JAVA Assignment

19BQ1A05N8

Q.) What are the Components of JAVA platform? Explain. Write a Java program to illustrate the Usage of Conditional statements and looping statements.

A) **Java Platform :-**

Java platform is a software or collection of programs that help us to exeute applications written in Java programming language. A Java platform Consists of a Java Compiler, a set of libraries, and an exeution engine.

- Java platform is independent of any Particular OS which makes Java Programming a platform-independent language.

★ Java Platform Consists of the following components.

→ Java Language.

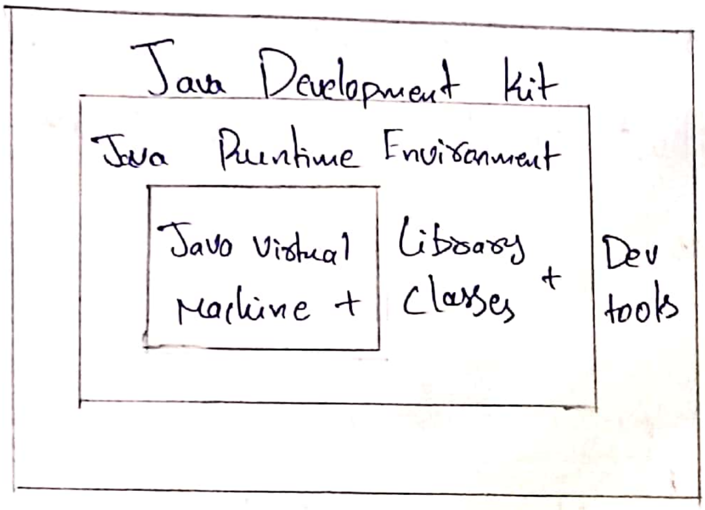
→ The Java Development kit (JDK)

→ The Java Runtime Environment (JRE)

→ The Java Compiler

→ The Java Virtual Machine.

- Apart from the above main Components, the Java platform also contains garbage collectors, a set of libraries and other additional Components and tools that are required to efficiently run the Java applications.



JRE = JVM + Library classes

JDK = JRE + Developer tools.

→ Java Language :-

Java is a programming language that the Java platform uses. Java is an object-oriented programming language whose syntax is derived from C and OOPS features are derived from C++. It has its syntax, rules, format and programming paradigm.

→ The Java Compiler :-

- This is a compiler for Java programming language and its function is to generate Java class files from the Java source code. Java class file contains a platform-independent Java byte code.

- After generating class files, JVM loads these class files and either interprets the byte code or compile it to machine code Using Just-in-time (JIT) compiler.

→ The Java Virtual Machine (JVM):-

- JVM is the center of Java programming language and Java platform. The JVM converts the byte code into a machine-specific code.

- JVM provides the functionality of garbage collection, memory management, security etc. JVM is platform-independent and we can customize its functionality using a virtual interface it provides which is not machine-dependent and is also independent of the OS.

- This platform-independence of JVM allows us to create Java programs on one machine and execute them on another machine (WORA - Write-Once-Run-Anywhere).

→ The Java Runtime Environment (JRE):-

JRE, as the name suggests, is the runtime environment that is required to execute Java programs and applications. JRE Consists of JVM and binaries and other classes to Succesfull exeute Java Programs.

- JRE is a subset of JDK and doesn't contain any development tools such as Java Compiler, debugger etc.

- The JRE includes the following Components.
  ↳ Code libraries, Property settings and resource file.
  ↳ DLL files
  ↳ Java extension files
  ↳ Files
  ↳ Applet Support classes
  ↳ True Type font files.

★ To exeute any Program/application written in Java, you need JRE installed in your system.

→ The Java Development Kit (JDK):-

- This is the core component of any Java environment. JDK contains JRE along with Java Compiler, Java debugger and other core (Components), classes.

- JDK is used for Java development as it provides the entire executable and binaries as well as tools required to compile, debug a Java Program.

- JDK is a platform-specific Softwore and thus we will have separate JDK installers for each OS.

★ JDK contains the following Components:

- jConsole :- This is a Java monitoring and management Console.
- jar :- This is the archiver. This tool is used to Package related class libraries into a single jar file as well as to manage jar files.
- jarSigner :- This tool is used for jar signing and Verifying
- javap :- This is a tool used for class file disassembler.
- javaws :- Java webstart launcher for JNLP applications.
- jstack :- Utility used to print stack traces for Java threads
- Javadoc :- This automatically generates documentation from the Source Code Comments.
- applet Viewer :- Used for applet execution and debugging without a web browser.
- apt :- Annotation Processing Tool.
- Key tool :- Using this utility you Can manipulate keystore.
- xjc :- This is a Part of XML binding (JAXB) API that accepts XML Schema and generates Java classes.

→ All these including Java language are Components of a Parent entity Called Java Platform which is a environment that helps to run Java Program.

