



Lesson(9) [State & Interactivity]

تحفيز: الغلط في الكود مش عيب، ده تعليم

StatelessWidget

◆ يعني إيه؟

بيكون "ثابت"، يعني مش بيحتفظ بحالة أو بيانات بتتغير Widgets ده نوع من الـ من النوع ده، مش هيتغير شكله أو بياناته طول ما التطبيق Widget ببساطة، لما تعمل شغال، إلا لو رجعت تبنيه من الأول.

مثال:

```
class MyBox extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Text("أنا ويدجت ثابت");  
  }  
}
```

 ملاحظات:

- مفيش أي حاجة بتتغير جواه.
- "بنستخدمه لما نكون بنعرض بيانات "مش متغيرة".

🔑 ملخص سريع 🧠:

- **StatelessWidget**: مش بيتغير، ويدجت ثابت، مش بيتغير.
- بنكتبه لما نكون مش محتاجين نغير في الواجهة وقت التشغيل.
- مفيهوش `setState()`.

تحفيز: كل كود بتكتبه، هو تدريب لعقلك

🔄 StatefulWidget

◆ يعني إيه؟

ده ويدجت "ديناميكي" 🏠، يعني بيتغير وقت ما التطبيق شغال.

زي لما تدوس) بنستخدمه لما نحتاج نغير شكل الواجهة أو البيانات حسب تفاعل المستخدم (TextField على زرار، أو تكتب حاجة في

📦 بيكون جزئين StatefulWidget هيكل الـ

```
class MyWidget extends StatefulWidget {  
  @override  
  State<MyWidget> createState() ⇒ _MyWidgetState();  
}
```

```
class _MyWidgetState extends State<MyWidget> {  
  // اللي ممكن تتغير (state) هنا هنحط الحالة  
  int counter = 0;  
  
  @override  
  Widget build(BuildContext context) {  
    return Column(  
      children: [  
        Text("العداد: $counter"),  
      ],  
    );  
  }  
}
```

```

ElevatedButton(
  onPressed: () {
    setState(() {
      counter++; // كل مرة بندوقس، العدد بيزيد
    });
  },
  child: Text("زود"),
),
],
);
}
}

```

؟ **setState()** إيه هو

- دي الفانكشن اللي بنستدعيها لما نغيّر أي حاجة في الحالة.
- "أنا غيرت حاجة، ابني الواجهة من ثاني" Flutter بتقول لـ.

```

setState(() {
  // هنا نغير في البيانات
});

```

ملخص سريع:

- **StatefulWidget**: ويدجت متغير بيتفاعل مع المستخدم.
- ويحتفظ ببيانات داخل الكلاس الثاني `build()` و `createState()` فيه.
- علشان التغيير يظهر على الشاشة `setState()` لما تحب تغير حاجة، تستخدم.

تحفيز: في البرمجة لما تلاقي الحل بنفسك، فرحتك بتكون أكبر