Loops in Dart

Dart Loop is used to run a block of code repetitively for a given number of times or until matches the specified condition. The main objective of the loop is to run the code multiple times. Dart supports the following type of loops:

1) Dart for loop

2) Dart for…in loop

3) Dart while loop

4) Dart do-while loop

For Loop

The for loop is used when we know how many times a block of code will execute.

```
void main()
{

   for(var num=1; num<=10; num++)       //for loop to print 1-10
                                           numbers

   {
      print(num);     //to print the number
   }
}
```

For in Loop

The for in loop is slightly different from the for loop. It only takes dart object or expression as an iterator and iterates the element one at a time. The loop will execute until no element left in the iterator.

```
void main()
{
    var list1 = [10,20,30,40,50];

    for(var i in list1)

   {
       print(i);
    }

}
```

While Loop

The while loop executes a block of code until the given expression is false. It is more beneficial when we don't know the number of execution.

```
void main()
{
   int i = 1;
   while (i < 5)
   {
      print( i);
      i++;
   }
}
```

Do While Loop

The do…while loop is similar to the while loop but only difference is that, it executes the loop statement and then check the given condition.

```
void main()
{
   var a = 1;
   do {
    print(a);
     a++;
   }while (a<10);
}
```

Functions

Functions are the building blocks of readable, maintainable, and reusable code. A function is a set of statements to perform a specific task. Functions organize the program into logical blocks of code. Once defined, functions may be called to access code.

```
String sayHelloWorld() {
    return "Hello, World!";
}
void main(){
sayHelloWorld();
}
```

Function with parameters

```
void main(){
  var sum = add(10,20);
  print("Sum Of Given No. Is : ${sum}");
}

int add(int n1, int n2){
    int result;
    result = n1+n2;
    return result;
}
```