SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER APPLICATIONS

23CAT607- CROSS-PLATFORM APP DEVELOPMENT

I YEAR II SEM

UNIT 2 – FLUTTER BASICS

TOPIC 4 – Introduction to Dart Programming

Dart – Data Types

Types of values that can be represented and manipulated in a programming language.

Data Types in Dart

Data Type	Keyword	Description
Number	int, double, num, BigInt	Numbers in Dart are used to represent numeric literals
Strings	String	Strings represent a sequence of characters
Booleans	bool	It represents Boolean values true and false
Lists	List	It is an ordered group of objects
Maps	Map	It represents a set of values as key-value pairs

1. Number

The number in Dart Programming is the data type that is used to hold the numeric value

int: data type is used to represent whole numbers (64-bit Max).
double: data type is used to represent 64-bit precise floating-point numbers.
num: type is an inherited data type of the int and double types.

2. String

Syntax:

```
String str_name;
```

3. Boolean

Syntax:

```
bool var_name;
```

4. List

1. Variable Size List

```
int var_name1 = []; // Alternative for the above declaration
List<int> var_name2;
```

2. Fixed Size List

```
List<int> var_name1 = new List<int>.fixed(size,0);
List<int> var_name2 = new List<int>.generate(size,
function/expres
```

5. Map

The Map object is a key and value pair. Keys and values on a map may be of any type. It is a dynamic collection.

1. Declaring Empty Map

```
// Method 1
Map? map_name;

// Method 2
Map<key_datatype , value_datatype>?
map_name;

// Method 3
var map_name = new Map();
```

Declaring Map with Elements inside it.

```
// Method 1
Map x={
key1 : value1;
key2 : value2;
};

// Method 2
Map<key_datatype , value_datatype> map_name{
key1 : value1;
key2 : value2;
};

// Method 3
var map_name{
key1 : value1;
key2 : value2;
};
```

```
// Dart Program to demonstrate
// Number Data Type

void main() {

  // declare an integer
  int num1 = 2;

  // declare a double value
  double num2 = 1.5;

  // print the values
  print(num1);
  print(num2);
  var a1 = num.parse("1");
  var b1 = num.parse("2.34");

  var c1 = a1+b1;
  print("Product = ${c1}");
}
```

```
2
1.5
Product = 3.34
```

```
// Dart Program to demonstrate
// String Data Type

// Driver Class
void main() {
    // Declaration of String type
   String string = 'Geeks"for"Geeks';
   String str = 'Coding is ';
   String str1 = 'Fun';

   print (string);
   print (str + str1);
}
```

```
GeeksforGeeks
Coding is Fun
```

