Day 2

JDK's Installation Directry Structure

- "C:\Program Files (x86)\Java\jdk1.8.0_251" is JDK's Installation Directry.
- bin:
    - It contains java language tools.
    - e.g javac, java, javap, jdb etc.
- include:
    - In context of java, C/C++ code is called native code.
    - Java Native Interface(JNI) is java's framework, which helps us to communicate with native code.
    - If we want to access native code in java then we need to include some header files into native code. Such files are available in include directory.
- jre:
    - It contains rt.jar and JVM implementation.
- lib:
    - It contains supporting lib. files required for thrird party tools.
- src:
    - It contains source code of Java API.
- db:
    - It contains database specific files.
    - e.g Sample databases, JDBC drivers etc.

History of Java

- Invented by James Gosling and his Team[Green Team]
- It is developed at SunMicrosystems in 1991. Now it is product of Oracle.
- Initial name of Java was "Oak". Later it was renamed to Java.
- Java is Object oriented programming Langguage.
- Current platform of java is Java Platform 14.
- For Java Platform 8 Software is JDK 1.8.

Relation between src.zip, rt.jar, java docs

- API = { enum, class, interface }
- src.zip : It contains source code of java API
- rt.jar : It contains compiled code of Java API
- java docs : It contains documentation of Java API

Simple Hello Application

```
//File Program.java
class Program
{
    public static void main(String[] args)
    {
```

```
        System.out.println("Hello World!!!");
    }
}

javac Program.java  //Output : Program.class
java Program
```

Entry Point Function

- According Java Language Specification(JLS) name of entry point function must be main.
- Syntax:

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

- JVM is responisble for invoking main method.
- We can overload main method in java.
- Per class, we can define main method but only one main method can be considered as entrypoint method.

What is meaning of "System.out.println"

- System is a final class declared in java.lang package.

```
package java.lang;
public final class System extends Object
{
    //Data member/field of System class.
    public static final InputStream in;     //System.in
    public static final PrintStream out;    //System.out
    public static final PrintStream err;    //System.err
}
```

- out is reference of java.io.PrintStream class and it is public static and final field of System class.
- print, println and printf are overloaded non static methods of java.io.PrintStream class.