Class Note:

Subject : Oop...Using...Java.

Faculty : .....

Topic: .....Java Basics: Review of Object Oriented Concepts.

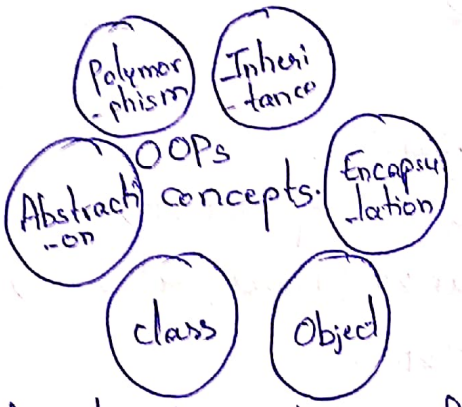
Unit No. : 1
Lecture No.: 1
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 01

Author: James Gosling.
Vendor: Sun Microsystems
Type : Open source software
IDE : Netbeans, Eclipse, etc.,
Initial name : OAK (coffee tree)
Present name : Java
Extension : .java, .class, .jar
Initial version : jdk 1.0 (Java Development kit)
Present version : Java 22 & jdk 22 (March, 19th 2024)
Languages of java : J2SE, J2EE, J2ME.
OS : Any OS (windows, Ubuntu etc)
Symbol : coffee cup with saucer.
slogan/motto : WORA (write once run anywhere)
objective : To develop web application.
Compilation : javac filename.java
execution : java classname.
o/p : ——
pre requisit languages : C, C++.
Textbooks: 1. Herbert Schildt and Dale Skine, "Java fundamentals, A Comprehensive Introduction. McGrawHill, 1st Edition 2013
2. Herbert Schildt - Java the Complete Reference, McGrawHill, 7th Edition 2011.

Review of Object Oriented concepts:-

Main concepts of OOPs are

i, Class
ii, Object
iii, Method and method passing.
iv, Pillars of OOPs.
- → Abstraction
- → Encapsulation
- → Inheritance
- → Polymorphism.
  - → Static polymorphism (compile time)
  - → Dynamic polymorphism (runtime).



- A Class is user defined blueprint or Prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. Using classes, you can create multiple objects with the same behavior instead of writing their code multiple times. This includes classes for objects occuring more than once in your code.
- In general, class declarations can include these components in order:

i, Modified: A class can be public or have default access.
ii, Class name: The class name should begin with the initial letter capitalized by convention Ex:- My Class.java.

Class Note:

Subject : ...Oops..Using.Java.

Faculty : .....

Topic: .....History.....of.....Java.

Unit No. : 1
Lecture No.: 02
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 2

iii, Super class: The name of the class's parent (super class), if any preceed by the keyword extends A class can only Extend (sub class) one parent.

iv, Interface: A comma separated list of interfaces implemented by the class, if any. Preceed by the keyword implements A class can implement more than one interface.

v, Body: The class body is surrounded by braces { }.

Objectr - It is a basic unit of OOP that represents real life entities.

- A typical java program creates many objects, which as you know, interact by invoking methods.
- The objects are what perform your codes, they are the part of your code visible to the viewer/user.

An object mainly consists of:

i, State: It is represented by the attributes of an object. It also reflect the properties of an objects.

ii, Behaviour: It is represented by the methods of an object. It also reflects the response of an object to other objects.

iii, Identity: It is a unique name given to an object that enables it to interact with other objects.

iv, Method: A method is a collection of statements that performs

some specific task and return the result to the caller.
- A method can perform some specific task without returning anything.
- Methods allow us to reuse the code without retyping, which is why they are considered time savers. In Java, every method must be part of some class, which is different from languages like C, C++

History of Javai

- Java was originally designed for interactive television, but it was too advanced technology for the digital cable television industry.
- The history of Java starts with the Green Team.
- Java team members (are also known as Green Team), initiated this project to develop a language for digital devices such as set of TV's etc.
- Which best suited for Internet Programming later it was incorporated by Netscape Navigator.
- The principles for creating Java Programming were "Simple, Secure, Robust, Portable, Platform-independent, High Performance, Multi-threaded, Architectural Neutral, Object oriented, Interpreted, and Dynamic.
- Java was developed by James Gosling, who is known as the father of Java, in 1995.
- Currently, Java is used in internet programming, mobile device, games, e-business solutions etc.,
- James Gosling, Mike Sheridan and Patrick Naughton initiated the Java language project in June 1991. The small team of Sun Engineers called Green Team.
- OAK is symbol of strength, and chosen as a national tree of many countries like USA, France, Germany and Romania etc.