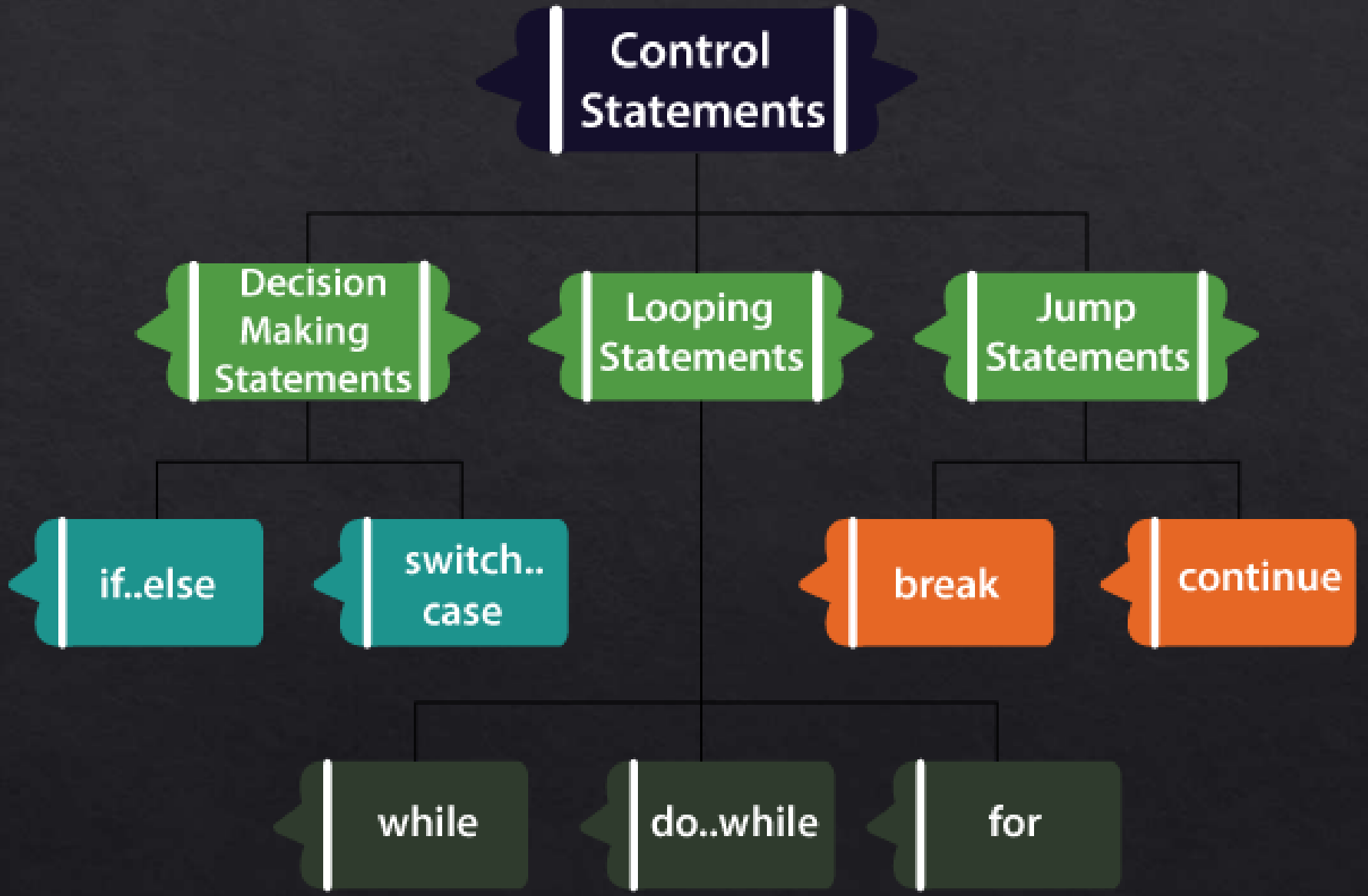
Dart Control Flow Statement

The control statements or flow of control statements are used to control the flow of Dart program. These statements are very important in any programming languages to decide whether other statement will be executed or not.

Categories of Flow Statement

1. Decision-making statements
2. Looping statements
3. Jump statements

Dart Decision-Making Statements



Types of Decision-making statement.

