# API Example Code

This directory contains the API sample code that is referenced from the API documentation in the framework.

The examples can be run individually by just specifying the path to the example on the command line (or in the run configuration of an IDE).

For example (no pun intended!), to run the first example from the `Curve2D` class in Chrome, you would run it like so from the api directory:

```
% flutter run -d chrome lib/animation/curves/curve2_d.0.dart
```

All of these same examples are available on the API docs site. For instance, the example above is available on this page. Most of the samples are available as interactive examples in Dartpad, but some (the ones marked with `{@tool sample}` in the framework source code), just don't make sense on the web, and so are available as standalone examples that can be run here. For instance, setting the system overlay style doesn't make sense on the web (it only changes the notification area background color on Android), so you can run the example for that on an Android device like so:

```
% flutter run -d MyAndroidDevice lib/services/system_chrome/system_chrome.set_system_u_i_ove
```

## Naming

The naming scheme for the files is similar to the hierarchy under packages/flutter/lib/src, except that the files are represented as directories (without the `.dart` suffix), and each sample in the file is a separate file in that directory. So, for the example above, where the examples are from the packages/flutter/lib/src/animation/curves.dart file, the `Curve2D` class, the first sample (hence the index "0") for that symbol resides in the file named lib/animation/curves/curve2_d.0.dart.

Symbol names are converted from "CamelCase" to "snake_case". Dots are left between symbol names, so the first example for symbol `InputDecoration.prefixIconConstraints` would be converted to `input_decoration.prefix_icon_constraints.0.dart`.

If the same example is linked to from multiple symbols, the source will be in the canonical location for one of the symbols, and the link in the API docs block for the other symbols will point to the first symbol's example location.

## Authoring

> For more detailed information about authoring examples, see the snippets package.

When authoring examples, first place a block in the Dartdoc documentation for the symbol you would like to attach it to. Here's what it might look like if you wanted to add a new example to the `Curve2D` class:

```
/// {@tool dartpad}
/// Write a description of the example here. This description will appear in the
/// API web documentation to introduce the example.
///
/// ** See code in examples/api/lib/animation/curves/curve2_d.0.dart **
/// {@end-tool}
```

The "See code in" line needs to be formatted exactly as above, with no wrapping or newlines, one space after the "**" at the beginning, and one space before the "**" at the end, and the words "See code in" at the beginning of the line. This is what the snippets tool and the IDE use when finding the example source code that you are creating.

Use `{@tool dartpad}` for Dartpad examples, and use `{@tool sample}` for examples that shouldn't be run/shown in Dartpad.

Once that comment block is inserted in the source code, create a new file at the appropriate path under `examples/api`. You should also add tests for your sample code under `examples/api/test`.

The entire example should be in a single file, so that Dartpad can load it.

Only packages that can be loaded by Dartpad may be imported. If you use one that hasn't been used in an example before, you may have to add it to the pubspec.yaml in the api directory.

## Snippets

There is another type of example that can also be authored, using `{@tool snippet}`. Snippet examples are just written inline in the source, like so:

````
/// {@tool dartpad}
/// Write a description of the example here. This description will appear in the
/// API web documentation to introduce the example.
///
/// ```dart
/// // Sample code goes here, e.g.:
/// const Widget emptyBox = SizedBox();
/// ```
/// {@end-tool}
````

The source for these snippets isn't stored under the `examples/api` directory, or available in Dartpad in the API docs, since they're not intended to be runnable, they just show some incomplete snippet of example code. It must compile (in the context of the sample analyzer), but doesn't need to do anything. See the snippets documentation for more information about the context that the analyzer uses.

## Writing Tests

Examples are required to have tests. There is already a "smoke test" that runs all the API examples, just to make sure that they start up without crashing. In addition, we also require writing tests of functionality in the examples, and generally just do what we normally do for writing tests. The one thing that makes it more challenging for the examples is that they can't really be written for testability in any obvious way, since that would probably complicate the example and make it harder to explain.

As an example, in regular framework code, you might include a parameter for a `Platform` object that can be overridden by a test to supply a dummy platform, but in the example, this would be unnecessary complexity for the example. In all other ways, these are just normal tests.

Tests go into a directory under test that matches their location under lib. They are named the same as the example they are testing, with `_test.dart` at the end, like other tests. For instance, a `LayoutBuilder` example that resides in `lib/widgets/layout_builder/layout_builder.0.dart` would have its tests in a file named `test/animation/curves/curve2_d.0_test.dart`