Class Note:

Subject : Oops using Java.

Faculty : .....

Topic: Features of Java | Java Buzzwords

Unit No. : 1
Lecture No.: 3.
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 3

- Finally the java is just a name, not an acronym.
- JDK 1.0 was released on Jan 23 1996.

features of Java & Java Buzz words:

- The primary objective of java language creation was to make it

i, Simple.
ii, Object oriented.
iii, Portable
iv, Platform independent
v, Secured
vi, Robust.
vii, Architectural neutral
viii, Interpreted
ix, High performance
x, Multi threaded
xi, Distributed
xii, Dynamic.

i, Simple: Java is very simple programming language becaz
- Java has eliminated all the difficult and confusion oriented concepts like pointers, multiple inheritances.
- The C, CPP syntaxes easy to understand and easy to write. Java maintains C and CPP syntax mainly hence java is simple language.
- Java tech takes less time to compile and execute the pgm.

ii, Object oriented :- Java is object oriented because to represent real data in the form of object. By using object reference we are calling all the methods, variables which is present in the class.

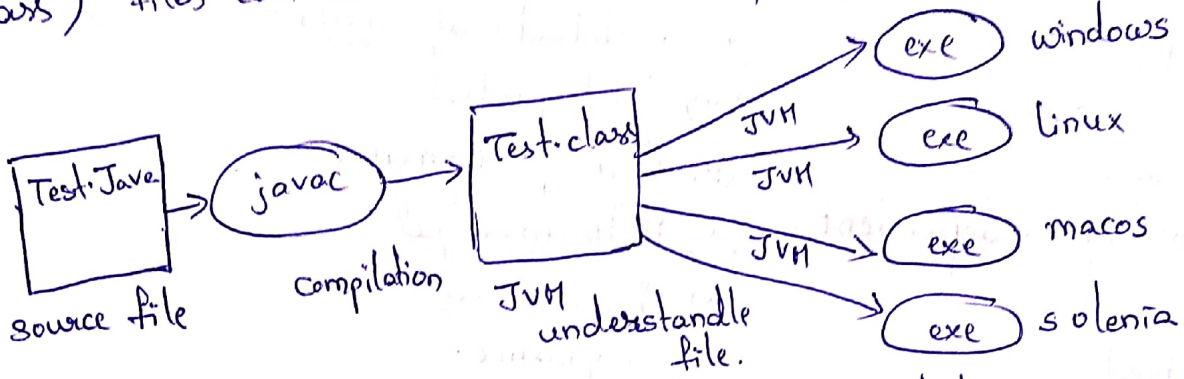
```
class Test {
    Test t=new Test();
    class data;
}
```
→ (data obj)

The total java languages is dependent on object only hence we can say java is a object oriented technology.

ii, Object oriented : Java is object oriented because to represent

iii, Platform Independent :

- Compile the Java program on one OS that compiled .file can execute in any OS is called P.I. The Java is P.I. The java app. allows its application compilation one OS that compiled (.class) files can be executed in any operating system.



iv, Architectural neutral : Java technolgy app. compiled in one architecture (Hlw ---RAM, HDD) and that compiled program runs on any Hlw architecture is called A.N.

v, Portable : In Java tech. the apps are compiled and executed in any OS and any Architecture (Hlw) hence we can say java is a portable language.

vi, Robust : Any tech. if it good at two main areas it is said to be Robust in exception handling and memory allocation.

vii, Secure : To provide implicit security java provide one component inside JVM called Security.

viii, Dynamic : Java in dynamic tech., it follows DM allocation (at runtime the memory is allocated) and dynamic loading to perform the operations.

