Mahindra École Centrale
COLLEGE OF ENGINEERING

Object Oriented Programming JAVA

Dr. Prafulla Kalapatapu
Computer Science Engineering
Mahindra Ecole Centrale
prafulla.kalapatapu@mechyd.ac.in

Mahindra École Centrale
COLLEGE OF ENGINEERING

JVM

Execution Engine

Mahindra École Centrale
COLLEGE OF ENGINEERING

What is the responsibility of execution engine?

- Converting the byte code instructions into machine code so that the processor will execute them.
- Execution engine contains two components
    1. Interpreter
    2. JIT Compiler (Just In Time)

JIT Compiler: It is the part of JVM which increases the speed of execution of a Java program

Adaptive Optimizer: Interpreter and JIT Compiler both working at the same time on byte code to convert it into machine code

Example

(i) Print a:

(ii) Print b:

Loop Construction

Repeat following statement 10 times from i=1 to 10.
(iii) Print a:

Looping instruction (An instruction which is in loop)

Java Interpreter Vs JIT Compiler w.r.t Example

Mahindra École Centrale
COLLEGE OF ENGINEERING

Java Interpreter :

(i) Print a:

↓ To convert to machine code and give it to Microprocessor

2 Nano sec

Now Interpreter comes back into memory and reads 2nd instruction

(ii) Print b:

↓ To convert to machine code and give it to Microprocessor

2 Nano sec

Now Interpreter comes back into memory and reads 3rd instruction

Loop construction

(iii) Print a: (for 10 times)

To convert to machine code

2 x 10 = 20 Nano sec

JIT Compiler :

Print a

To convert to machine code

2 Nano Sec

JIT compiler will keep it in

Block Of memory

It takes 2 Nano Sec

So, In total Interpreter
2+2+20=24 Nano sec

After executing first Instruction 2 nano sec for converting machine code and pushes into block of memory in 2 nano sec
Total 2+2=4 nano sec
From next instruction onwards it doesn't convert into machine code. It uses machine code in block of memory

Reason for not giving individual instructions to JIT Compiler is interpreter takes 2 nano sec where as JIT compiler takes 2+2=4 Nano sec

So, In total JIT Compiler + interpreter
2+2+4=8 Nano sec

- After loading the .class code into memory, JVM first of all identifies which code is left to interpreter and which one to JIT compiler. So, that performance is better.

HOT SPOT :

The Blocks of code allocated for JIT Compiler are also called HOT SPOTS.

Mahindra
École Centrale
COLLEGE OF ENGINEERING

LANGUAGE FUNDAMENTALS

Prafulla Kalapatapu

Language Fundamentals

Mahindra École Centrale
COLLEGE OF ENGINEERING

1. Identifiers
2. Reserved words
3. Data Types
4. Literals
5. Arrays
6. Type of Variables

1. Identifiers

Mahindra École Centrale
COLLEGE OF ENGINEERING

- A Name in java program is called identifier. It can be a class name, variable name or method name.

Ex:

```
class Sample
{
public  static void main(String []args)
{
    int x=10;
}
}
```

Rules

Mahindra École Centrale
COLLEGE OF ENGINEERING

- The only allowed characters in java identifiers are

A to Z
a to z
0 to 9
_
$

- Identifiers cant start with digit but it can start with '_'

  Ex: 1abc N $abc Y
  abc1 Y _abc Y

- Java identifiers are case sensitive

  Ex: int a;
  int A;
  } Are different

- There is no length limit for java identifiers,but it is not recommended to take more than 15 length (>15)
- All predefined java class names and interfaces can be used as identifiers

  Ex: int String=10;

- Reserved words cant be used as identifiers

Examples

Mahindra École Centrale
COLLEGE OF ENGINEERING

Which of the following are valid java identifiers :

- Java2Share Y
- 4Shared N
- all@hands N
- _$_ Y
- total# N
- int N
- Integer Y

2. Reserve words

Mahindra École Centrale
COLLEGE OF ENGINEERING

- There are some words that you cannot use as object or variable names in a Java program. These words are known as reserved words, they are keywords that are already used by the syntax of the Java programming language.

