First Session

- Dart basics
- **Data Types**
- if else
- loobs

by karim toson

Section 1: Data Types _ Variables

its like boxs

Built-in (Primitive) Types

storing direct values single peace of data .

- **int**: For whole numbers (e.g., `42`, `-100`).
- **double**: For decimal numbers (e.g., `3.14`, `2.718`).
- **bool**: For true/false values (e.g., `true`, `false`).
- **String** : for words ("names ").

complex (Non-Primitive) Types

Store complex data structures or references to objects. They can hold multiple values.

- List
- set
- map

Task

defing its data type and display the out put

age = 30

job = ' software engineer '

male = true

score = 95.5

If-Else Statements in Dart

make decisions based on certain conditions.

Basic Syntax and Decision Making

At its core, an `if` statement checks a Boolean condition. If the condition is `true`, the code block immediately following it is executed. Otherwise, that block is skipped.

```
if (condition) {
  // Code to execute if condition is true
}
```

The `else` keyword can be added to specify a block of code to be executed when the `if` condition is `false`.

```
if (condition) {
  // Code to execute if condition is true
} else {
  // Code to execute if condition is false
}
```

Practical Examples in Dart

1. Simple `if` Statement

This example checks if a user's age is greater than or equal to 18 to grant access.

```
void main() {
  int age = 20;

  if (age >= 18) {
    print('Access Granted: You are old enough to enter.');
  }
}
// Expected Output: Access Granted: You are old enough to enter.
```

If-Else If Chain Example

This example demonstrates an `if-else if` chain to assign a performance message based on a numerical grade. The code checks conditions sequentially, executing the first block whose condition evaluates to `true`.

```
void main() {
  int grade = 85;

  if (grade >= 90) {
    print("Excellent");
  } else if (grade >= 75) {
    print("Very Good");
  } else if (grade >= 50) {
    print("Pass");
  } else {
    print("Fail");
  }
}
```

Expected Output:

```
Very Good
```

Task

Write a Dart program that takes a temperature value and prints a message based on the weather condition

If the temperature is greater than 30 =======> print `"It's Hot`

If the temperature is between 15 and 30 (inclusive) ==============> print `"Nice Weather`

If the temperature is less than 15 ===========> print `"It's Cold`

```
void main() {

  int temp=30;

  if(temp>30)
    {
      print('hot');

    }
  else if(temp>15&&temp<=30)
    {
      print('nice');
    }
  else
    {
      print('cold');
    }


}
```

`switch-case` Statements

```
switch (expression) {
  case value1:
    // Code to execute if expression equals value1
    break;
  case value2:
    // Code to execute if expression equals value2
    break;
  case value3:
  case value4: // Multiple cases can share the same block
    // Code to execute if expression equals value3 or value4
    break;
  default:
    // Code to execute if expression doesn't match any case
    break;
}
```

Task

Write a Dart program taks an operator ('-' , '+')

case 1 '+' print => "add operator "

case 2 '-' print => "subtract operator "

Understanding Loops in Dart

1. For Loop

The `for` loop is used when **you know exactly how many times** you want to iterate through a block of code.

Syntax

```
for (initialization; condition; increment/decrement) {
  // code to be executed
}
```

When to Use

Ideal for iterating over a fixed range, such as iterating through elements in a list, or performing a task a specific number of times.

2. While Loop

The `while` loop repeatedly executes a block of code as long as a specified condition is true. The condition is checked before each iteration.

Syntax

```
while (condition) {
  // code to be executed
}
```

When to Use

Use when the number of iterations is not known beforehand, and the loop needs to continue as long as a certain condition remains true (e.g., reading data until the end of a file).

Example

```
void main() {
  int count = 0;
  while (count < 3) {
    print("While loop count: $count");
    count++;
  }
}
// Expected Output:
// While loop count: 0
// While loop count: 1
// While loop count: 2
```

Task

Write a Dart program using a `for loop` to print only the **even numbers** between **1 and 20**

write a Dart program using a `while loop` to calculate the **sum of all numbers from 1 to 50** and print the result.

```
void main()

{

for (int i = 1; i <= 20; i++)

{

 if (i % 2 == 0)

{

print(i); }

}

 }
```

```
void main()
{
int sum = 0;
for (int i = 1; i <= 50; i++)
{
 sum += i;
}
 }
```

Codewars

Reversed Strings

Complete the solution so that it reverses the string passed into it. 'world' =&gt; 'dlrow' 'word' =&gt; 'drow' create a two number calculator where you can (+ , - , / , *) .

Assignment

https://www.codewars.com/kata/5672a98bdbdd995fad00000f