★ Example to illustrate the usage of of conditional and looping statements

```
import java.util.*;
public class Example {
    static Scanner sc =new Scanner(System.in);
    Void SumOfDigits(){
        int r=0;
        int n= sc.nextInt();
        while(n!=0){
            r+=n%10;
            n/=10;
        }
        System.out.println("Sum of Digits:"+r);
    }
  void pattern(){
    int n=sc.nextInt();
    for(int i=0;i<=n;i++){
        for (int j=1; j<=i; j++)
            System.out.print(j+" ");
        System.out.println();
    }
  }
  public static void main(String[] args){
    Example ex= new Example();
    int n= sc.nextInt();
    switch (n){
        Case 1: ex.sumOfDigits(); break;
        Case 2: ex.Pattern(); break;
      default :System.out.println("plz enter 1 or 2"); } }
```

19BQ1A05N8

Q2) Write any six significant differences between Procedure Oriented Programming and Object Oriented Programming. why JAVA is Robust Programming language? Explain.

A)

Procedure Oriented Programming	Object Oriented Programming
• Program is divided into small parts called functions	• Program is divided into small parts called objects.
• Follows top-down approach	• Follows bottom up approach
• Has no access specifier	• Have access specifier like private, public, protected etc.,
• Adding a new date and function is not easy.	• Adding a new data and function is easy.
• Doesn't have proper way of hiding data, so it is less secure	• Provides data hiding so it is more secure.
• Overloading is not possible	• Overloading is possible.
• Function is more important than data	• Data is more important than function.
• Based on unreal world.	• Based on real world.
Ex: C, FORTRAN, Pascal, Basic etc.	Ex: C++, Java, Python, C# etc.

JAVA is a Robust language. Below are the features which make Java Programming language Robust.

- Builtin Memory Management :- Memory allocation/deallocation is performed internally in Java and pointers are not exposed to the developer. Hence run time Segmentation Fault kind of errors (due to Pointer misuse) do not occur (or very rare).
- Garbage collector :- Since garbage collector automatically cleans unreferenced objects, memory leaks are controlled.
- Exception handling :- avoids Applications crash & lets programmer to easily handle exception scenarios, and improves Robustness.
- Certain features of Java Compiler such as strongly typed :- avoids automatic conversion, which reduces unexpected run time behaviour.

→ So, robustness characteristic of Java lets Java Applications to run with minimal/no run time errors relatively.

Q3) Define a class ParkingLot with the following descriptions:

Instance Variables/data members:

int Vno - To store the Vehicle number.

int hours - To store the number of hours the vehicle is parked int the Parking lot

double bill - To store the bill amount

Member Methods:

Void input () - To input and store Vno and hours

Void Calculate() - To compute the parking charge at the rate of Rs.3 for the first hour or part thereof, and Rs.1.50 for each additional hour or part thereof.

Void display() - To display the detail.

Write a main method to create an object of the class and Call the above methods.

19BQ1A05N8

A)

```
import java.Util.* ;
class ParkingLot {
    int Vno, hours;
    double bill;
    Void input() {
        Scanner sc =new Scanner (System.in);
        Vno = Sc.nextInt();
        hours = Sc.nextInt();
    }
    Void Calculate() {
        bill = 3+(hours-1)* 1.50;
    }
    Void display() {
        System.out.println("Total Amount: Rs."+bill);
    }
    Public static Void main (String []args) {
        ParkingLot  parkinglot =new ParkingLot();
        parkinglot.input();
        parkinglot.Calculate();
        parkinglot.display();
    }
}
```

Sample input:-

```
1234
6
```

Sample output:-

```
Total Amount: Rs.10.5
```

24) Design a class to overload a function Joystring() as follows:

i) Void Joystring(String s, char ch1, char ch2) with one string and two character arguments that replaces the character argument ch1 with the character argument ch2 in the given string s and prints the new string.

Ex:
Input:
s = " TECHNALAGY"
ch1= 'A'
ch2='O'
Output : " TECHNOLOGY

ii) Void Joystring (String s) with one string argument that prints the position of the first space and the last space of the given string s.

Ex: Input:
S = "Cloud computing means Internet based Computing"
First Index: 5
Last Index: 36

iii) Void Joystring(String s1, String s2) with two string arguments that combines the two strings with a space between them and prints the resultant string.

Ex: Input :
S1= " COMMON WEALTH"
S2 = " GAMES"

Output : " COMMON WEALTH GAMES"

A)
```
import java.util.*;
public class Overload {
      Void Joystring(String s, char ch1, char ch2) {
          String op="";
          for(int i=0; i<s.length(); i++) {
              char ch=s.charAt(i);
              if  ((h==ch1)
                      ch= ch2;
              op+ =ch;
          }
        System.out.println(op);
      }
      Void  Joystring(String s) {
          int in=0;
          int sp=0;
          for(int i=0; i< s.length(); i++) {
              char ch= s.charAt(i);
              if ((h==' ') {
                  in=i;
                  ++sp;
                  if (sp==1)
                          System.out.println("First Index: "+in);
          }
        }
        System.out.println("Last Index:"+in);
    }
    Void Joystring(String s1, String s2){
        System.out.println(s1+"  "+s2);
    }
    public static void main(String[] args){
        Overload obj= new Overload();
        String s= "TECHNALAGY";
        Char ch1= 'A';
        Char ch2='O';
        String ss="Cloud computing means Internet based Computing";
        String s1="COMMON WEALTH";
        String s2="GAMES";
        obj.Joystring(s,ch1,ch2);
        obj.Joystring(ss);
        obj.Joystring(s1,s2);
    }
}
```

Output:-

```
TECHNOLOGY
First Index: 5
Last Index:36
COMMON WEALTH GAMES
```

Source:-

Q1

- Softwaretestinghelp.com

Q2

- geeksforgeeks.org
- quora.com