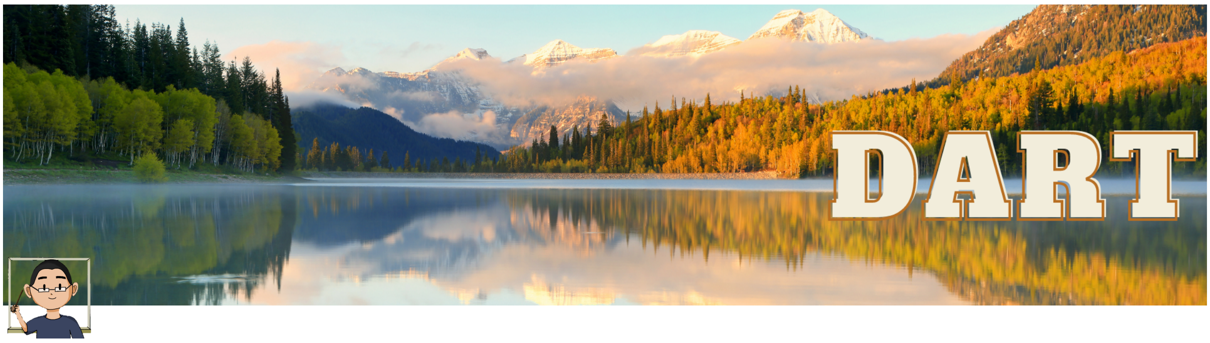


Flutter 基礎 3 - Dart Functions

參考資料 + 翻轉教室輔助影片

Flutter 基礎與實作 3 - Dart Functions

APP 程式設計實務 (2021 Fall)

https://youtu.be/JmZwCXSNbkM

Home

Tutorials about Dart and Flutter.

https://flutterbyexample.com/

Flutter Tutorial

Flutter is an open source framework to create high quality, high performance mobile

https://www.tutorialspoint.com/flutter/index.htm

課程設計

Dart 基礎程式設計 二：Functions

Dart – Function

Dart - Functions - GeeksforGeeks

Function is a set of statements that take inputs, do some specific computation and produces output.

https://www.geeksforgeeks.org/dart-programming-functions/?ref=rp

https://www.geeksforgeeks.org/dart-anonymous-functions/?ref=r

An anonymous function in Dart is like a named function but they do not have names associated with it. An function can have zero or more parameters with optional type annotations. An anonymous function

https://www.geeksforgeeks.org/dart-anonymous-functions/?ref=rp

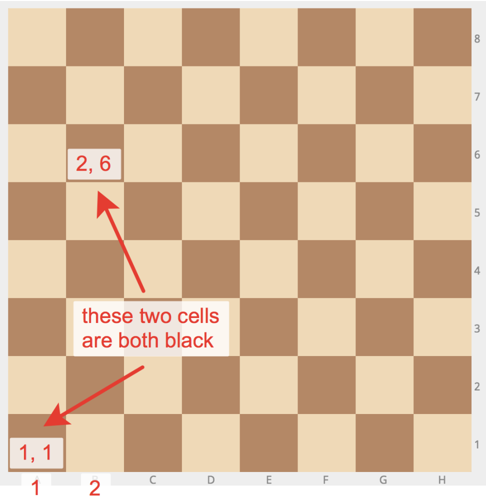
Different Types of Functions in Dart - GeeksforGeeks

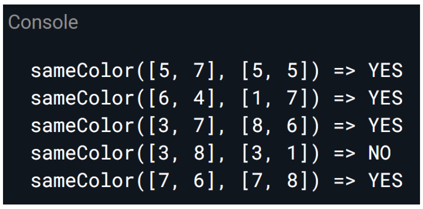
The function is a set of statements that take inputs, do some specific computation, and produces output. Functions are created when certain statements are repeatedly

https://www.geeksforgeeks.org/different-types-of-functions-in-dart/?ref=rp

Function Example : Check if the two positions of the board are the same color?

```
import 'dart:math';

bool color(int x, int y) {
  return x%2==y%2;
}

bool sameColor(List a, List b){
  return color(a[0],a[1])==color(b[0],b[1]);
}

List randCell() {
  Random rnd = Random();
  return [rnd.nextInt(8)+1,rnd.nextInt(8)+1];
}

void main() {
  List c1, c2;

  for(int i=0; i<5; i++) {
    c1 = randCell();
    c2 = randCell();
    String msg = (sameColor(c1,c2))?"YES":"NO";
    print("sameColor($c1, $c2) => $msg");
  }
}
```

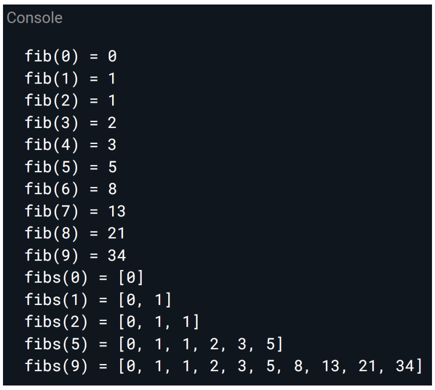



Console

```
sameColor([5, 7], [5, 5]) => YES
sameColor([6, 4], [1, 7]) => YES
sameColor([3, 7], [8, 6]) => YES
sameColor([3, 8], [3, 1]) => NO
sameColor([7, 6], [7, 8]) => YES
```