1. If Statement
2. If-else Statements
3. If else if Statement
4. Switch Case Statement

Dart Looping Statements

1. Dart for loop
2. Dart for….in loop
3. Dart while loop
4. Dart do while loop

Dart Jump Statements

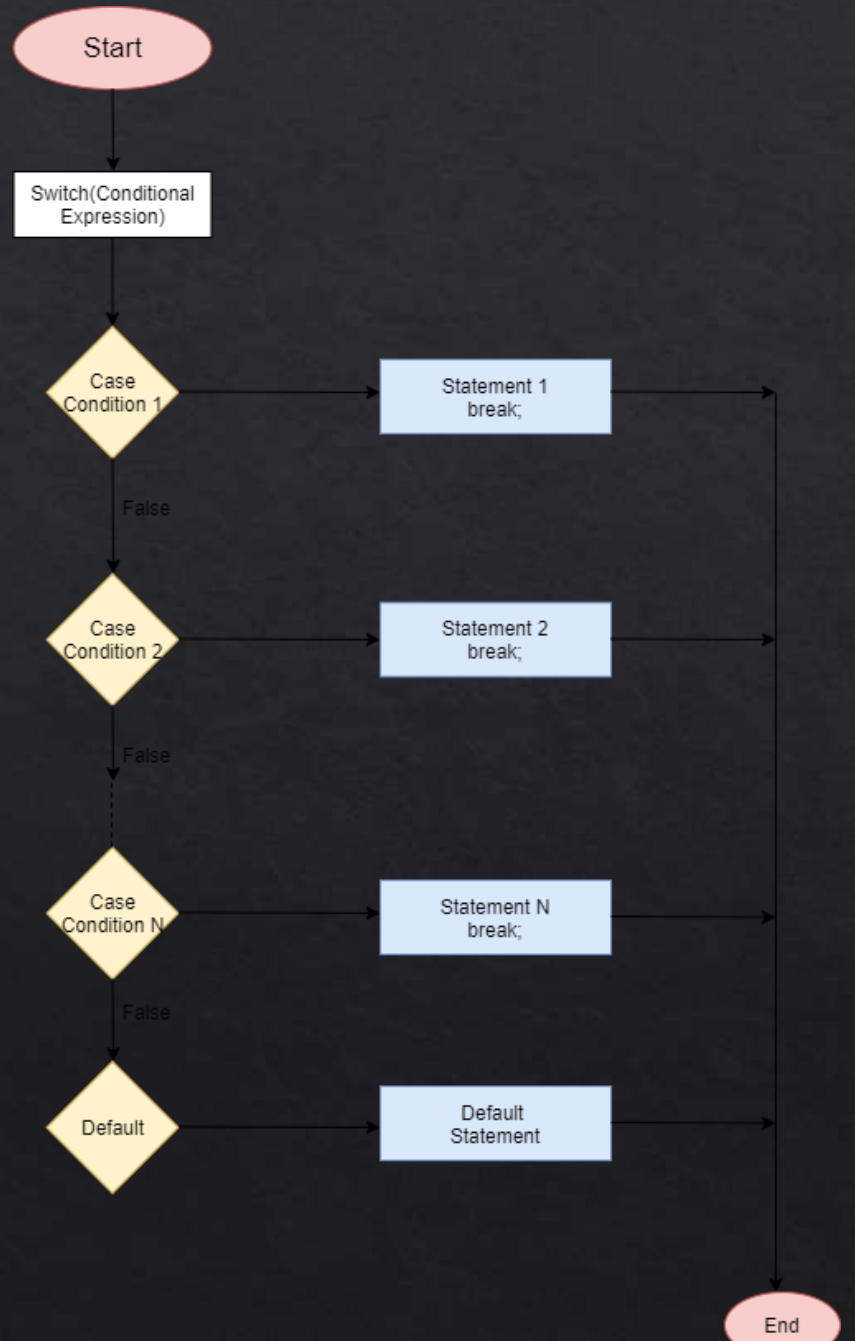
1. Dart Break Statement
2. Dart Continue Statement

