Day 1: Introduction to OOP and Java

Programming Approach - It means what will be the procedure of solving any problem. It could be of two types-
1. Procedural Approach- It means solving any problem step by step.
2. Object Oriented Approach

Object Orinted Programming System-
It is the based on real life concepts.The basis Concepts of OOPs are-
1. Class
2. Object

1. Class - It is a like a blueprint or prototype. A Class contains two things-
a) Attributes
b)Behavior
Attributes specifies the properties of the system while behavior specifies the functionality.
ex.

```
class human
{
   mind,hands,legs,eyes.........;  }
Attributes
   __________
   thinking()
   working()        => Behaviors
   walking()
   vision()
}
```

2. Object- These are instances of class,

They contains all the properties & methods whatever is defined inside the class. They are the active entities which directly interact with the external environment.

Class	Object
Car	Maruti-800,
Mobile	Nokia-X6, Samsung Galaxy
Pen	Raynold Jetter
Dog	Tommy

Features of OOPs-

1. Encapsulation & Abstraction
2. Polymorphism
3. Inheritance
4. Message Passing

Encapsulation - This is the concept of data hiding. It says represent only the essential information infront of the user.

Abstraction- This is concept of mechanism hiding.

or

to using any thing without knowing the background details of that thing is known as abstraction.

Polymorphism- This is concept of one to many. when an entity perform different operations on different occasions then we can say entity implements the polymorphism.

Inheritance- This is concept of reusability.

Message Passing - This is concept of

runtime communication of the objects.

Java
1. Java was developed by James Gosling in 1990 at sun macrosystem.
2. It is a case sensitive langauge.
3. The Extention of Java Program must be .java.
4. Java is a plateform independent language.
5. Java is technicaly known as JVM(java virtual machine)
   JVM= JDK (Java Developement Kit)+ JRE (Java Runtime Environment)
6. Java uses compiler for converting high level code into machine level code.
7. There are a lot of built in classes & methods available in java & remember the built in classes are always written in PascalCase Case & Methods are written in camelCase.
Example
   PascalCase =>InputStreamReader
   camelCase=>readLine()
8. The java program must be start with class keyword
9. Initially the name of program must be same as name of class.
10. for compiling a java program we use javac command & for ruuning a java program we use java command.
11. Before executing your program you must set the path-
   Like
   set path=C:\Program Files\Java\jdk1.7.0_04\bin

12. For compiling & running-
    javac programname.java
    java programname

Tokens- The smallest individual units of a program are known as tokens.It consist the following elements-
1. Identifiers
2. Constant
3. Keywords
4. DataType
5. Operators

1. Identifiers- The name of any programming element is known as identifiers. like name of variable, function, class etc.
2. Constant- They are the fixed values which did not change during execution of a program.
like
    10,20,30
    "ram","shyam"
3. Keywords- They are the reserve words which implements any particular meaning in your program. there are 50 keywords in java language.
like -
    int,for,if,while,do,public,private,class etc

4. Datatype- It specifies the type of a variable.
    Variable- They represent a memory location where we can put any value

according to their datatype.

Syntax

```
datatype variable_name;
```

Datatype

- char
- number
  - fixed point
    - int
    - long
  - floating point
    - float
    - double
- boolean

5. Opeartors- They are the symbols which perform any particular operation with their operands. They could be classified into following categories-

1. Assignment Operator(=)
2. Arithmetical Operator(+,-,*,/,%)
3. Relational Operator(<,<=,>,>=,==,!=)
4. Logical Operator(&&,||,!)
5. Increment/Decrement Operator(++,--)
6. Conditional Operator(?:)
7. Shorthand Operators(+=,-=,*=,/=)

Example- Program for adding two values-

```
class add
{
    public static void main(String args[])
    {
        int a,b,c;
        a=10;
        b=20;
        c=a+b;
```

```
        System.out.println("Sum : "+c);
    }
}
```

Que- WAP for swapping the value of two variables with each other.

Reading a Value in Java at RunTime- You can read a value in java by two methods. These are-

1. Through Command Line Argument
2. Through Built-in Classess like BufferedReader & Scanner Class

1. Throug Command Line Argument- Here we pass the value on command prompt means infront of program name like the following-

```
java programname value-1   value-2   n
       args[0]   args[1]   args[n-1]
```

Que- Read a name and print it with hello message.

```
class test
{
    public static void main(String args[])
    {
        String s=args[0];
        System.out.println("Hello : "+s);
    }
}
javac test.java
java test Utkarsh
```

Wrapper Classes - Java provides these

classes for converting one data type to another. The Classes & Methods are

Class Name	Method Name	Pupose
Integer	parseInt()	Convert String into Integer
	toString()	Convert Integer into String
Float	parseFloat()	Convert String into Float
Double	parseDouble()	Convert String into Double

```
class test
{
   public static void main(String args[])
   {
      int a,b,c;
      a=Integer.parseInt(args[0]);
      b=Integer.parseInt(args[1]);
      c=a+b;
      System.out.println("Sum : "+c);

   }
}
```

```
javac test.java
java test 10 20
```

Que- Read marks a student in P,C,M & calculate total & Percetnage.

Decision Making Statements- They are used for making the decision inside your program. The Decision Making Statements are-

1. if statement
2. switch statement

1. if statement- We can use different variations of if statement, These are-
   a. if else statement
   b. if else if ladder statement
   c. nested if statement

   a. if else statement- if we have a condition & their are two aspect of the condition, first for true and second for false, then we use such type of statement.

```
   Syntax
   if(condition)
   {
      true statement block
   }
   else
   {
      false statement block
   }
```

   Que- Check given number is even/odd.
   Que- Read age of a person & print whether he/she can vote or not?

   b. if else if ladder statement- if we have more than one conditions then we can use if else if ladder statement.

```
   Syntax
   if(condition-1)
      statement-1
   else if(condition-2)
      statement-2
```

   _____

```
    _________________
    else if(condition-n)
       statement-n
    else
       default statement
```
Que- Read 3 values & print the largest one.

Que- Read marks of a students in physics, chemistry & math & calculate and print their total , percentage & division.

c. netsed if statement - if, if statement contains one or more than one another if statement into its body then this term is known as nested if statement.

Syntax

```
    if(condition)
    {
       _________________
       _________________
       if(condition)
       {
          ---------------
          ----------------
       }
    }
```
Que- Read 3 values & print the largest one.

```
    class demo
    {
       public static void main(String args[])
       {
          int a,b,c;
          a=Integer.parseInt(args[0]);
          b=Integer.parseInt(args[1]);
```

```
            c=Integer.parseInt(args[2]);
            if(a>b)
            {
               if(a>c)

    System.out.println("Largest :"+a);
               else

    System.out.println("Largest :"+c);
            }
            else
            {
               if(b>c)

    System.out.println("Largest :"+b);
               else

    System.out.println("Largest :"+c);
            }
       }
    }
```

switch case statement- It is also a decision making statement.It works faster than if statement, but the disadvantages is that it did not support logical and relational operator.

Syntax

```
switch(expression)
{
   case value-1:
      statement-1;
      break;
   case value-2:
      statement-2;
      break;
```

```
        _____________________________
        _____________________________
        case value-n:
            statement-n;
            break;
        default:
            default statement
    }
```

Que- read day number and print day name.

Que- Read two values & then print the following menu-

1. Add
2. Subtract
3. Multiply
4. Divide

then read choice(1-4) & according to choice perform the operation.