Class Note:

Subject : Oops..Using..Java

Faculty : .....

Topic: .....JVM.....Architecture, Datatypes, Variables.

Unit No. : 1.
Lecture No.: 4
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 4

ix, Distributed: By using java tech. we are standalone app. and distributed app. There are standalone, web app, distributed app.

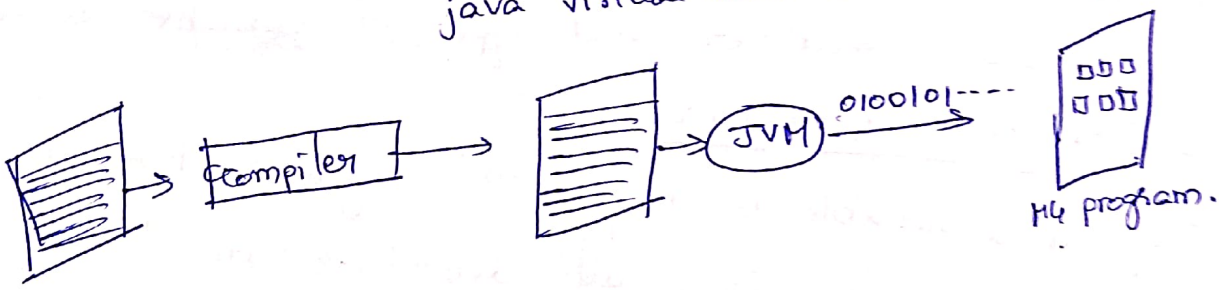
x, Multithreaded: Thread is a light weight process and a small task in large program.

xi, Interpretive: Java tech. is both interpretive and completive by using interpreter we are converting source code into byte code and the interpreter is a part of JVM.

xii, High performance: If any technology having features like robust, security, PI, dynamic and so on that technology is high performance.

JVM Architecture:

Java source file (*.java)
↓
java compiler (javac)
↓
java byte code file (*.class)
↓
java virtual machine (JVM).



Datatypes:

- Datatypes are used to represent the type of the variable and type of the expression.

- Datatypes are used to specify the how much memory is allocated for variable.

Datatype	Size	Range	Default values
Byte	1	-128 to 127	0
short	2	-32 768 to 32767	0
int	4	- 2147483648 to 214747	0
long	8	-2^{31} to $2^{31}-1$	0
float	4	-3.4e38 to 3.4e308	0.0
double	8	-1.7e308 to 1.7e308	0.0
char	2	0 to 65535	single space character
boolean	NA	NA	false, True.

Variable:

- Variables are used to store the value. By storing that values we are achieving the functionality of the project.
- While declaring variable we must specify the type of variable by using data types concept.
- In the java language we are having three types of variables.

i, Local variable: The variable which we are declare inside a constructor is called Local variable.

ii, Instance variable: The variables which are declare inside a class and outer of the method is called Instance variables

iii, Static variable: The instance variable which are declared as a static modifier such type of variable are called.

Scope and life time of variable:

Scope of a variable is the part of program where the is accessible like C/C++ in Java, all identifiers are lexically (or statiscally) scoped. .ie, scope of a variable can be determined at compile time and independent of function call stack.

Class Note:

Subject : ..Oops..Using..Java.

Faculty : .....

Topic: ..Scope..and..life..time of variables, Arrays.

Unit No. : 1
Lecture No.: 5
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 5

- Java programs are organized in the form of classes. Every class is part of some package. Java scope rules can be covered under following categories.

member variable (class level scope)

These variable must be declared inside the class ~~are~~ (outside any function). They can be directly accessed anywhere in class.

```
public class Test
{
    // All variable defined directly inside a class are
    member variable.
    int a;
    private String b;
    void method1() {---}
    int method2() {---}
    char c;
}
```

- We can declare class variables anywhere in class but outside methods.
- Access specified of member variable doesn't affect scope of them with in a class.
- Member variables can be accessed outside a class with following rules.
- Member variables can be accessed outside a class with following rules.