Dart if else-if Statement

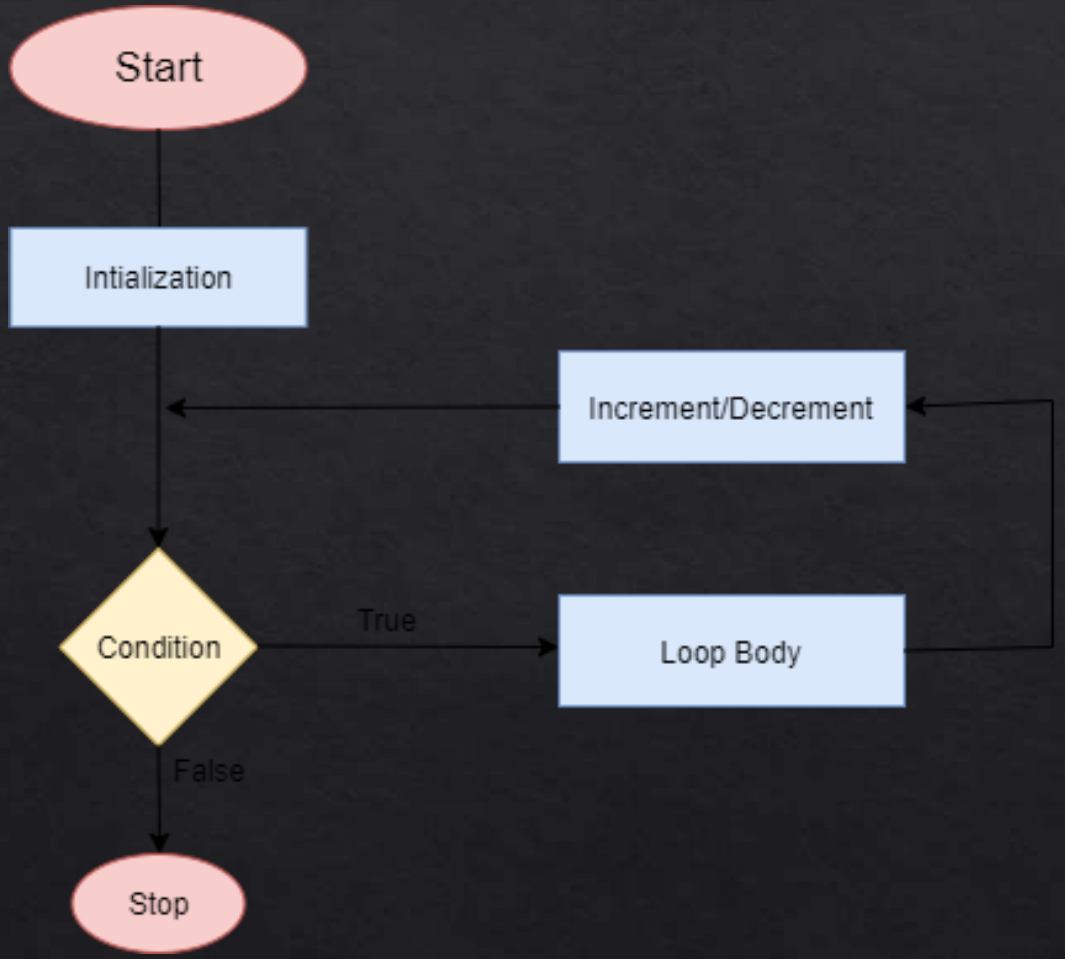
```
void main() {
var marks = 74;
if(marks > 85)
{
      print("Excellent");
}
 else if(marks>75)
{
     print("Very Good");
}
else if(marks>65)
{
     print("Good");
}
else
 {
     print("Average");
}
}
```

