Java: Oracle product

James Gossling

Mr khosla

What is Classes: A class is a template or blueprint that which containes combination of data members and member function. Member of the class

can be accessible by using instance of class.

Class Class\_name

{

data members;

member function;

}

What is Object: Object is real world Entity which provide 3 features;

Car: Object

1) State: Moving, parked

2) Behaviour: left turn, right turn

3) Identity: Registration

Object oriented: Java support all the object oriented programming: It is approach that provide a way of Modularizing programming, in which it used to

create small partition memory area for both data and object, Whic we can use as temmplate for creating copies of such module on demand.

OOPS Features:

1) Encapsulation: Wrapping of data and function into single unit.

2) Abstraction: Hidding the irrelevant details of object that do not contribute to its essential charachrastic known as Abstartion

\*\*\* Encapsulation focus on showing relevant details where abstraction focus on hiding the irrilevent details.\*\*\*

3) Inheritance: Deriving the new entity from already existing entity, we can reduce size of programming.

4) Polymorphism: Ability to take more then one form is known as polymorphisam.

Ex: Learners, Developer, Son,

Access Modifier: private: Allow you to access memebrs with in current class

public: Allow you to access memeber in entire application.

protected: Access member with in derived class

Friendly: H/W

Data type: Two types of data types: 1) value types:Int, float, double, long, char,boolean, byte Value type will be stored in Stack memory

2) Refrence type: String, object and Custom type: refrence type will be store in heap memory

-> Plateform Independent: window, mac, linux.

->Oops: Object Oriented Programming:

-> Applcation : web, mobile, console, Service Oriented, Desktop application, Network application.

--> Packages: Contain group of classes and interface that belong to one comman group, If you want to overcome the naming conflict between the class

we need packages.

Ex: I got message

Ex2: I got watsapp message.

Java Features: Simple: Java Use simple c and C++ syntex as programmer and no need to take care of internal details like memory management, How much memory

required to run programe , The same memory will be automatically allocated.

Compile and Interpreter: Java programe are compiled first during the process java code will convert into byte code then .class will be created

in interpretation it read byte code line by line and execute it.

Portable: Ability to run the programe without changing the source code, Java programes are programes that can be run without changing source

code.

Secure: It provide various level security: Compiler level: H/W

ByteCode: H/W

Class Loader Level:H/W

Distributed: Java support all the distributed and roboust application

Java Architecture:

Java Programe: Java code which can be stored with extension .java

Java Class: Java code after compilation it converted into byte code, then .class file will be created.

JVM: java virtual machine:

it containes components: Class Loader: used to load all classes which are required to run the programe

Two type of class loader: 1) Primordical Class Loader

2) Class loader Object

, JIT Compiler, Execution Manager

It provide run time env, which allow to run java programe, In this we have JDK and JRE

Java API

-------------------------------------------------------------------------------------------------------------------------------

How do you define method main() in java

public static void main(String[] args){

....

.....

.....

}

How do i display a value and message in java

System.out.println("Welcome to java Progarmming"):

System--> class

out---> Console

Print--> Method

-----------------------------------------------------------------------

Class: Declare a class in java

Public: Access Modifier which represent visibility, It is visible to all

Static: Static is keyword. Advantage of static method is that there is no need to create object to invoke the static method. The main method is executed by JVM.

Void: it is the return type of the method. It does not return any value.

main: Represent the start of the programe/

String[]args: It is used for command line atgument

System.out.println("Hello"); : Used to print statement.

--------------------------------------------------------------------------

Class Demoprograme {

public static void main(String[] args)

{

System.out.println("Welcome to java");

}

}

Demoprograme.java

After Compile

Bytecode

Demoprograme.class

Path: JDK path set for global environment.

JVM: Java Virtual Machin: It provide runtime environment in which java byte code can be executed.

Main task for JVM:

Load code

Verify Code

Execute Code

Provide run time environment

JRE: Java Run time Environment. It is the implantation of JVM. It contain libraries and use at run time.

JDK: Java development Kit:

Java keywork:

Byte, else, if, do, while, for, switch, return, throw, new etc.

Variable: Variable are nothing but reserved memory location to store values.

1) Local Variable: Variable define inside method, constructor or block are called local variable.

2) Class variable( Static variable): H/w

3) Instance variable: When we declare in class and outside of the method.

Java Identifier: Names used for classes, method and variable called Identifier.

· Identifier are case sensitive.

Valid Ex: age, $salary, \_value, \_1\_value

Non Valid: 123abc, -Salary

Datatype in java: type of value you want to store in variable called datatype.

Follow Tree Structure:

1) Primitive datatype:

Boolean: boolean: true or false

Numeric: int, byte, short, double, long,

byte Ex: student strength

Class Demoprograme {

public static void main(String[] args)

{

byte b= (byte)128; // range of byte -128 to 127

System.out.println (b);

}

}

o/p: -128

Class Demoprograme {

public static void main(String[] args)

{

byte $b= 20;

byte $a= 10; // range of byte -128 to 127

System.out.println ($a+b$);

}

}

o/P: 1020

o/p: 30

2) Non primitive datatype or reference type:

String, Array etc.