

Flutter Pokédex

Setup your IDE

1. [Download and install the flutter SDK](#)
2. [Setup your favorite code editor](#)
3. Optional: [Make sure everything works](#)

Clone repository

Clone [the GitHub repository](#) to get started.

Start the Pokédex

TODO: Install deps and run on emulator.

Build your Pokédex

This is your Pokédex now.

It isn't pretty and you can't scroll.

Let's fix that.



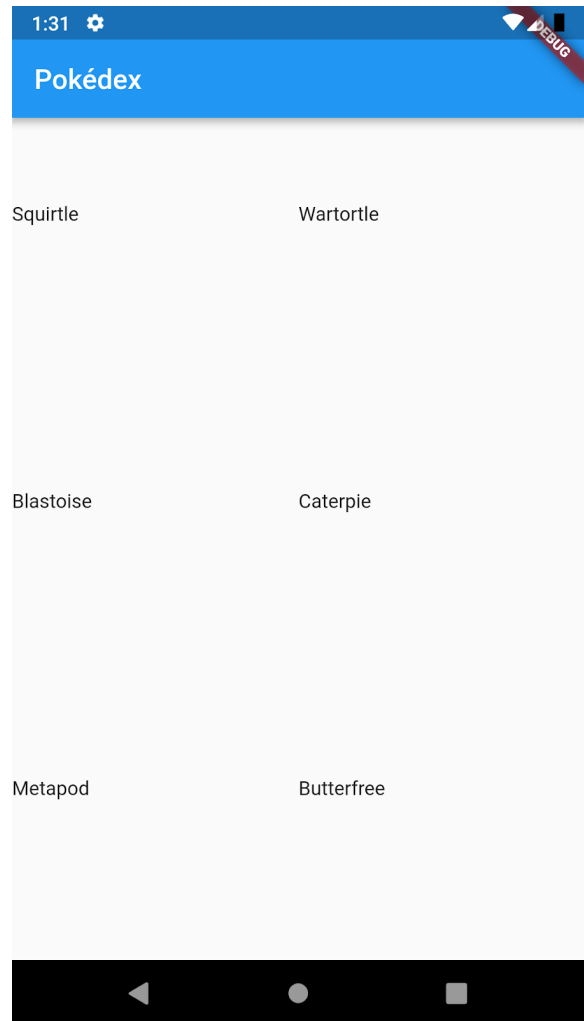
1) Using a GridView

Use a GridView to enable scrolling and show the name of the Pokémons in a grid.

Have a look at the [GridView cookbook](#) for inspiration.

You can also have a look at the [API documentation](#), but it is quite extensive.

The final result should look something like what you see on the right.



2) Your first widget

We want to show more than just the name of the Pokémon. Let's start by creating a widget for that.

Create a stateless widget called `PokemonGridItem`.

The widget should take a `Pokemon` as a constructor argument and save it in a property on the class.

Use the `Text` widget to show the name of the Pokémon.

Use your `PokemonGridItem` in the `GridView`.

An example of a basic stateless widget:

```
class MyWidget extends StatelessWidget {  
  ...  
  
  @override  
  Widget build(BuildContext context) {  
    return Text(...);  
  }  
}
```

3) Showing images

Use the `Column` and `CachedNetworkImage` widgets to show the Pokémon images.

Have a look at the [Image cookbook](#) for inspiration.

Hint: The `cached_network_image` package, is already installed in the project.

Go crazy with styling.

Try using the `Container` widget to add some padding and colors.

Have a look at the [widget catalog](#) for inspiration.

The final result could look something like what you see on the right.



4) Showing details

The `Pokemon` class contains more information than we can show in the grid view.

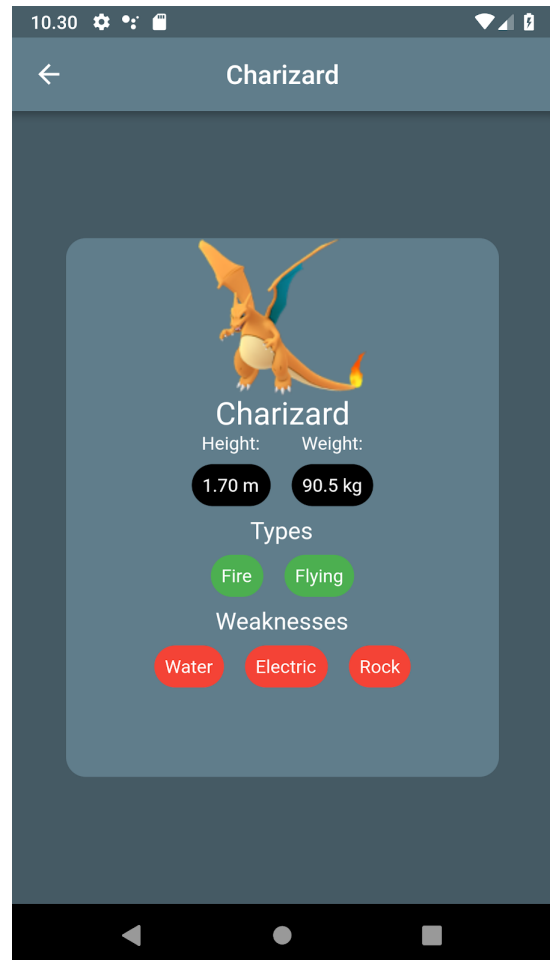
Let's create a screen where we can show more details such as the height, weight and weaknesses of the Pokémon.

Create a stateless widget called `DetailsScreen`. It should get a `Pokemon` just like the `PokemonGridItem` and show all its details.

The Pokédex should open this screen when you tap on a Pokémon in the grid view.

Use the `GestureDetector` widget for detecting taps. Have a look at the [GestureDetector cookbook](#) for inspiration.

Use the `Navigator` to navigate to the details screen on tap. Have a look at the [Navigator cookbook step 2](#) for inspiration.



5) Animations

To add a hero animation to your pokémon images, wrap the images in a `Hero` widget.

Give the `Hero` widget a unique tag, like the pokémon's name or number.

```
Hero(
  child: CachedNetworkImage(imageUrl: pokemon.image),
  tag: pokemon.number,
),
```