

# PROGRAMMING IN JAVA

~~3/19~~

Java → Programming language

Platform  
(environment in which a program runs)

API → Application Programming Interface

HTML → Java (Servlet)

Java → HTML (JSP)

## Features of Java

1. Simple
2. Object oriented
3. Portable
4. Platform independent
5. Secure
6. Robust
7. Architectural neutral.
8. Dynamic interpreted
9. High performance
10. Multithreaded
11. distributed

Object Code  $\Rightarrow$  It is a form that only OS can interpret.

Machine Code  $\Rightarrow$  Set of instruction in man machine language in binary that can be executed by directly CPU.

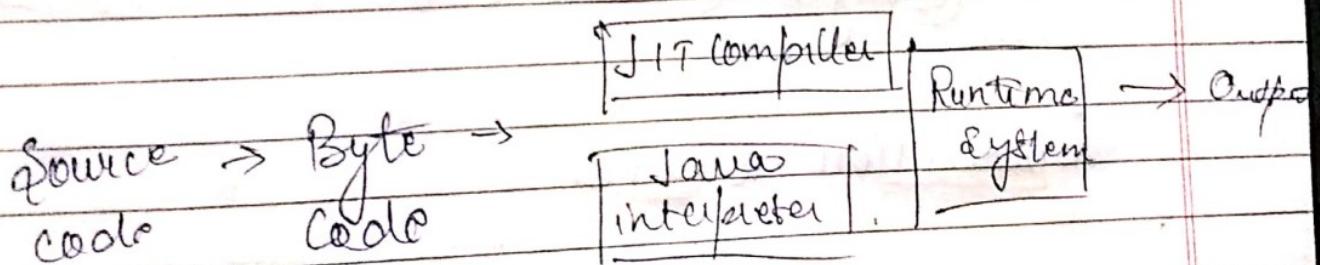
Byte Code  $\Rightarrow$  Code generated from compiling source code which can be executed by Java Virtual Machine

## ~~18/19~~ Architecture of JVM

Class loader

Memory Area

Execution Engine

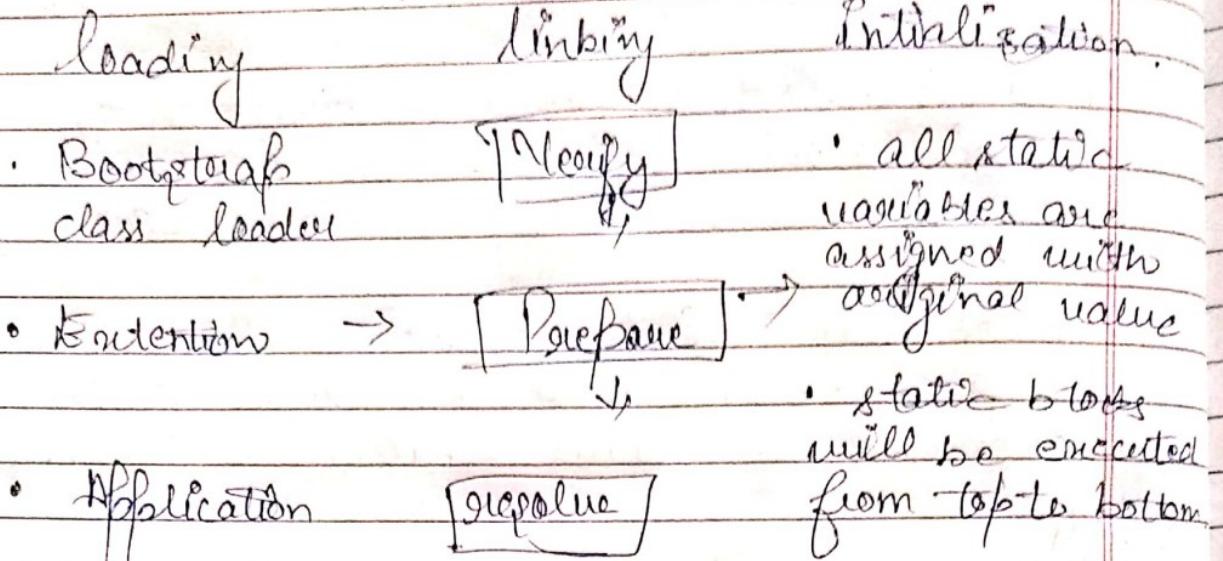


JVM  $\Rightarrow$  application based virtual machine.