Recursive Function

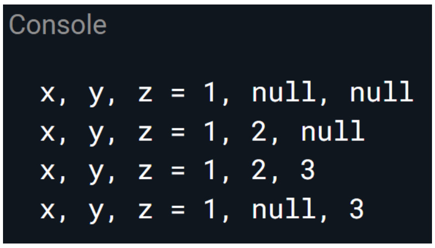
```
int fib(int n) {
  return n < 2 ? n : (fib(n - 1) + fib(n - 2));
}

List fibs(int n) {
  List f = [0];
  if (n<=0) return f;
  f.add(1);
  for(int i=2; i<=n; i++){
    f.add(f[i-2]+f[i-1]);
  }
  return f;
}

void main() {
  for (int i = 0; i < 10; i++) {
    print('fib($i) = ${fib(i)}');
  }
  print("fibs(0) = ${fibs(0)}");
  print("fibs(1) = ${fibs(1)}");
  print("fibs(2) = ${fibs(2)}");
  print("fibs(5) = ${fibs(5)}");
  print("fibs(9) = ${fibs(9)}");
}
```


Console

```
fib(0) = 0
fib(1) = 1
fib(2) = 1
fib(3) = 2
fib(4) = 3
fib(5) = 5
fib(6) = 8
fib(7) = 13
fib(8) = 21
fib(9) = 34
fibs(0) = [0]
fibs(1) = [0, 1]
fibs(2) = [0, 1, 1]
fibs(5) = [0, 1, 1, 2, 3, 5]
fibs(9) = [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

Optional Positional Parameter

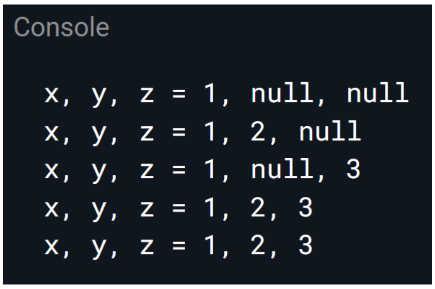
```
void foo(int x, [int? y, int? z]) {
  print("x, y, z = $x, $y, $z");
}

void main() {
  foo(1);
  foo(1,2);
  foo(1,2,3);
  foo(1,null,3);
}
```



```
Console

 x, y, z = 1, null, null
 x, y, z = 1, 2, null
 x, y, z = 1, 2, 3
 x, y, z = 1, null, 3
```

Optional Named Parameter

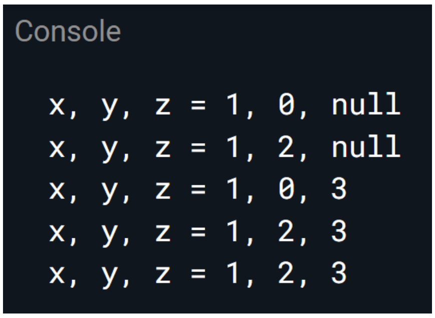
```
void foo(int x, {int? y, int? z}) {
  print("x, y, z = $x, $y, $z");
}

void main() {
  foo(1);
  foo(1,y:2);
  foo(1,z:3);
  foo(1,y:2,z:3);
  foo(1,z:3,y:2);
}
```



```
Console

 x, y, z = 1, null, null
 x, y, z = 1, 2, null
 x, y, z = 1, null, 3
 x, y, z = 1, 2, 3
 x, y, z = 1, 2, 3
```

Optional parameter with default values

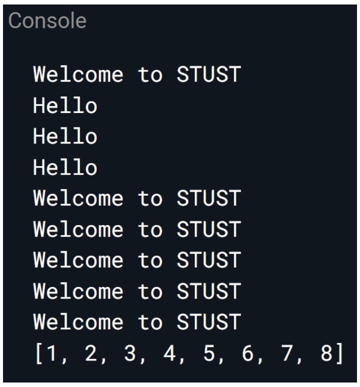
```
void foo(int x, {int? y=0, int? z}) {
  print("x, y, z = $x, $y, $z");
}

void main() {
  foo(1);
  foo(1,y:2);
  foo(1,z:3);
  foo(1,y:2,z:3);
  foo(1,z:3,y:2);
}
```



```
Console

 x, y, z = 1, 0, null
 x, y, z = 1, 2, null
 x, y, z = 1, 0, 3
 x, y, z = 1, 2, 3
 x, y, z = 1, 2, 3
```

Lambda/Arrow Function