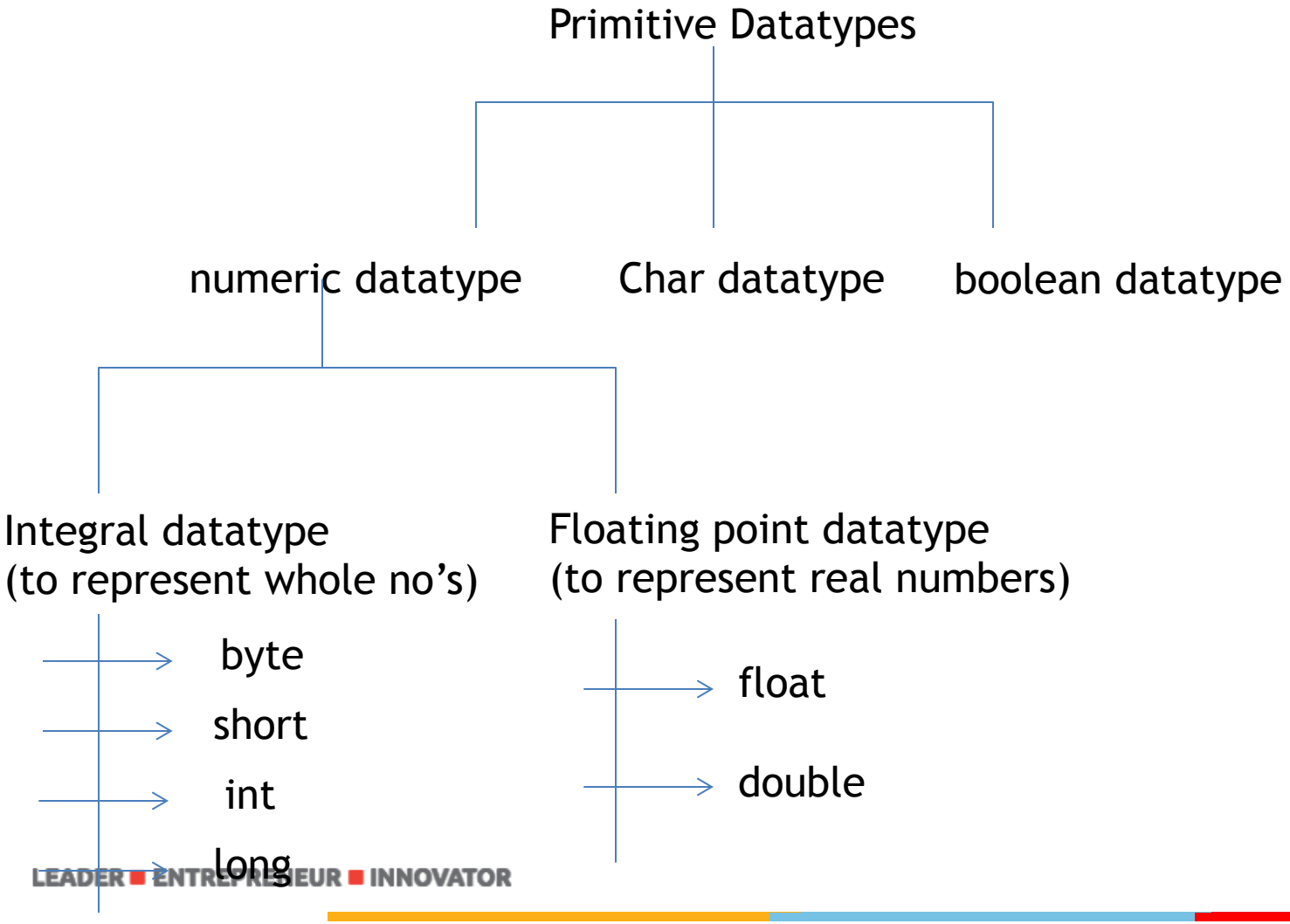
- A keyword is one of **50** reserved words that have a predefined meaning in the language.

- It is specific / particular to the language.