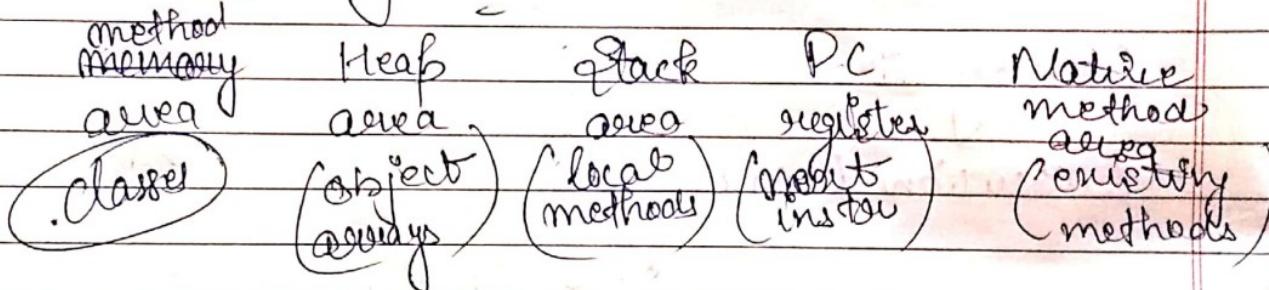
Task of JVM's

- load . class file
- execute . class file

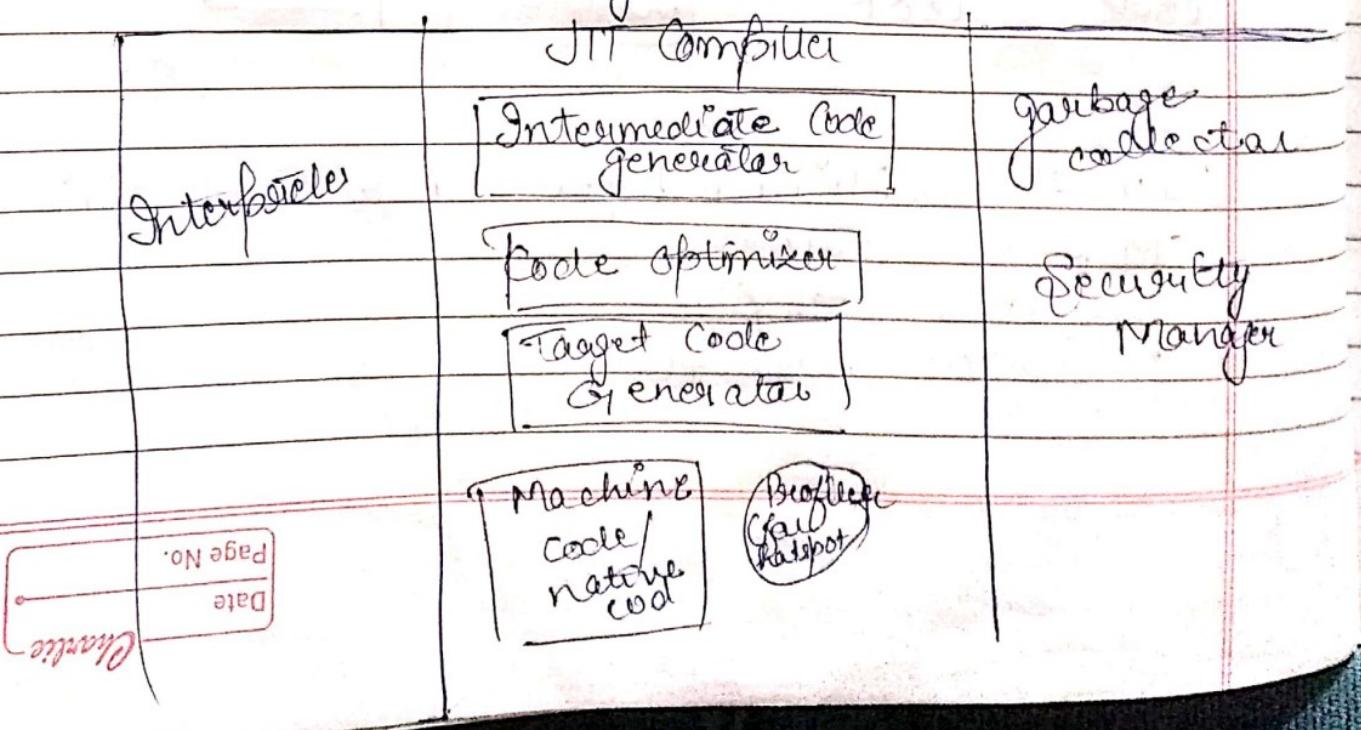
## Class loader



## Memory Areas



## Execution Engine



# Java Native Interface (JNI)

↳ Native libraries

## Java Variables

types of variable

- local → inside the method
- instance → inside the class outside the method
- static → which cannot be changed

class A {

    int a = 50; (instance).

    static int m = 100; (static variable)

    void method()

    {  
        int n = 100; (local variable)  
    }

}

~~6/8/19~~

## Data types in Java

- Primitive data types (int char float)
- Non-Primitive data types (class, interface, array)

Char uses 2 bytes in java.

! 0000 → lowest range of unicode

java uses unicode not ASCII  
highest range → ' UFFF '

## Type conversions

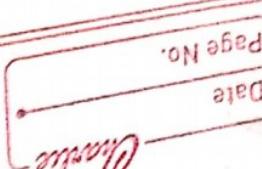
- Widening or Automatic Type conversion.  
It is possible if:
  - two data types are compatible with each other.
  - when we assign a smaller data type to larger data type.

Byte → short → int → long → float → double.

- Narrowing or Explicit Conversion is

When we assign large value to smaller data type

double → float → long → int → short → byte.



# Type Promotion in Java Expressions

## Condition for type promotions

- Java ~~will~~ automatically promote each byte, short, or char operand to int when evaluating an expression.
- If one operand is a long, float or double the whole expression is converted to long float or double.

(1)

```
int g = 3;  
System.out.println(++g * 8);
```

(2) char a = 'A';

a++;

```
System.out.println((int)a);
```

(3) boolean var1 = true;

boolean var2 = false;

```
System.out.println(var1 & var2);
```

(4) Class Test