Dart Switch Case Statement

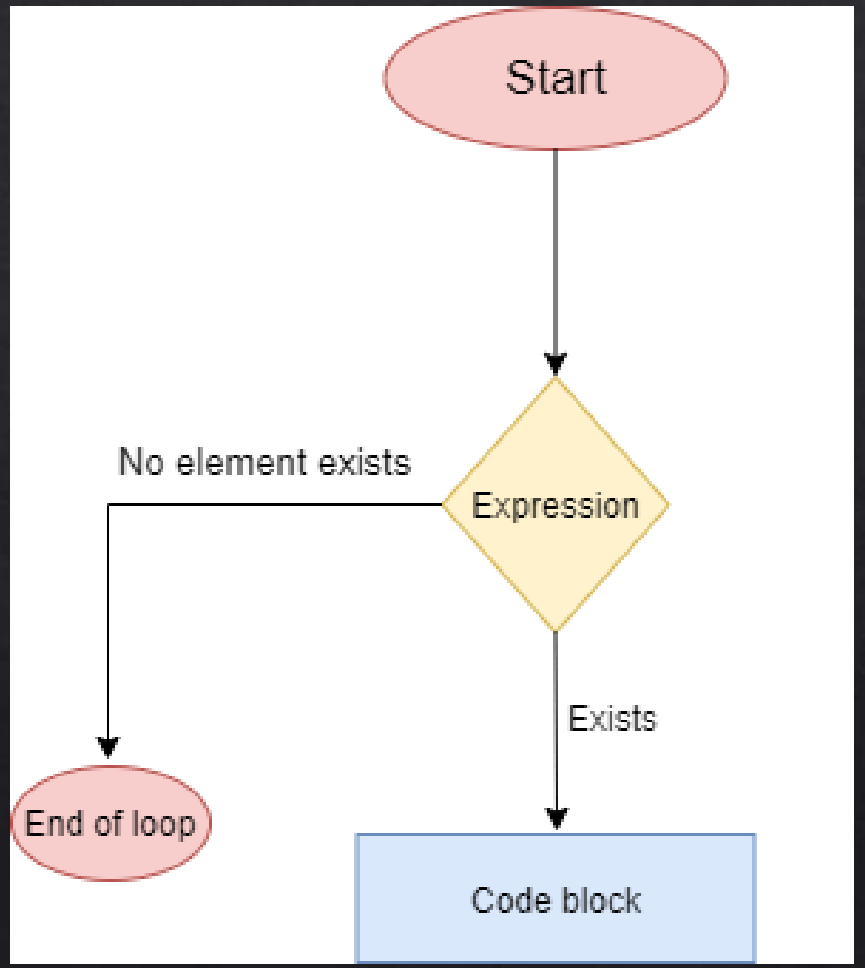
```
void main() {
    int n = 3;
    switch (n) {
      case 1:
        print("Value is 1");
        break;
      case 2:
        print("Value is 2");
        break;
      case 3:
        print("Value is 3");
        break;
      case 4:
        print("Value is 4");
        break;
      default:
        print("Out of range");
        break;
    }
  }
```



