Java

Programming language

Technology

C/C++

Java is platform independent.

Oops

C   c++ java   python

Procedural language

C++ has object oriented capability and also have extra libraries.

Speed of C++ > Java

- Memory management in c/c++ is done by the programmer
- Malloc calloc .. structures.
- It is managed automatically
- Gaming purpose / fast streaming / online games c++ over java

Java was developed by sunmicrosystem, oops concept.

WORA (write once run anywhere)

Types of **java application:**

1. Standalone Application: example – antivirus software
2. Web Application: dynamic web pages, applets, Servlet, JSP,Spring, Hibernate, JSF, Struts etc.
3. Enterprise Application: Amazon, high level security. J2EE.. EJB is used here.
4. Mobile Application: Android, iOS , kotlin.

Java Editions:

1. **Java SE Java Standard** Edition / Platrform : libraries, String, Core Java, Swing, Collection.

2. Java EE: all the libraries, web application developed

3. Java ME: dedicated for mobile application
4. Java FX: internet application to develop lightweight API.

Java Current version 17 (September)

Java 8.

Properties/ features of java

- OOPS
- Interpreted language
- Multithreading
- Portable
- JDK JRE JVM JIT

Java program (program1.java) ---program1.class

Java Variable:

Variable: container

Types of variable:

- Local
- Global
- Static: static keyword is used. You can not use it as local variable. Memory allocation happens only once.
- Instance: inside the class outside the method. We can not declare it as static.

Data Types in Java:

Primitive Data Type: int, Boolean, byte, char etc.

Non-Primitive Data Type : classes, interfaces, array

Operators:

- Unary
- Arithmetic
- Shift
- Relational
- Bitwise
- Logical
- Ternary
- Assignment

■

```
\n
\b
\t
\f : form feed
```

Math in Java:

Math.max(a,b)

A B   A OR B

1 1	1
10	1
00	0
01	1

--- If Else

---- Switch

--- While loop

--- For loop

---- Break/ Continue

---- if else

```
If (condition)
{
}

Else
{   }
```

---- if “else if” else

--- “if” keword in If statement is always the first statement and ‘else’ keyword is always the last statement in control statement.

---- there are (10, 20,30) numbers and find out the largest among all using else if.

Decision Statements:

- If statement
- Switch

Loops statements

- Do while
- While
- For
- For each

Jump statements:

- Break
- Continue

Switch------------

If-else if-if

Contains multiple blocks of code called as cases. And a single case executed based on variable being switched.

Syntax:

```
Switch (expression)
{
Case value1:
Statement:-------------
Break;
Case value2:
Statement--------
Break;
-
-
-
Case n:
Statement -----
Break;
Default:
Defult statement
}
```

WAP for implementing all the operators. (choose 3)

WAP to find out prime number.

WAP to check year is LEAP year or not.

WAP to find sum of first 10 number.

WAP to implement switch statement. (any use case)

LOOP:

Iteration

```
Sytem.out.println("Hello");
Sytem.out.println("Hello");
Sytem.out.println("Hello");
Sytem.out.println("Hello");
Sytem.out.println("Hello");
```

Redundancy : repetition of code must not be there in programming.

```
I=1
While(i<5)
{
Sytem.out.println("Hello");
```

}

---- if you have the starting and ending defined then you can use for/while but prefer for loop.

----Example for while loop: We chat (end)

While (i==end)

```
****
****
****
****
****
```

Class & Object:

User defined data type.

Object: real time entity.

OOPs (Object Oriented Programming System)

--- Abstraction

---- Encapsulation

---- Inheritance

---- Polymorphism

Inheritance: parent and child, to achieve runtime polymorphism.

Polymorphism: Taking more than one form Operation Overloading.

Abstraction: hiding the exact code from end user.

Encapsulation: wrapping up / combining of data and method into single unit.

Example: capsule, lunch box, car engine

Method:

Functions, reuse it.

Access Modifier:

- Public: accessible to all the classes
- Private: method is accessible only in the class where you have defined
- Protected: within the same package or subclass in different package
- Default: only for same package

```
int fun1 ()
{
}
```

---- Predefined methods

----- User-defined methods.

```
public static void main(String[] args)
void fun1()
{}
```

There are two ways to call the function:

- With the help of class
- With the help of object

If you are defining methods as static so you can directly call that function / you call it by classname dot method.

Constructor:

--- block of codes similar to the method.

---- name should be same as class name

---- invoked by new keyword

---- new keyword is used to create the object of class

---- it should not return something

---- it can not be abstract, static, final.

Types of constructors:

- Default constructor (no arguments)
- Parameterized constructor

```
FirstClass obj1 = new FirstClass();
```

Polymorphism

Method Overloading:

Method Overriding:

```
FirstClass()
{
}
```

Static keyword:

1. Variable
2. Method
3. Block
4. Nested class

```
Static int a;
```

Static block: executed before main method and at the time of class loading.

```
static
{
Statement(s)
}
```

This keyword:

---- it is a reference that refers to the current object.

--- to invoke current class method

--- to invoke current class constructor

----- to invoke the current class

Inheritance:

```
Class A

{

}


Class B extends class A

{

}
```

- Single level Inheritance
- Multilevel Inheritance
- Multiple Inheritance (We can not perform this in java)

```
Class A

{

}


Class B extends class A

{

}

Class C extends class B

{

}
```

Multiple Inheritance:

```
Class A
{
}

Class B
{
}
Class C extends class B, class A
{
}
```

Method overriding: // runtime polymorphism