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

byte b = 49;

char c = 'a';

short s = 1024;

int i = 50000;

float f = 5.67f;

double d = 123.4;

double result = (f \* b) + (i / c) - (d \* s)

```
System.out.println("result = " + result);
```

~~8/19~~

Operations in Java is

+,-,\*,/

$10 \ll 2$  (left shift)

$$10 \times 2^2 = 10 \times 4 \\ = 40$$

$10 \ll 3$

$$10 \times 2^3 = 10 \times 8 \\ = 80$$

$10 \gg 2$  (right shift)

$$\begin{array}{r} 10 \\ \times 2^2 \\ \hline 10 \\ - \end{array} = \frac{10}{4} = 2$$

$10 \gg 3$

$$\begin{array}{r} 10 \\ \times 2^3 \\ \hline 10 \\ - \end{array} = \frac{10}{8} = 1$$

$10 \gg 2$

$20 \gg 2$  (Parity changes)

$$\begin{array}{r} 20 \\ \times 4 \\ \hline 20 \\ - \end{array} = 5$$

0101

①101

|           |      |          |
|-----------|------|----------|
| Classmate | Date | Page No. |
|-----------|------|----------|

## Control Statement

Java Selection statements

if

if else

nested if  
if-else-if

switch case

jump break, continue, return

label:

start; stat2;

do b

~~Q~~ User input :-

- WAP to display upper case characters from A to Z.
- WAP to count vowels, digits, spaces and consonants.
- WAP to print:  

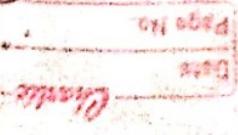
|       |
|-------|
| 1     |
| 2 3   |
| 4 5 6 |

4. WAP to find ASCII value of character

5. WAP to generate random numbers

6. WAP to enhance for loop.

7. WAP to add digits in a no.



① class atax

{ public static void main (String args [ ] )

int i;  
for (i = 'A'; i <= 'Z'; i++)

System.out.println ((char) i);  
}

for (data type item : collection)

Random

import java.util.\*;

for (counter = 1; counter <= 5; counter++)  
System.out.println (num.nextInt ());

8/8/19

## Arrays

Arrays in Java are implemented as objects.

