



Lesson(5) [Images & Icons]

تحفيز: البرمجة مش مجرد شغل، دي حياة مليانة إبداع



Image.asset


✓ **Image.asset** يعني إيه ؟

هي ويدجت بتعرض صورة محفوظة جوا ملفات المشروع بتاعك.

يعني لو عندك صورة في مجلد داخل المشروع، وعايز تعرضها في التطبيق → تستخدم

Image.asset

 **إزاي أضيف صورة وأستخدمها؟**

 **أنشئ مجلد للصور 1.**

افتح مشروعك، واعمل فولدر مثلاً

```
assets/images/
```

وخط جواه الصورة، وليكن

```
logo.png
```

🚧 2. عرّف الصورة في ملف `pubspec.yaml`

اللي في جذر المشروع، ودور على الجزء ده `pubspec.yaml` افتح ملف

```
flutter:
```

ضيف تحته (بمسافتين) السطر ده

```
assets:  
  - assets/images/
```

!خلي بالك من المسافات

لازم السطر يكون داخل بمسافتين تحت `flutter:`

حساس للمسافات جدًا YAML عشان الـ

🚧 3. استخدم الصورة في الكود بـ `Image.asset`

```
Image.asset('assets/images/logo.png')
```

وبكده الصورة هتظهر في التطبيق ✅

🔑 أهم الخصائص (Attributes) لـ `Image.asset`

🔧 `width` و `height`

لتحديد عرض وارتفاع الصورة

```
Image.asset(  
  'assets/images/logo.png',  
  width: 100,
```

```
height: 100,  
)
```



عشان نتحكم في طريقة عرض الصورة جوا المساحة

```
Image.asset(  
  'assets/images/logo.png',  
  fit: BoxFit.cover, // تغطية كاملة  
)
```

🧠 أهم القيم:

- `BoxFit.cover`: تغطي كل المساحة
- `BoxFit.contain`: تبين كل الصورة بدون ما تتقص
- `BoxFit.fill`: تملأ المساحة تمامًا حتى لو الصورة تشوهت



تحدد مكان الصورة جوا المساحة

```
Image.asset(  
  'assets/images/logo.png',  
  alignment: Alignment.centerRight,  
)
```

⚡ ملخص سريع 🎯

- `Image.asset`: بتعرض صورة من ملفات المشروع.
- `pubspec.yaml`: لازم تضيف المسار بتاع الصور في.
- أهم الخصائص:
 - `width`, `height`: حجم الصورة.

- طريقة عرض الصورة : `fit`.
- مكان الصورة في الشاشة : `alignment`.

تحفيز: كل سطر كود بتكتبه، بيخليك أقرب لهدفك



Image.network

✓ **Image.network** يعني إيه ؟

هي ويدجت بتعرض صورة من على الإنترنت، مش من ملفات المشروع

تقدر تعرضه بسهولة في التطبيق من غير ما تنزله محليًا، (URL) يعني لو عندك لينك صورة



Image.network كود بسيط لاستخدام

```
Image.network('https://example.com/image.png')
```

وهي دي كل السحر

بيجيب الصورة من الإنترنت ويعرضها Flutter، بمجرد ما تحط اللينك

🔑 **Image.network** لـ (Attributes) أهم الخصائص



width و **height**

عشان تتحكم في حجم الصورة

```
Image.network(  
  'https://example.com/image.png',  
  width: 150,  
  height: 150,  
)
```



fit

زي ما شرحنا قبل كده، بيتحكم في طريقة عرض الصورة

```
Image.network(
  'https://example.com/image.png',
  fit: BoxFit.cover,
)
```

🧠 أهم القيم:

- `BoxFit.cover`: تغطي المساحة وتقص من الصورة.
- `BoxFit.contain`: تبين الصورة بالكامل.
- `BoxFit.fill`: تمتد الصورة تملأ المساحة.



loadingBuilder

تقدر تستخدمه عشان تعرض ويدجت مؤقتة (مثلاً لودر) أثناء تحميل الصورة

```
Image.network(
  'https://example.com/image.png',
  loadingBuilder: (context, child, loadingProgress) {
    if (loadingProgress == null) return child;
    return CircularProgressIndicator(); // تحميل لحد ما الصورة تيجي
  },
)
```



errorBuilder

لو حصلت مشكلة والصورة ما اتحملتش، تقدر تعرض ويدجت بديلة

```
Image.network(
  'https://example.com/image.png',
  errorBuilder: (context, error, stackTrace) {
    return Icon(Icons.error); // لو حصل خطأ
  },
)
```

⚡ ملخص سريع 🎯

- URL بتعرض صورة من الإنترنت عن طريق: `Image.network`.
- سريعة وسهلة في الاستخدام.
- أهم الخصائص:
 - تنسيق عرض الصورة: `fit`.
 - ويدجت تظهر أثناء التحميل: `loadingBuilder`.
 - ويدجت تظهر لو حصلت مشكلة: `errorBuilder`.
 - للتحكم في الحجم: `width` / `height`.

