

Nested Functions

Many programming languages including Dart allow you to nest functions inside other functions. These are called nested functions. They are useful for giving a name and a scope to a reusable piece of functionality that doesn't necessarily belong to the outer scope of the function (such as its containing class), rather they are scoped to another function.

Further reading:

- [Lexical scope - dart.dev](https://dart.dev/lexical-scope)
- [How to create a nested function in Dart - Educative.io](https://educative.io/curriculum/how-to-create-a-nested-function-in-dart)
- [dart Tutorial => Function scoping - riptutorial.com](https://riptutorial.com/dart/tutorial/10000/function-scoping)
- [Nested function - Wikipedia](https://en.wikipedia.org/wiki/Nested_function)
- [Learning Functions for Dart - Seth Ladd's Blog](https://sethladd.com/blog/learning-functions-for-dart/)



Examples

```
1  const names = ['Foo', 'Bar', 'Baz', 'Foo'];
2  // here we have a function that prints the list of
3  // "names" with their indices
4  void printOrderedNames() {
5      // inside this function we have a helper, nested
6      // function that produces a String describing
7      // the index of each name in the list of names
8      String prefixFor(int index) =>
9          'Value ${index + 1} = ';
10     // then we create an indexed list of names and
11     // iterate through them
12     names.asMap().forEach((index, value) {
13         final prefix = prefixFor(index);
14         print('$prefix = $value');
15     });
16 }
```