Java Programming Language

Boilerplate Code

JavaBasics.java

```
public class JavaBasics{
    public static void main(String args[]){

    } // function.

} //class
```

System.out.println("Hello World");

- println — function; ln → line space; ; → terminator
- // outputline

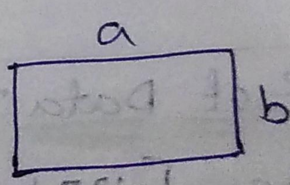
On terminal – to run

javac JavaBasics.java

java JavaBasics.java (classname)

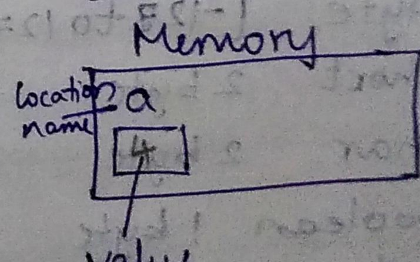
Variables in Java.

area of reactangle



2 * (a + b)

- 2 → literal
- a, b → variables



Data Types in Java

Primitive : byte, short, char, boolean, int, long, float, double

Primitive → already exist

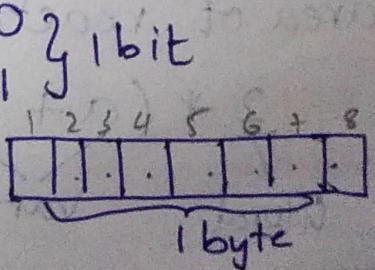
Non-primitive : String, Array, Class
non-exist (we need to create) object, interface

ex:-

```
public class JavaBasics{
    public static void main(String args[]){
        byte b = 8;
        char ch = 'a';
        boolean var = false;
        float price = 10.5;
        int number = 25;
        long n = ___;
        double price = ___;
        short n = 240;
    }
}
```

Size of Data Type

byte	[-128 to 127] 1byte
short	2 byte
char	2 byte
boolean	1 byte
int	4byte
long	8 byte
float	4byte
double	8byte



Comments in Java

```
// Hi there!        single

/*
  Hi there!        multi line
*/
```

Input in Java

next
nextLine
nextInt
nextByte
nextFloat
nextDouble
nextBoolean
nextShort
nextLong

```
import java.util.*;
public class JavaBasics{
    public static void main(String args[]){
        Scanner sc = new Scanner(System.in);
        String input = sc.next();
        System.out.println(input);
    }
}
```

Type Conversion — widening / implicit

One type to another type only
1) Compatible int ↔ float (no to no)
2) Destination type > Source type
   (greater than)

byte → short → int → float → long → double

narrowing/explicit

Type casting: force conversion

float → int

```
import java.util.*;
public class JavaBasics{
    public static void main(String args[]){
        Scanner sc = new Scanner(System.in);
        float number = 99.999f;
        int number2 = (int) number;
        System.out.println(number2);
    }
}
```

Type Promotion in Expressions

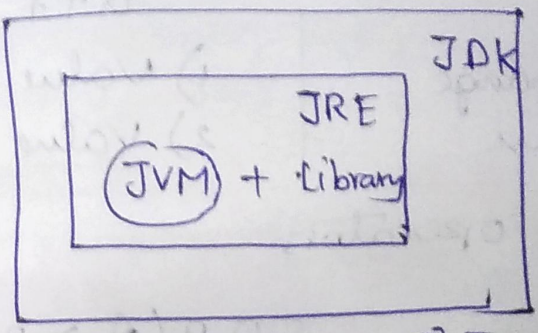
1) Java automatically promotes each byte, short & char operands to int when evaluating an expression.

2) If one operand is long, float or double the whole expression is promoted to long, float & double respectively.

```
byte a = 2
byte ans = (byte)(2 * 2)
```

expression exist. ✓ Converted into integer.

How is our Code Running?



② Execution

Source code .java → Compiler → ByteCode .class → JVM → NativeCode

① Compilation

Operators in Java

Symbols that tells compiler to perform some operation.

expression

Sum = a + b

operands

Types of Operators

Arithmetic Operators (Binary/Unary/Ternary)
Relational Operators
Logical Operators
Bitwise Operators
Assignment Operators

Arithmetic Operators

Binary (2 operands)
+
−
*
/
%(modulo)

Unary (1 operand)
++ (increment)
-- (Decrement)