Day 3

Java Buzzwords

1. Simple
2. Object Oriented
3. Architecture Neutral
4. Portable
5. Robust
6. Multithreaded
7. Dynamic
8. Secure
9. High Performance
10. Distributed.

Java is simple

- Java programming language is derived from C and C++.
- A syntax of Java is simpler than C and C++ hence Java is considered as simple.
    1. No need to include header files.
    2. Java do not support structure and union.
    3. There is no concept of declaration and definition. Everything is definition.
    4. We can not define any element global.
    5. Java do not support operator overloading
    6. Java do not support delete operator and destructor.
    7. Java do not support friend function and class.
    8. Java do not support private and protected mode of inheritance.
    9. Java do not support multiple implementation inheritance.
    10. Java do not support default argument, constructor's member initializer list, copy constructor, const member function etc.
    11. We can not declare pointer in Java. Hence java do not support pointer arithmetic.
- S/W which is required to develop java application is small in size(180 MB).

Java is object oriented.

- Object oriented programming structure(OOPS) is invented by Alan Kay int 1960.
- Grady Booch is inventor of Unified Modelling Language(UML).
- According to Grady Booch there are 4 major and 3 minor elements /parts / pillars of OOPS.

4 Major pillars

1. Abstraction : To achive simplicity
2. Encapsulation : To achive data hiding & security
3. Modularity : To minimize module dependacny
4. Hierarchy : To achieve resuability

- Here word "Major" means language without any one of the above feature will not be OO.

3 Minor pillars

1. Typing/Polymorphism : To reduce maintenance of application
2. Concurrency : To utilize H/W resc(CPU) efficiently
3. Persistance : To maintain state object on HDD.

- Here word "Minor" means, if language support to these features then it will be considered as useful but not essential to classify language OO.
- Since Java support all major and minor pillars of oops, it is object oriented.

Java is Architecture Neutral programming language.

- Following are CPU architecture:

1. X86/X64
2. ARM
3. Power PC
4. SPARC
5. ALPHA
6. Itanium
7. IBM-z

- Java compiler is responsible for converting java source code(.java) into compiled code / bytecode(.class).
- Bytecode is architecture neutral code hence Java is considered as Architecture Neutral.

```
//File Name : Program.java
class Program
{
    public static void main( String[] args )
    {
        System.out.println("Hello World!!!");
    }
}
```

- Compilation steps

```
javac Program.java  //Output : Program.class
java Program        //Output : Hello World!!!
javap -c Program.class  //Output : print bytecode
```

- javap is Java class file disassembler
- javap -c : Prints out disassembled code, that is, the instructions that comprise the Java bytecodes.
- Bytecode is an object oriented assembly language code designed for JVM.

Java is portable

- Term portable is associated with executable.
- ".class" file is executable file of java.
- Since Java is architecture neutral, it is portable too.
- On all the platforms size of data type is constant/same hence Java is truly portable:
    1. boolean : Not specified
    2. byte : 1 byte
    3. char : 2 bytes
    4. short : 2 bytes
    5. int : 4 bytes
    6. long : 8 bytes
    7. float : 4 bytes
    8. double : 8 bytes
- Since Java is portable it doesn't support sizeof operator.

Java is a Robust Programming Language

1. Architecture Neutral
2. Object Oriented
3. Memory Management
4. Exception Handling

Java is Multithreaded programming language.

- Program is collection of statements. Any instruction given to the computer is called statement. Every instruction is made of tokens. Hence token is basic unit of program. ".java" file contains Java program.
- Program in execution / running instance of a program is called as process / task.
- Process may contain one/more threads. Thread is a sub process/light process.
- Single Tasking : An ability of OS to execute single process/task at a time is called single tasking.
- Multi Tasking : An ability of OS to execute multiple processes/tasks at a time is called multi tasking.
- We can achive multitasking using 2 ways:
    1. Process
    2. Thread
- If any application uses single thread then it is called single threaded application. If any application uses multiple threads then it is called multi threaded application.
- When JVM starts execution of java application(.class), It starts execution 2 threads:

1. Main thread:
    - It is User Thread.
    - It is responsible for invoking main method.
2. Garbage Collector(GC).
    - It is also called as finalizer
    - It is Daemon Thread.
    - It is responsible for deallocating/releasing/reclaiming unused memory.

- Due to these two threads every java application is multi threaded.

C++ Versus Java Terminologies:

- C++ Java

1. Data member Field
2. Member Function Method
3. Class Class
4. Pointer Reference
5. Object Instance
6. Access Specifier Access Modifier
7. Namespace Package
8. Base class Super class
9. Derived class Sub class
10. Derived From Extends From

```
namespace std   //std is namespace
{
    //Base class
    class Person    //Person is a class
    {
    private:    //Access Specifier
        string name;    //Data Member
        int age         //Data Member
    public:   //Access Specifier
        void showRecord( void )     //Member Function
        {   }
    };
    //Derived class
    class Employee : public Person
    {   };
}
```

```
package std   //std is package

//Super class
class Person    //Person is a class
{
    private string name;    //field
    private int age         //field
    public void showRecord(  )     //Method
    {   }
};
//Sub class
class Employee extends Person
{   };
```

Coding/Naming Convention

- To write Java program, two types of conventions are suggested:

1. Camel Case convention
2. Pascal Case convention

Camel Case Coding Convention

- Example:
    1. main
    2. parseInt
    3. showInputDialog
    4. waitForPendingFinalizer
- In this convention, execpt first word, first character of each word must be in upper case.
- We should use thid convention for:
    1. Method parameter and method local variable.
    2. Field
    3. Method
    4. object reference.

Pascal Case Coding Convention

- Example:
    1. System
    2. StringTokenizer
    3. NullPointerException
    4. StringIndexOutOfBoundsException
- In this convention, including first word, first character of each word must be in upper case.
- We should use thid convention for:
    1. Enum, Class, Interface name.
    2. File Name

name of the package should be in lower case

- Example:
    - java.lang
    - javax.servlet.http
    - com.mysql.jdbc
    - org.sunbeam.cjo1;

Name of final field and name of members of enum should be in upper case.

```
enum Color  //Pascal Case Convention
{
    RED, GREEN, BLUE;   //Enum Members
}
class Math
{
    final double PI = 3.14;
}
```

Data Type

- It describes size, nature, range and operations that we can perform on object.
    1. Size : How much memory is required to store the data.
    2. Nature : Which kind of data memory can store
    3. Operations : Which operations are allowed to perform on the data store inside memory.
    4. Range : How much data, we can store inside memory.
- Types of data type:
    1. Primitive Data types / Value Types
    2. Non Primitive Data types / Reference Types

Primitive Data types

- Primitive types are also called as Value Types.
- Primitive Type Size Default Value Wrapper Class

1. boolean : Not specified false java.lang.Boolean
2. byte : 1 byte 0 java.lang.Byte
3. char : 2 bytes \u0000 java.lang.Character
4. short : 2 bytes 0 java.lang.Short
5. int : 4 bytes 0 java.lang.Integer
6. long : 8 bytes 0L java.lang.Long
7. float : 4 bytes 0.0f java.lang.Float
8. double : 8 bytes 0.0d java.lang.Double

- In Java, primitive types are not classes. But for every primitive type java has provided a class, it is called Wrapper class. All the Wrapepr classes are declared in java.lang package.

Non Primitive Data types

- Non Primitive types are also called as reference Types.

1. Interface
2. Class
3. Enum
4. Array