We have to declare an array and then instantiate it using new operator.

All arrays are allocated dynamically.

datatype arrname[]  
arr

datatype[] arrname

length is for the length of array.

## Types of Arrays

Single Dimensional

Multi-dimensional

arrayrefvar = new datatype [size],

## Instantiating an array is

When an array is declared, only a reference of array is created. To actually give memory to array it is created via `arrname = new datatype[size];`

|       |      |          |
|-------|------|----------|
| Class | Date | Page No. |
|-------|------|----------|

`int [] intArray = new int[5];`

ArrayList

`int [] intArray = new int[]{1, 2, 3, 4, 5, 6, 7, 8};`

ArrayList index out of bound exception.

Multidimensional array.

datatype [ ] arrayname [ ]

`int [][] arr = new int[3][3]`  
(3 rows & 3 columns)

What is the class name of this array?

For array object, a proxy class is created whose name can be obtained by getClass() method on the object.

Class C = arr.getClass();

String name = arr.getName();

System.out.println(name);

14/8/19 Java String

In java, string is an object that represents a sequence of characters.

Class → `java.lang.String`

- methods to perform operation on strings
  - `compare()`
  - `concat()` . . .

Two ways to create a String object

1. By string literal.
2. By new keyword.

3. By passing character

String s1 = "welcome";

String s = new String("Welcome");

• Immutable String in java.

• Java of String Comparison

↳ authentication.

↳ sorting

is reference matching

⇒ Methods to compare string

↳ By `equals()` method

↳ By `==` operator

↳ By `compareTo` method.

1. `Object.equals()` method  
It compares values

Public boolean equals(Object another)

2. Compare by `=` operator  
It compare references.

3. Compare by `compareTo()` method

### ME-8

1. Write a Java program to check if array contains specific value

2. WAP to check whether two arrays are equal

3. WAP to print class name of an array.

4. WAP to sum value of an array

(`java.util.Scanner`)

Scanner `R = new Scanner(System.in)`

`int a = R.nextInt()`

~~20/8/19~~

## String Constructors

- ① String (byte [] byte arr)

byte [] b-arr = {71, 101, 101, 107, 115};

String s-byte = new String (b-arr);

convert an array of byte data type  
into array.

- ② String (byte [] byte arr, Charset charset)

Charset → It defines mapping b/w seq.  
of characters and seq. of bytes.

byte [] b-arr = {71, 101, 101, 107, 115};

charset cs = Charset.defaultCharset();

String s-byte-char = new String (b-arr, cs);

- ③ String (byte [] byte arr, String charset name)

byte [] b-arr = {71, 101, 101, 107, 115};

String s = new String (b-arr, "US-ASCII");

④ `String(byte[] byteArr, int start_index,  
int length)`

`byte[] bArr = {71, 101, 101, 07, 115};`

`String s = new String(bArr, 1, 3);`

⑤ `String (byte[] byteArr, int start_index,  
int length, CharSet charSet)`

⑥ `String (byte[] byteArr, int start_index,  
int length, String charSet name)`

⑦ `String (char[] charArr)`

⑧ `String (char[] charArray, int start_index,  
int count)`

⑨ `String (int[] uniCodePoints, int offset,  
int count)`

⑩ `String (StringBuffer sBuffer)`

1. `String (StringBuilder sBuilder)`

# String Methods



1. int length()
2. char charAt(int i)
3. String substring(int i)
4. String Substring (int i, int j)
5. String concat (String str)
6. int indexof (String s, int i)
7. int indexOf (String s, int i)
8. int lastindexof (" ", " "); (String s);
9. boolean equals (Object otherObj)
10. boolean equalsIgnoreCase (String anotherString)
11. int compareTo (String anotherString)
12. int compareToIgnoreCase (String anotherString)
13. String toLowerCase(),
14. String toUpperCase()
15. String trim()
16. String replace (char oldChar, char newChar)

~~9/18/19~~

## Class & Objects

Object  $\Rightarrow$  entity that has state and behaviour.

