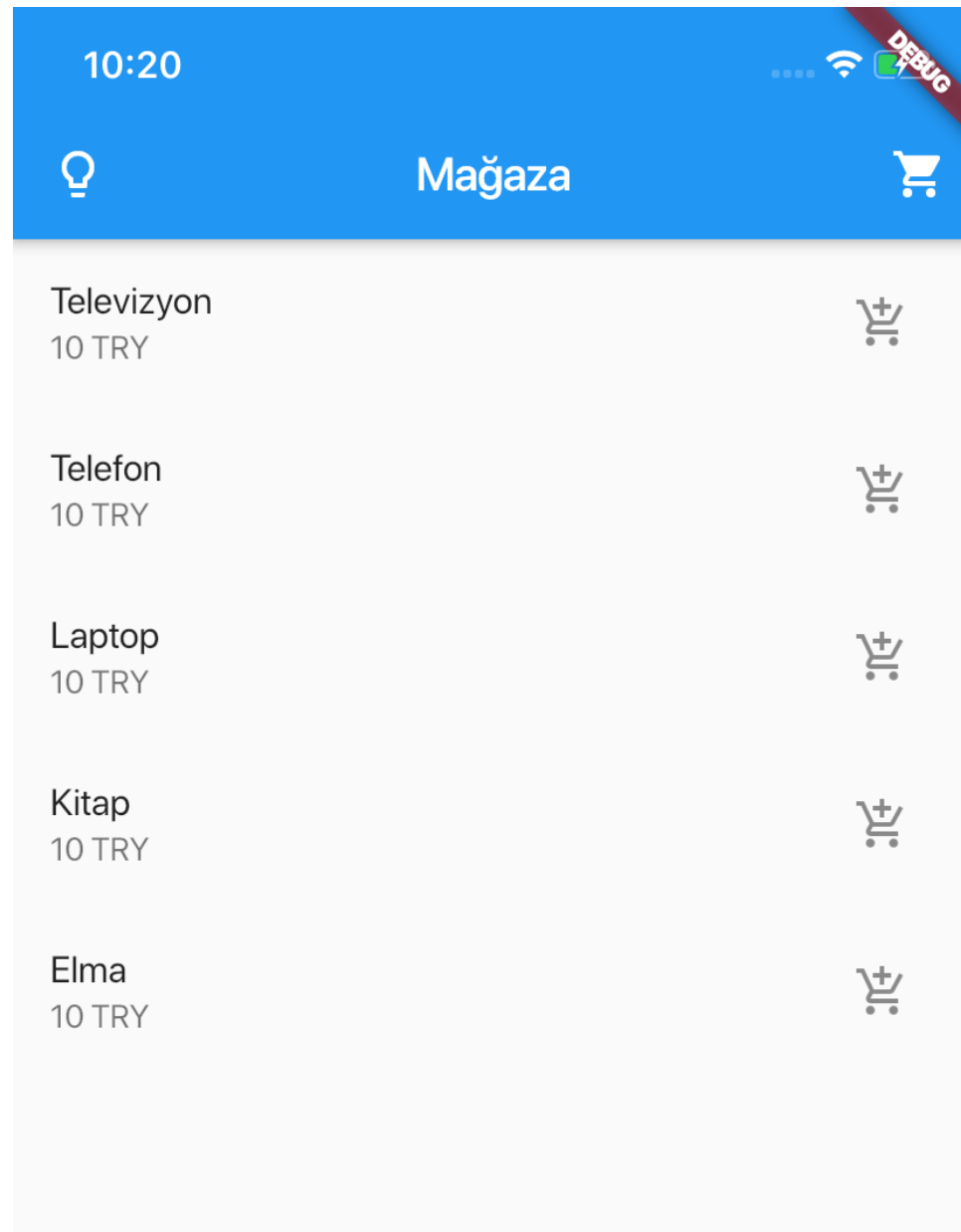
Provider

PROVIDER

- ▶ State yönetimi konusunda Flutter ekibi tarafından önerilen çözüm
- ▶ Anlaşılması kolay, fazla koda ihtiyaç yok
- ▶ Basit ve uygulamanız için muhtemelen yeterli

<http://bit.ly/sepetdemo>

PROVIDER İLE ALIŞVERİŞ SEPETİ ÖRNEĞİ



KAYNAKLAR

<https://flutter.dev/docs/development/data-and-backend/state-mgmt> - Flutter Docs, State Management

<https://fireship.io/lessons/flutter-state-management-guide/> - Flutter State Management Guide

https://www.youtube.com/watch?v=d_m5csmrf7I - Pragmatic State Management

<https://www.youtube.com/watch?v=zKXz3pUkw9A> - Keep it Simple, State