12/9/2020 JAVA ASSIGNMENT-1 CSE-D

Q.1 What are the components of JAVA platform? Explain. Write a java program to illustrate the usage of conditional statements and looping statements.

A platform is the hardware or software environment in which a program runs. The java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware based platforms.

The Java platform has two components

- Java Virtual Machine (JVM)
- Java Application programming Interface (API)

A Java Virtual Machine (JVM) is a virtual machine that enables a computer to run Java programs as well as Programs written in other languages, also compiled to Java bytecode.

The API is a large collection of ready-made software components that provide many useful capabilities. It is graped into libraries of related classes and interfaces; these libraries are known as packages.

Java has the following conditional statements:

- if
- if-else
- nested-if
- if-else-if
- Switch case

If Statement: It is used to decide whether a block of statements will be executed or not. i.e if a statement is true then the block of statements is executed otherwise not. The 'if' structure is called as a single selection structure because it selects or ignores a single action

```
class If
{
    public static void main (String s[])
    {
        int n=26;
        if (n%5==0)
            System.out.println("25 is a multiple of 5");
        System.out.println("This is not a part of If block");
    }
}
```

If-else:- This is used when a condition is true it will execute a block of statements and if the condition is false it won't. The else statement is used along with the if Statement to execute a block of code when the condition is false.

```
Class IfElse
{
    public static void main(String args[])
    {
        int i=11;
        if (i<10)
            System.out.println(i+" is smaller than 10");
        else
            System.out.println(i+" is longer than 10");
    }
}
```

iii) Nested-if :-

Nested if statements means an if statement inside an if statement.

```
Class Nested If
{
   public static void main(String args[])
   {
      int n=7;
      if (n>0)
      {
        if (n<10)
            System.out.println(n+ " is less than 10");
        if (n>5)
            System.out.println(n+ " is greater than 5");
        else
            System.out.println (n+ " is greater than zero
                                   and less than 5");
      }
   }
}
```

if-else-if ladder:-

As soon as the condition is met, the corresponding set of statements get executed, rest gets ignored. If none of condition is met then the statements inside "else" gets executed.

```
Class IfElseIf {
    Public static Void main (String args[]) {
     int  num = 1234;
     if (num < 100 && num >= 1) {
        System.out.println ("Its a two digit number");
     }
     else if (num < 1000 && num >= 100) {
        System.out.println (" Its a three digit number");
     }
     else {
      System.out.println (" Number not between 1 and 999");
     }
    }
}
```

Switchcase: It is a multiway branch statement.

```
class Switchcase
{
  public static void main (String args[])
  {
    int i=6;
    Switch (i)
    {
      case 0:
          System.out.println (" i is zero");
          break;
      case 1:
          System.out.println (" i is one");
          break;
      case 2:
          System.out.println (" i is two");
      default:
          System.out.println (" i is greater than two");
    }
  }
}
```

Looping statements are the statements that execute one or more statements repeatedly several number of times.

For loop:- The for loop is used when you know exactly how many times you want to loop through a block of code.

Syntax:

```
for (initialization condition; testcondition; increment/decrement)
{
  Statement (s)
}
```

```
Class ForLoopExample {
    public static void main (String args[]) {
        for (int i=10; i>1; i--) {
            System.out.println(" The value of i is: "+i);
        }
    }
}
```

While loop: A while loop iterates through a set of statements till its boolean conditions returns false. i.e when we do not know the exact number of iterations.

Syntax:

```
while (boolean condition)
{
    loop statements
}
```

Sum of digits using While loop

```
class whileloop
{
  public static void main (String args [])
  {
    int n=1106, s=0;
    while (n>0)
    {
      int r=n%10;
      s+=r;
      n=n/10;
    }
    System.out.println (" Sum of digits in 1106 is " +s);
}
```

Do While: Do while is similar to a while loop, execute that it executes atleast one time. It is an exit-controlled loop.