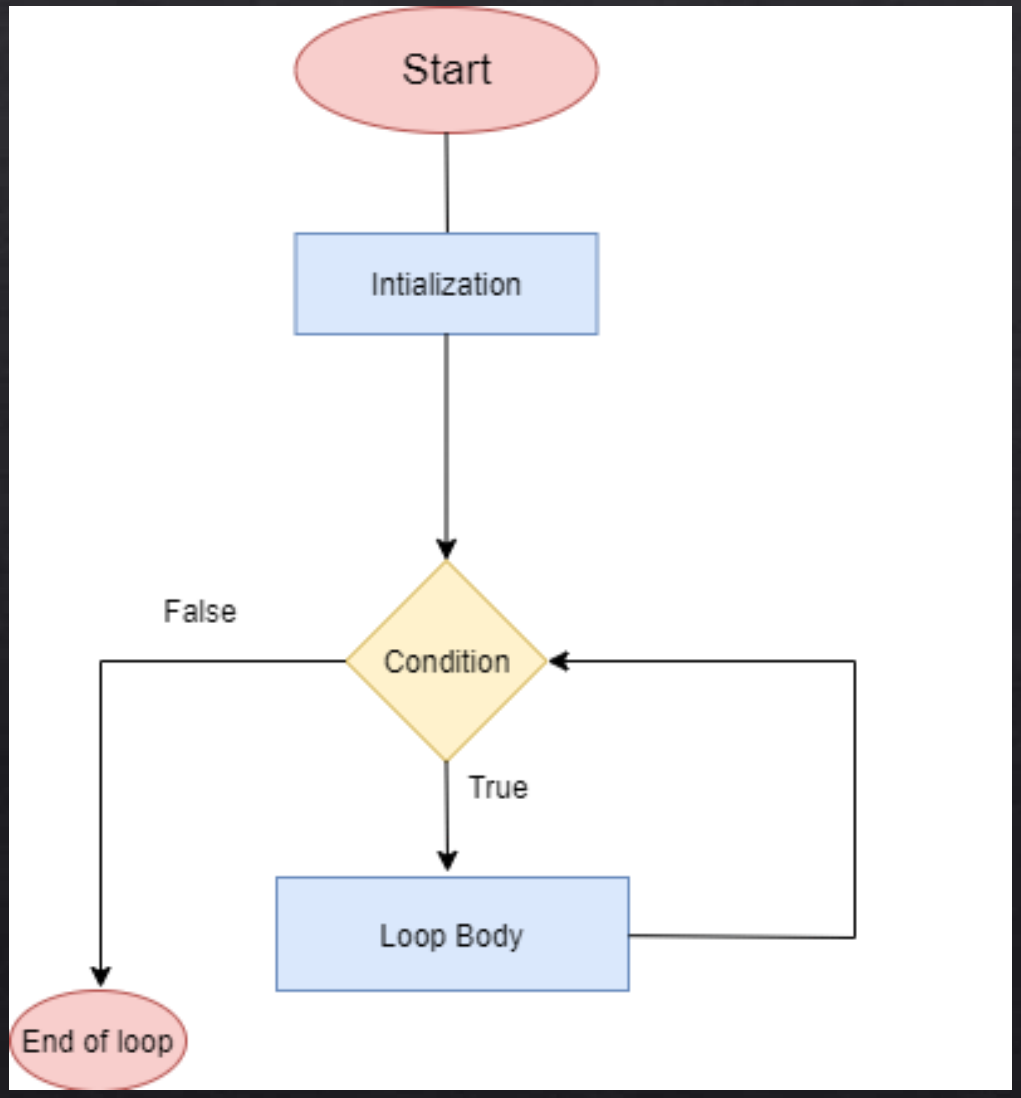
Dart for Loop



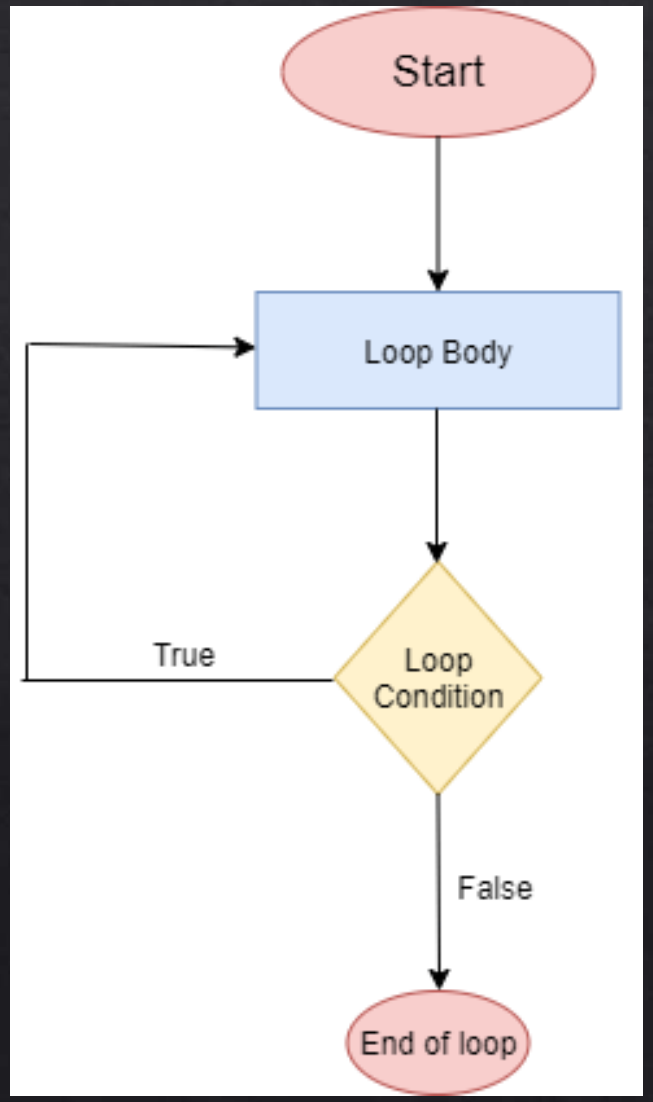
Dart for..in Loop



Dart While Loop



Dart do while Loop



THANK YOU