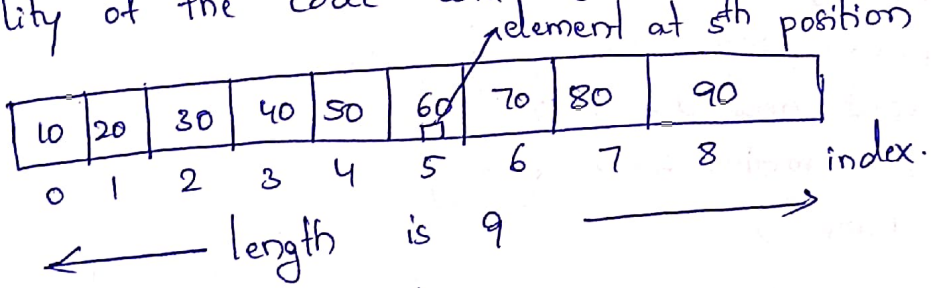
Modifier	Pkg	subclass	world
public	✓	✓	✓
protected	✓	✓	✗
default/no modifier	✓	✗	✗
private	✗	✗	✗

Arrays:→A

→ Array is a final class inheritance is not possible
→ Arrays are used to store the multiple numbers of elements of single type.
→ The length of the array is established at the time of array creation. After creation the length is fixed.
→ The items presented in the array are classed elements. Those elements can be accessed by index values. The index is begin from (0).

Advantages:

1. Length of the code will be declared.
2. We can access the element present in the array location.
3. Readability of the code will be increased.



Root:

java.lang.Object
---java.lang.reflect.Array
public final class arrary extends object.

Single/1D array declaration:

```
int [] a;
int [] a;
int a[];
```

Class Note:

Subject : ..Oop..using..Java

Faculty : .....

Topic: ....Operators..,..Control statements.

Unit No. : 1.
Lecture No.: 6
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 6

Declaration & instantiation & initialization:

Method 1 :- int a[ ] = {10,20, 80,40};

Method 2 : int[ ] a = new int [100];

```
a[0]= 10;
a[1] =20;
a[2] = 80;
a[4] =40;
```

Printing the array elements

```
class Test
{ public static void main(String[] args)
  {
     int[] a = {10,20, 80,40};
     System.out. println (a[0]);
     System.out. println (a[1]);
     System.out. println (a[2]);
     System.out. println (a[3]);
  } }
```

Operators:

Operators are used to perform operations on variables and values (+) operator to add together two values.

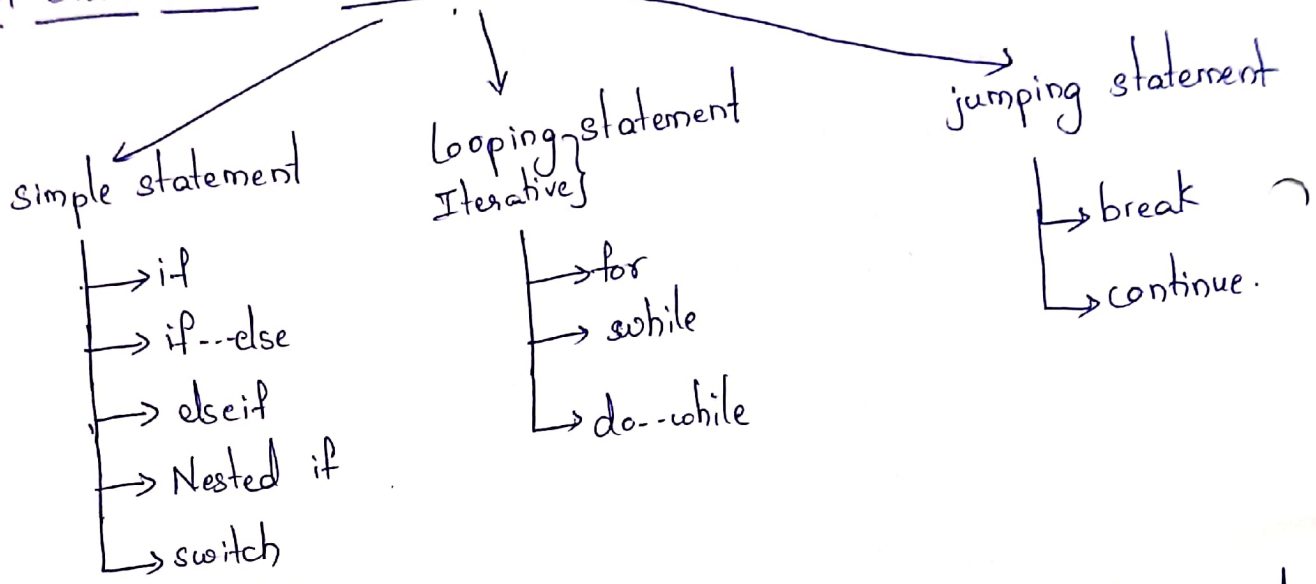
```
int a= 100+50;
int sum1 = 100+50 // 150
int sum2 = sum1+250; //400
```