Syntax:

```
do
{
  Statements
}
While (condition);
```

```
class Dowhileloop
{
   public static void main(String args[])
   {
      int j=11;
      do
      {
        System.out.println("value of j = " +j);
        j++;
      }
      while (j<10);
   }
}
```

O/P:- value of j=11

Q2. Write any six significant differences between procedural Oriented Programming and object oriented programming. why Java is Robust Programming Language ? Explain ?

A:

Procedural Oriented programming	Object Oriented Programming
Program is divided into Small Parts called functions.	Program is divided into small parts called objects.
It follows top down approach	It follows bottom up approach

Adding new data and function is not easy.	Adding new data and function is easy.
There is no access specifier in Procedural programming.	OOP have access Specifiers like private, public, protected, etc.
Procedural Programming does not have any proper way for hiding data so it is less Secure.	Object Orlented Programming provides data hiding so it is more secure
In procedural programming, function is more important than data Ex:- C, Basic, Fortran, pascal etc.	In object oriented programming, data is more important than function. Ex: Java, Python, C++, C# etc.

Java is Robust because it contains exception handling. It is highly supported language, portable across many Operating Systems. Java also has feature of Automatic memory management and garbage collection. Bugs, especially System Crashing bugs, are very rare in Java.

3. Define a class Parking lot with the following description:
Instance variables / data members:
int Vno - To store the Vehicle number
int hours - To store the number of hours the vehicle is parked in 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 charges 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 method

→
```
import java.io.*;
import java.util.Scanner;
Class ParkingLot
{
   Int vno, hours;
   double bill = 0;
   Void Input()
   {
     Scanner sc = new Scanner ( System.in);
      vno = sc.nextInt();
      hours = sc.nextInt();
   }
   Void calculate()
   {
      if (hours > 1)
      {
         Bill = (hours - 1) * 1.0;
      }
      bill + = 3
   }
   Void display()
   {
     System.out.println("Vehical number : " +vno);
     System.out.println("No. of hours: "      +hours);
     System.out.println("Bill Amount : "     +bill);
   }
}
public class Parking
{
     public static Void main [String S[])
     {
       ParkingLot P = new Parking Lot();
```

```
        P.Input();
        P.calculate();
        P.display();
    }
}
```

Q4 Design a class to overload a function Joystring() as follows:
i) void Joystring (String s, char ch1, char ch2) with one string & two characters arguments that replaces the character argument ch1 with the character argument ch2 in the given string S and prints the newstring
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.
iii) Void Joystring (string S_1, string S_2) with two string arguments that combines the two strings with a space between them and Prints the resultant string

Example:
Input value of S_1 = "COMMON WEALTH"
S_2 = "GAMES"
Output: "COMMON WEALTH GAMES"

→
```
import java.io.*;
import java.util.Scanner;
class OverloadFune
{
  String S, S1, S2;
  char ch1, ch2;
  public void Joystring (string s, char ch1, char ch2)
  {
    for (int i=0; i<S.length(); i++)
    {
      if (S.charAt(i) == ch1)
      {
        S = S.replace (ch1, ch2);
      }
    }
```

```
            System.out.println(s);
    }
    public void Joystring (String s)
    {
        int FirstIndex=0, lastIndex=0;
        for (int i=0; i<s.length(); i++)
        {
            if (s.charAt(i)==" ")
            {
                FirstIndex = i;
                break;
            }
        }
        LastIndex = s.LastIndexof (' ')
        System.out.println(" FirstIndex :" + FirstIndex);
        System.out.println (" LastIndex :" + LastIndex);
    }
    void Joystring (String s1, String s2)
    {
      System.out.println (s1 + " " +s2);
    }
}

public class Overload
{
    public    static Void main (String args[])
    {
      Overload Func df = new Overload func ()
       Of . Joystring ("Technology", 'a', 'o')
       Olf . Joystring (" cloud computing means internet based computing");
       Olf . Joystring ("common wealth", "Games");
    }
}
```