19BQ1A05L2 CSE-D

17/9/2020

JAVA ASSIGNMENT-1

Q1. What are the components of JAVA platform? Explain. Write a java program to illustrate the usage of conditional statements and looping statements.

A. 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
- 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 grouped 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.

i) **if statement**: It is used to decide whether or not a statement or a block of statements will be executed

or not i.e. if a stc condition is true then the block of statements is executed otherwise not.

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

(ii) 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 larger than 10");
    }
}
```

(iii) Nested - if : Nested if statements means an if statement inside an if statement.

19BQ1A05L2

```
class NestedIf
{
    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");
        }
    }
}
```

(iv.) if-else-if ladder:-

```
class ifElseif
{
    public static void main(String args[])
    {
        int i=20;
        if(i==10)
            System.out.println("i is 10");
        else if(i==15)
            System.out.println("i is 15");
        else if(i==20)
            System.out.println("i is 20");
        else
            System.out.println("i is not present");
    }
}
```

v) Switch - case:- 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 thrithitwo");
                break;
            default:
                System.out.println("i is is greater than 2");
        }
    }
}
```

* Looping statements are the statements that execute one or more statement repeatedly several no.of times.

i) 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 ; test condition; increment/decrement)

```
{
    Statement(s);
}
```

```
class forLoop  // Printing 2 table
{
    public static void main(String args[])
    {

        for(int i=1; i<5; i++)
            System.out.println(" 2 * "+i+" = "+2*i);
    }
}
```

O/P:-
```
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
```

(ii.) **while loop**:- A while loop iterates through a set of statements till its boolean condition returns false. i.e when we do not know the exact number of iterations.

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

```
class whileLoop  // Sum of digits.
{
    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);
    }
}
```

(iii.) do-while :- Do while is similar to a while loop, except 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
1.) Program is divided into small parts called functions.	Program is divided into small parts called objects.
2.) It follows top down approach.	It follows bottom up approach.

3.) Adding new data and function is not easy.	Adding new data and function is easy.
4.) There is no access specifier in procedural programming.	Oop have access specifiers like private, public, protected, etc.
5.) Procedural programming does not have any proper way for hiding data so it is less secure.	Object oriented programming provides data hiding so it is more secure.
6.) In procedural programming, function is more important than data.	In object oriented programming, data is more important than function.
Ex:- C, Basic, Fortran, Pascal, etc.	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.

Q3. Define a class ParkingLot 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)*1.5;
        }
        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 ParkingLot();
        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 character arguements that replaces the character arguement ch1 with the character arguement ch2 in the given String s and prints the new string.

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

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

Example:
Input value of s1 = "COMMON WEALTH"
s2 = "GAMES"
Output: "COMMON WEALTH GAMES."

→

```
import java.io.*;
import java.util.Scanner;
class overloadfunc
{
```

```
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("First Index : "+FirstIndex);
    System.out.println("Last Index : "+LastIndex);
}
void Joystring(String s1,String s2)
{
    System.out.println(s1+" "+s2);
}
}
```

```
public class Overload
{
    public static void main(String args[])
    {
        overloadfunc df = new overloadfunc();
        olf.Joystring("Technalagy", 'a', 'o');
        olf.Joystring("Cloud computing means
                       internet based computing");
        olf.Joystring("Common Wealth", "Games");
    }
}
```