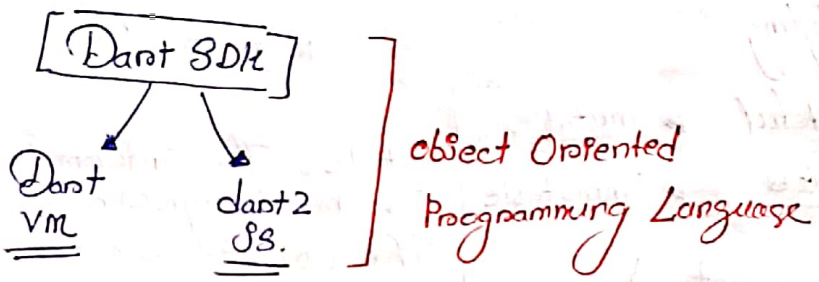
Dart

→ Open Source general purpose Programming Language developed by google.

→ Supports application development in both Client & Server Side.

→ Bears a strong resemblance to Java, C & JS.

→ Launched in 2011, became popular after 2015.



Compilation : dart <file-name> **

Comments : Portions of the code, not executed by the compiler. They are kept as a description of the Program.

→ They are a set of statements which provide provide a proper documentation of the code.

Types of dart comments :

1) Single line comment (//) : Used to comment until a break occurs.

2) Dart Multiline comment (/* ... */) : Used to comment a whole section of the code.

/* → Start
*/ → end

```
// This is a Single line.
/*
* This is
* multiline
*/
```

Documentation Comments : Special comments used to Provide references to Packages.

1) /// → C# Style

2) /** ..... */ → Javadoc Style.

Variables in Dart : It is a name assigned, to a memory location, where the user Stores data. The data can be fetched any time by calling the name of the variable.

↳ There are various types of variables that Store various types of data.

↳ The variable we shall use, depends on the type of data to be Store.

To declare a ~~var~~ variable

type <variable_name>

Types of Variables	Rules
Integer (-1, 1, 6, 2, ...)	i) Variable names can't be identifiers or key words.
Double (1.23, 6.11, 2.3, ...)	ii) Variable names can contain alphabets or number.
String ("a", "name", ...)	iii) Cannot contain Spaces or Special characters, except "$" or "_"
Booleans (true, false)	iv) Variable names cannot begin with numbers
List	
Maps	

* "dart supports type checking"

Keywords

A standard set of reserved words, whose functions are Predefined. They Cannot be used as variable names.

abstract	continue	new	this	as	false
true	Final	null	default	throws	finally
catch	do	for	try	catch	get
dynamic	rethrow	typedef	if	else	var
return	break	enum	void	int	String
double	bool	list	map	implements	set
Switch	case	while	static	import	export
in	external	this	super	with	class
extends	is	const	yield	factory	

Datatypes

→ a variables Stores some data, which can be numbers, textual data or logic (true/False).

→ Each variable is preceeded with a datatype that determines the type of data to be stored.

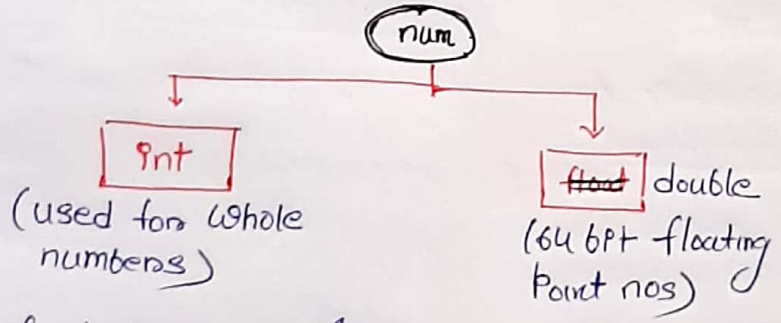
<data-type> <variable-name>

↳ In 'Dart' we have special keywords to determine the data type for each variable.

↳ Since dart is a Statically typed language the datatype of a variable is decided before runtime. and cannot be changed later, during runtime.

Data type	Keyword	Description
Numbers	int, double, num, BigInt	used to represent, numerical literals (1,2,3, 6.403, etc)...
Text	String	"Pokemon", 'abc', "andy" etc.
Booleans	bool	represents boolean values true or false.
List	List	ordered groups of objects. [1,2,3], ["a", "b", "c"]
Maps	Map	represents a set of key value pairs.

Numbers : used to hold numerical data



Refer to Program - lec01

Text : used to store textual data, to represent text, the String keyword is used.

↳ A String is a sequence of characters. (UTF-16).

↳ A String can be represented using
Single quotes (' ')
double quotes (" ")

Boolean: The boolean datatype is used to store a logical True/False value.
We use the bool keyword to declare the bool datatype.

List: A List is an ordered collection of objects. Where within the list each object is accessed by an index. The index starts at '0'

Set: A Set also stores a collection of objects, but it only stores unique objects. No repeated elements can be stored in a Set.

Maps: A map also can store a collection, but it does so, in the formation of key-value pairs.
Here we map one object (key) to another object (value).

---

* **dynamic:** It is a keyword, which when preceded before a variable name signifies that the variable can store any datatype. (The value can also be changed during runtime).

* **Var:** Var is also used to store multiple datatypes, but a moment a value is assigned to it. It takes the datatype of the assigned value. Once it adopts a given datatype, it cannot be changed during runtime.