int sum3 = sum2 + sum2; //800

Java divides the operators into following groups:

- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Bitwise operators.

Control statement(s) Conditional statements



Normal if statements

Use the if statement to specify a block of Java code to be executed if a condition is true.

```
if (syntax)
{
  // block of code
}
```

Ex: if (20>19)

```
{
  System.out.println("20 is greater than 19");
}
```

if---else statements

```
if (syntax)
{ // block of code to be executed }
```

Class Note:

Subject : ..Oops...using.java.

Faculty : .....

Topic: ...Control...statements

Unit No. : 1
Lecture No.: 7
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 7

```
else
{
  // block of data
}.
int time=20;
if (time <18)
{ System.out. println("Good day");      o/p: Good evening.
 }else{
    System.out. println ("Good evening");
  }
```

Else if ladder:-

```
int time =22;
if (time <10)
{ s.o.p ("Good morning");      o/p: Good evening.
 } Else if (time<18) {
    s.o.p ("Good. day");
 }else {
    s.o.p ("Good evening");
 }
```

Switch statement

```
switch(expression)
{
  case x; // block of code
          break;
  case y: // block of code
          break;
  default: // code block
}
```

Switch: The switch expression is evaluated once.

- The value of the expression is compared with the values of each case.
- if there is a match, the associated block of code is executed.
- The break and default keywords are optional, and will be described later in this chapter.

Ex:

```
int day=4;
switch (day)
{
  case 1: S.O.P ("Monday");
          break;
  case 2: S.O.P ("Tuesday");
          break;
  case 3: S.O.P ("Wednesday");
          break;
  case 4: S.O.P ("Thursday");
          break;
  case 5: S.O.P ("friday");
          break;
  case 6: S.O.P ("Saturday");
          break;
  case 7: S.O.P ("Sunday");
          break;
}
```

O/P: Thursday.

Class Note:

Subject : Oops...using..Java

Faculty : .....

Topic: ....Control...statements.

Unit No. : 1
Lecture No.: 8
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No.8

while loop:-

- Loops can execute a block of code as long as a specified condition is reached.
- Loops are handy because they save time, reduce errors and they make code more readable.

```
while (syntax)
{
  // code block to be executed.
}
```

```
int i=0;
while(i<5)
{
   s.o.p(i);
   i++;
}
```

o/p: 0 1 2 3 4

do..while loop statement:

- The do-while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as condition is true.

```
do{
   // code block
} while (condition);
```

Ex:

```
int i=0;
do{
   s.o.p(i);
   i++;
}while(i<5);
```

o/p: 0 1 2 3 4

for loop statement:

- When you know exactly how many times you want to loop through a block of code, use the for loop instead of while loop

Syn: for (st1; st2; st3)
{ // block of code
}

- st1 is executed (one time) before the execution of the code block.
- st2 is executed, condition for block of code.
- st3 is executed (every time) after the code block has been executed.