Ex:

abstract	assert	boolean	break	byte	case
catch	char	class	const*	continue	default
double	do	else	enum	extends	false
final	finally	float	for	goto*	if
native	new	null	package	private	protected
public	return	short	static	strictfp	super
true	try	void	volatile	while	

3. Data Types


Primitive Datatypes
numeric datatype
Char datatype
boolean datatype
Integral datatype
(to represent whole no's)
byte
short
int
long
Floating point datatype
(to represent real numbers)
float
double

Integral Datatype [1]

Mahindra École Centrale
COLLEGE OF ENGINEERING

1. byte :

   size = 8 bits or 1 byte

   max value = 127

   min value = -128

Ex: byte b=100; Y

byte b=127; Y

byte b=130; N

CE: possible loss of precision
Found: int
Required: byte

CE: incompatible types
Found: String
Required: byte

byte b="do it"; N

CE: incompatible types
Found: boolean
Required: byte

byte b=true; N

byte b=123.45; N

CE: possible loss of precision
Found: double
Required: byte

Integral Datatype [2]

Mahindra École Centrale
COLLEGE OF ENGINEERING

2. short:

size = 16 bits or 2 byte

max value = 32767

min value = -32768

Ex: short b=32767; Y

short b=-32768; Y

CE: possible loss of precision
Found: int
Required: short

short b=32768; N

CE: possible loss of precision
Found: double
Required: short

short b=123.456; N

CE: incompatible types
Found: boolean
Required: short

short b=true; N

Note: suitable for 16 bit processor like 8086,so this processor is completely outdated, hence corresponding short datatype is outdated

Integral Datatype [3]

Mahindra École Centrale
COLLEGE OF ENGINEERING

3. int:

size = 4 bytes

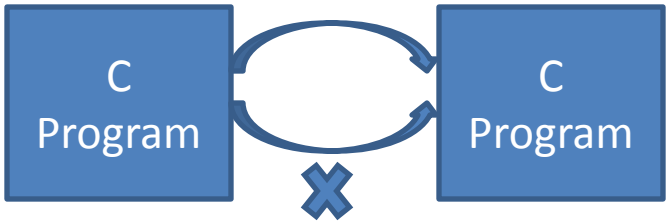
max value = 2147483647

min value = -2147483648

In C-language(int)

16 bit processor (2 bytes)

32 bit processor (4 bytes)

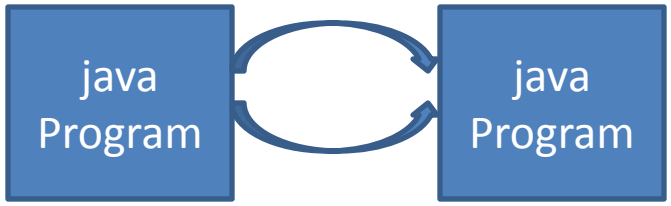


Not Robust

In Java(int)

16 bit processor (4 bytes)

32 bit processor (4 bytes)



Robust

Integral Datatype [4]

Mahindra École Centrale
COLLEGE OF ENGINEERING

4. long:

size = 8 bytes

Range value = $- 2^{n-1}$ to $2^{n-1} - 1$

= $- 2^{63}$ to $2^{63} - 1$

Floating-point Datatypes

Mahindra École Centrale
COLLEGE OF ENGINEERING

float	double
Size = 4 bytes	Size = 8 bytes
Range= $-3.4e^{38}$ to $3.4e^{38}$	Range= $-1.7e^{308}$ to $1.7e^{308}$
If we want 7 decimal places of accuracy	If we want 15 decimal places of accuracy
float follows single precision	double follows double precision

boolean Datatype

Mahindra École Centrale
COLLEGE OF ENGINEERING

- Only allowed values are true or false.
- Which of the following boolean declarations are valid ?

boolean b=true; Y

boolean b=0; N

CE: incompatible type
Found: int
Required: boolean

boolean b=True; N CE: cant find symbol

boolean b="false"; N

CE: incompatible types
Found: String
Required: byte

boolean True=true; boolean b=True; Y

char Datatype

Mahindra École Centrale
COLLEGE OF ENGINEERING

- In C/C++ char size = 1byte
- In java char size = 2 bytes

- ASCII characters takes 0 to 255 [1 byte enough to store].

- Unicode characters takes > 255 [1 byte not enough to store, we need 2 bytes].

- Default values for all data types

Data type	Size(bytes)	Default values
byte	1	0
short	2	0
int	4	0
long	8	0
float	4	0.0
double	8	0.0
char	2	Blank space [\u0000]
boolean	NA	false