Characteristics:

- state • behaviour • Identity

Class  $\Rightarrow$  group of objects which have common properties

|       |      |          |
|-------|------|----------|
| Class | Date | Page No. |
|-------|------|----------|

Class contains :-

Fields

Methods

Constructors

Blocks

Nested class and interface.

## Instance variable in Java

Variable which is created inside the class but outside the method.

## Method in Java

Method is like a function used to expose the behaviour of an object.

## Advantage of Method

- Code Reusability
- Code Optimization

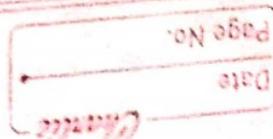
New keyword :-

Memory allocation at runtime.

All objects get memory in Heap memory area.

main outside the class

If you define multiple classes in file, it is good to save the file name with class name along with main func.



### 3 Ways to Initialize object

- By reference
- By method,
- By constructor.

### Anonymous Object

Object with no reference

`new Calculation();` // anonymous object  
// calling method.

`new Calculation().fact(5);`

Lab

WAP to implement string const.

Implementation of string buffer  
handle by methods

in WAP to compare the performance

of two string

WAP to check if it is empty or

not

WAP to split two string using delimiter

B  
29/8/19

Constructors: It is a block of codes similar to the methods and is called when ~~its~~ <sup>the</sup> first - the instance of class is created.

Types of constructor is

- Default
- Parameterized

Constructor Overloading

Constructor overloading is a technique of having more than one constructor with diff. parameters.

| Constructor   | Method  |
|---|---|
| <ul style="list-style-type: none"><li>• Initialize the state of object.</li><li>• A constructor must not have a <u>return type</u>.</li><li>• Invoked implicitly.</li></ul> | <ul style="list-style-type: none"><li>• expose the behaviour of an object</li><li>• A method have a return type.</li><li>• invoked explicitly</li></ul> |

Java.lang.reflect

'This' is a reference variable that refers to the current object

this : To refer the current class instance  
variable.

3) to invoke current class methods.

4) to invoke current class constructor.

~~garbage~~

## Garbage collection

garbage collector removes the unreferenced  
~~object~~ from the heap memory

How to unrefernce object

- By nullifying reference  
Employee e = new Employee();  
e = null;

- By assigning a reference to another.

class

- By anonymous object etc.

new Employee()

finalize() method

finalize() method is invoked each time  
before the object is garbage collected.

protected void finalize()



## gc() method

It is used to invoke the garbage collector to perform cleanup processing.

protected

It can be overridden by subclasses who need to define behaviour of it.

why  
not  
class  
use  
finalise

no proper specification is there.

## Method Overloading

Method → same name,  
diff parameters

28/9/19 Inheritance

Inheritance represents is-a relation  
parent → child

Why use Inheritance

1. Method overloading (so runtime polymorphism can be achieved)
2. Code Reusability

## Syntax

Class Subclass name extends Super class name

// methods and fields

extend keyword → you are making a new class from extended class.

Types of Inheritance.

Multiple Inheritance is not supported in java.

→ chances of error

Method Overriding: If subclass has the same method as declared in the parent class, it is known as method overriding in java.

Some name + some parameters

Super Keyword

It is a reference variable which is used to refer immediate parent class object.

Usage:

1. Immediate parent class instance variable
2. to invoke immediate parent class method
3. immediate parent class constructor.

|       |      |          |
|-------|------|----------|
| Class | Date | Page No. |
|-------|------|----------|

## Final Keyword

It is used to the user  
It can be  
variable  
method  
Class

## Java final keyword

- Stop value change
- Stop Method Overriding
- stop inheritance

## MES

- Copy constructor/Passing obj as parameter
- Method Overloading
- User of this keyword

29/8/19

Dynamic method Dispatch

(Runtime polymorphism)

Polymerphism  $\Rightarrow$  Single action in diff  
ways

polymorphism

runtime

compile time

|         |          |
|---------|----------|
| Date    | Page No. |
| 20/8/19 | 1        |

Upcasting is If the reference variable of parent class refers to the object of child class

## Advantage of Dynamic Method Dispatch

- Support overloading of methods

## Abstract Class

A class declared with the abstract keyword

It cannot be instantiated.