```
void hello() => print("Welcome to STUST");

void repeat(var f, int n) {
  for(int i=0; i<n; i++) {
    f();
  }
}

List range(var a, var b) => [for(int i=a; i<b; i++) i];

void main() {
  hello();
  repeat(()=>print("Hello"),3);
  repeat(hello,5);
  print(range(1,9));
}
```

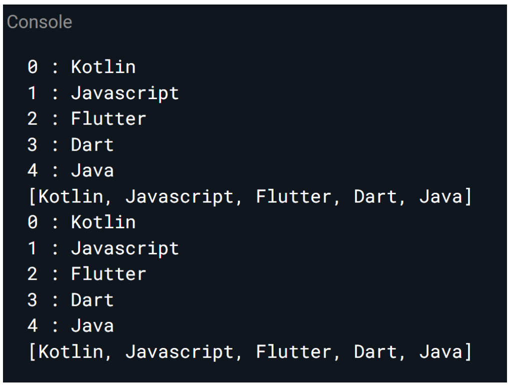


```
Console

 Welcome to STUST
 Hello
 Hello
 Hello
 Welcome to STUST
 Welcome to STUST
 Welcome to STUST
 Welcome to STUST
 Welcome to STUST
 [1, 2, 3, 4, 5, 6, 7, 8]
```

Anonymous Function

```
// Dartprogram to illustrate
// Anonymous functions in Dart
```

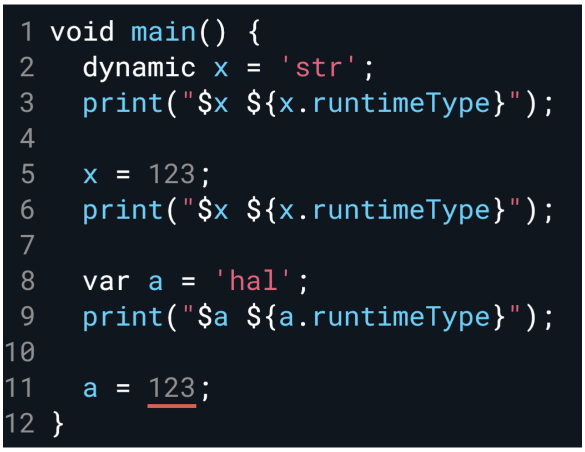
```
void main() {
  var box = ["Kotlin", "Javascript",
             "Flutter", "Dart", "Java"];

  box.forEach((item) {
    print('${box.indexOf(item)} : $item');
  });
  print(box);

  for (int i = 0; i < box.length; i++) {
    print('$i : ${box[i]}');
  }
  print(box);
}
```



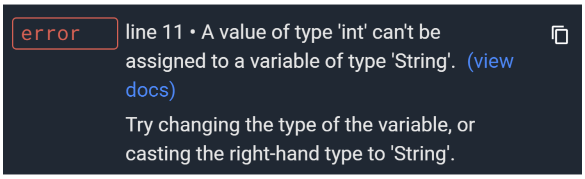
Console

```
0 : Kotlin
1 : Javascript
2 : Flutter
3 : Dart
4 : Java
[Kotlin, Javascript, Flutter, Dart, Java]
0 : Kotlin
1 : Javascript
2 : Flutter
3 : Dart
4 : Java
[Kotlin, Javascript, Flutter, Dart, Java]
```

dynamic vs. var



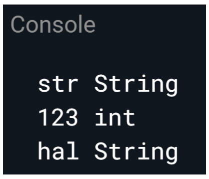
```
 1 void main() {
 2   dynamic x = 'str';
 3   print("$x ${x.runtimeType}");
 4
 5   x = 123;
 6   print("$x ${x.runtimeType}");
 7
 8   var a = 'hal';
 9   print("$a ${a.runtimeType}");
10
11   a = 123;
12 }
```



error line 11 • A value of type 'int' can't be assigned to a variable of type 'String'. (view docs)

Try changing the type of the variable, or casting the right-hand type to 'String'.

```
void main() {
  dynamic x = 'str';
  print("$x ${x.runtimeType}");

  x = 123;
  print("$x ${x.runtimeType}");

  var a = 'hal';
  print("$a ${a.runtimeType}");
}
```

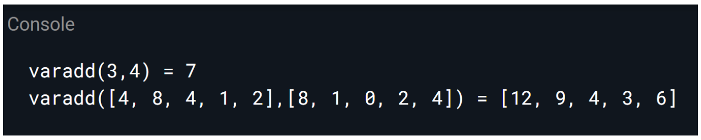


Console

```
str String
123 int
hal String
```

```
import 'dart:math';

dynamic varadd(var x, var y) {
  if (x.runtimeType != int) {
    return [for (var i = 0; i < x.length; i++) x[i] + y[i]];
  } else {
    return x + y;
  }
}

List rndList(int n) {
  var rnd = Random();
  return [for (var i = 0; i < n; i++) rnd.nextInt(10)];
}
```

```
void main() {
  var a = rndList(5);
  var b = rndList(5);

  print("varadd(3,4) = ${varadd(3, 4)}");
  print("varadd($a,$b) = ${varadd(a, b)}");
}
```



```
varadd(3,4) = 7
varadd([4, 8, 4, 1, 2],[8, 1, 0, 2, 4]) = [12, 9, 4, 3, 6]
```

更詳細的操作說明，請參閱『翻轉教室輔助影片』...