تحفيز: كل تحدي في البرمجة هو فرصة جديدة لتتعلم وتنمو

`Image.file`

✅ يعني إيه؟

بتستخدمها لما تكون الصورة موجودة على **جهاز المستخدم** (يعني اتخزنت محلياً `Image.file`) مثلاً بعد ما المستخدم صور بكاميرا أو اختار صورة من المعرض.

كود استخدام بسيط:

```
File imageFile = File('path/to/image.jpg');  
  
Image.file(imageFile)
```

| `import 'dart:io'` ول لازم يكون عندك 🙏

```
import 'dart:io';
```

إمتي تستخدمها؟

لما تجيب صورة من

- الكاميرا
- المعرض (Gallery)
- أو أي مكان تخزن فيه الصور جوه الجهاز

🧠 أهم الخصائص (نفس اللي فاتت تقريبا)

- `fit`
- `width` / `height`
- لو حصلت مشكلة في عرض الصورة `errorBuilder`

🧠 ملخص سريع 🎯

- لعرض صورة من على جهاز المستخدم : `Image.file`
- `File` لازم تستخدم
- `import dart:io` بتحتاج

تحفيز: البرمجة هي اللغة الوحيدة اللي بتوصلك لعالم مختلف

🧠 `Image.memory`

✅ يعني إيه؟

وده بيكون مفيد لو ، **(bytes)** بتعرض صورة من البيانات الخام `Image.memory`

- bytes ك API الصورة جاية من
- أو الصورة متخزنة بصيغة `Uint8List`

📦 كود بسيط:

```
Image.memory(imageBytes)
```

| `Uint8List` من نوع `imageBytes` وللازم تكون

مثال عملي:

```
Uint8List imageBytes = // مثلاً API البيانات جاية من
Image.memory(imageBytes)
```

الخصائص المهمة:

نفس الخصائص اللي اتكلمنا عنها قبل كده

- `fit`
- `width` / `height`
- `errorBuilder`

ملخص سريع

- `Image.memory` : bytes بتعرض صورة من
- API مفيدة جداً مع الصور اللي جاية من
- `Uint8List` بتستخدم

تحفيز: مع البرمجة، المستقبل بين إيديك




يعني إيه؟

يعني أيقونات زي: نجمة، Material Icons هي ويدجت بتعرض أيقونات جاهزة من مكتبة `Icon` (سهم، بيت، الخ...)

دي بتستخدمها كتير في التطبيقات عشان تدي معاني مرئية، زي زر الرجوع، أو رمز الإعدادات

مثال بسيط:

```
Icon(Icons.home)
```

"دي هتظهرلك أيقونة البيت .

🔑 أهم الخصائص (Attributes):



color

تغيير لون الأيقونة

```
Icon(Icons.star, color: Colors.yellow)
```



size

تحديد حجم الأيقونة

```
Icon(Icons.star, size: 40)
```



semanticLabel

لو عايز توصف الأيقونة لبرامج قراءة الشاشة (لذوي الاحتياجات)
لو عايز تشوف الوصف اضغط علي الأيقونة ضغطة مطولة

```
Icon(Icons.star, semanticLabel: 'نجمة التقييم')
```

⚡ ملخص سريع 🎯

- بتعرض أيقونة من مكتبة جاهزة: `Icon`.
- لتغيير اللون: `color`.
- للتحكم في الحجم: `size`.
- لو عايز تضيف وصف صوتي للأيقونة: `semanticLabel`.

تحفيز: لما الكود يشتغل، بتحس إنك سيطرت على الدنيا



CircleAvatar




يعني إيه؟

هي ويدجت بتعرض صورة أو حرف داخل دائرة، وغالبًا بنستخدمها في صور `CircleAvatar` البروفایل أو المستخدمين.

مثال بسيط:

```
CircleAvatar(  
  backgroundImage: AssetImage('assets/images/user.png'),  
)
```

دي هتجيب صورة من المشروع وتعرضها جوا دايرة .

أهم الخصائص (Attributes):



`backgroundImage`

`Network` : أو من `assets` تحط صورة من

```
CircleAvatar(  
  backgroundImage: NetworkImage('https://example.com/user.jpg'),  
)
```



`child`

لو مش حاطط صورة، ممكن تحط حرف أو أي ويدجت جواه:

```
CircleAvatar(  
  child: Text('A'),  
)
```



`backgroundColor`

تغير لون الخلفية:

```
CircleAvatar(  
  backgroundColor: Colors.blue,
```

)



radius

تتحكم في حجم الدائرة

```
CircleAvatar(  
  radius: 40,  
)
```

⚡ ملخص سريع 🎯

- **CircleAvatar**: دائرة بتعرض صورة أو حرف.
- **backgroundImage**: عشان تعرض صورة.
- **child**: تحط محتوى جوا الدائرة لو مفيش صورة.
- **radius**: تتحكم في الحجم.
- **backgroundColor**: تغير لون الخلفية.

تحفيز: الكود هو السلاح السري للمبدعين