## Rules

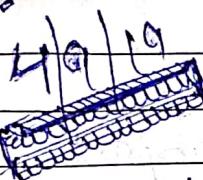
- Abstract must be declare with an abstract keyword
- It can have abstract and non abstract methods,
- It can have constructors and static methods also
- It can have

~~Q919~~

## Dynamic method Dispatch

Abstract class

Fibonacci series



# Exceptions in Java

to handle the runtime errors.

Advantage

- to maintain the normal flow of application

## Exception

↳ checked

↳ To exception

↳ To exception

↳ SQL exception

Compile time

↳ unchecked

↳ error.

Runtime

array index

out of bound

Arithmatic

null pointer  
exception

↳ error

→ Out of memory

→ Assertion failure

## Java exception keyword

try

Catch

finally

Throw

throws

## Syntax of Java try - catch

(24)

try {

// code that may throw exception

}

catch (Exception class Name ref) { }

## Syntax of try - finally block

try {

finally { }

~~finally~~

1. Implementation of interface

2. Interface Inheritance

3. Multiple inheritance

4. Default method.

~~10/9/19~~

## Java Nested Try Block

try {

Statement 1;

Statement 2;

try {

Statement 1;

Statement 2;

}

catch (Exception e)

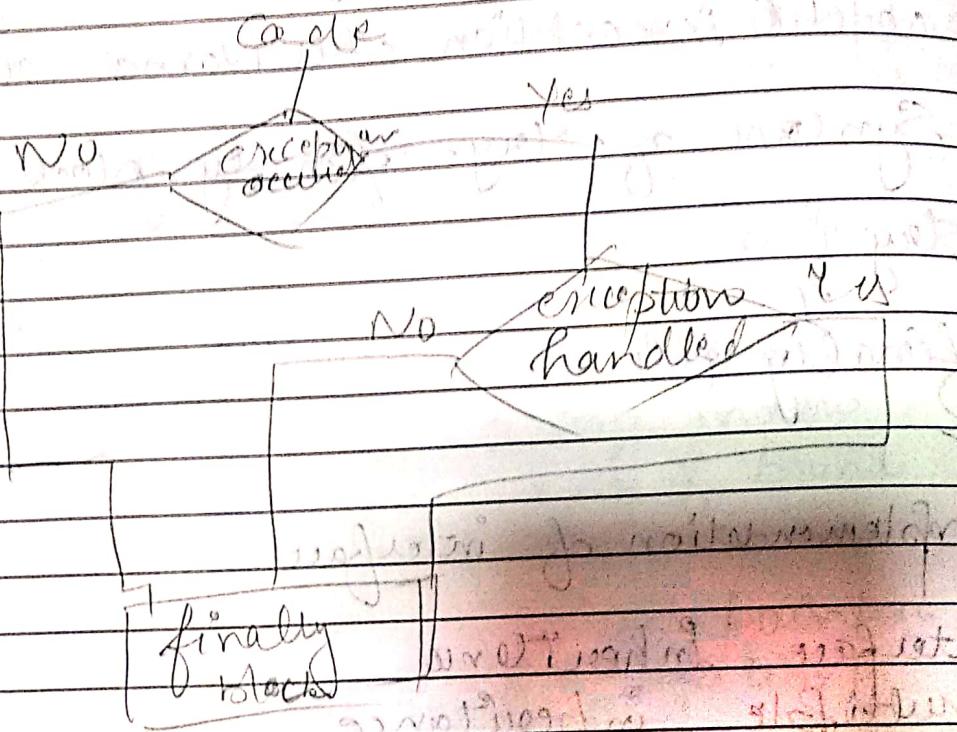
{}

|             |          |
|-------------|----------|
| Challan No. | Page No. |
| _____       | _____    |

Challan  
Catch (Exception e)  
{ }

## Java Finally block

It is always executed whether there is an exception or not.



finally block is not executed if program exits.

Why use finally?

- Used to put cleanup code.

Java throws exception

It is good to explicitly throw the exception.

~~11/9/19~~ Eg :- throw new ArithmeticException.

~~11/9/19~~

Java throws Keyword

It is used to declare an exception.

~~12/9/19~~

Java throws Keyword.

return type method\_name() throws  
exception class\_name

{ //method code  
}

Difference between throw and throws

Java Custom exception.