Ex:
```
for(int i=0; i<5; i++)
{ S.O.P(i);
}
```

o/p:
0
1
2
3
4

Jumping statements:

- The break statement can also be used to jump out of a loop.

```
for (int i=0; i<10; i++)
{ if(i==4)
  { break;
  }
  S.O.P(i);
}
```

o/p:-
0
1
2
3

continue:

- The continue statement breaks one iteration (in the loop), if a specified conditions occurs, and continues with the next iteration in the loop.

```
for (int i=0; i<10; i++)
{ if(i==4)
  { continue;
  }
  S.O.P(i);
}
```

o/p:
0
1
2
3
5
6
7
8
9

Class Note:

Subject : O.o.p...using...Java

Faculty : .....

Topic: .....Types conversion.. and casting, Simple Java pgm.

Unit No. : 1
Lecture No.: 9, 10.
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 9

Type conversion and casting:

- There are two terms { i, type casting, ii, type conversion } are used in a program to convert one data type to another type.
- The conversion of data type is possible only by the compiler when they are compatible with each other.

Type casting:

- When a datatype is converted into another type by a program or user while writing a program code of any programming language, the mechanism is known as Type casting

Syn: definition_dt= (target_dt) variable;

(dt) is a casting operator.

Type conversion:

- If a dt is automatically converted into another dt at compile time is known as type conversion.
- The conversion is performed by the compiler if both dt are compatible with each other.
- Remember that the destination dt. should not be smaller the source type it is also known as widening conversion.

Simple Java programs:

```
class Test
{
    public static void main(String args[]) // main menu
    {
        System.out.println("Hai CSD");
    }
}
```

olp: Hai CSD.

Constructors:

- A constructor is a block of codes similar to a method. It is called when an instance of the class is created.
- At the time of calling constructor, memory for the object is allocated in the memory.
- It is a special type of method which is used to initialize the object.
- Every time an object is created using the new(), atleast one constructor is called.
- There are 2 types of constructors in java.
    - i) No-args / zero constructor
    - ii) Parameterized ".

Rules for creating java constructor:

→ Constructor name must be same as its class name.

→ A constructor must have no explicit return type.

→ A java constructor cannot be abstract, static, final and synchronize

Class Note:

Subject: Oops using Java.

Faculty: .....

Topic: Constructors, Methods., static data, static method, Strings, Java API document.

Unit No.: 1.
Lecture No.: 11, 12
Link to Session
Planner (SP) S.No. .....of SP
Date Conducted:
Page No. 10

Methods:

- A method is a block of code which only runs when it is called and can pass data, known as parameters into a method.
- Methods are used to perform certain actions, and they are also known as functions.
  - → A method must be declared within a class.
  - → It is ~~declared~~ defined with the name of the method, followed by paranthesis ( ).

Ex: System.out.println()
↳ paranthesis.

Static block in Java:

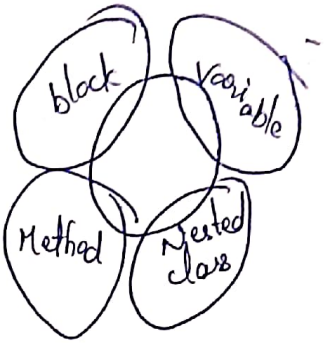
- When a block is decorated or associated with the word static, it is called a static block.
- Static block is known as the static class.
- A static block can be used for the static initialization of a class
- The code that is written inside the static block run code When the class is getting loaded into the memory.

Static data:

- The static keyword in java is used for memory mgt.
- We can apply static keyword with variables, methods,

blocks and nested classes.
- The static keyword belongs to the class then an instance of the class.

The static can be:
→ Variable (also known as a class variable)
→ Method (" " class method)
→ Block.
→ Nested class.



String and String Buffer classes:-
- String is sequence of characters.
- which immutable which means that a constant cannot be changed once created.
- String buffer is synchronized .i.e., thread safe. It means two threads can't call the string buffer simultaneously.
- String Buffer is less efficient than String Builder.
- String Buffer was introduced in Java 1.0.

Using Java API documentr
- We can create document API in Java by the help of java doc tool.
- In this java file, we must use the documentation comment /** ... */ to post information for the class, method, constructor, fields etc.

```
Package com.asc; /** This class is a user-defined class that
                    contains one methods cube */
public class M{
 /** The cube method prints cube of the given number */
 public static void cube (int n)
 { System.out.println(n*n*n);
 }
```