Assignment-1

1. **Write a java program to find the area of rectangle**

Program:-

```
public class Area
{
 public static void main (String args[]){
  int l=Integer.parseInt(args[0]);
  int b=Integer.parseInt (args[1]);
  System.out.print ("Area of Rectangle is "+l*b);
  }
  }
```

Output:-

```
F:\full Stack>javac Area.java

F:\full Stack>java Area 4 5
Area of Rectangle is 20
```

2. **Write a java program to check the given no is Armstrong or not(153 is Armstrong no 1*1*1+5*5*5+3*3*3=153)**

Program:-

```
public class Armstrong{
public static void main(String args[]){
 int n=Integer.parseInt(args[0]);
 int s=0;
 int c=n;
 int r;
 while(c>0){
     r=c%10;
     s=s+(r*r*r);
     c=c/10;
     }
     if(s==n){
     System.out.println(" Number is Armstrong");
     }
     else{
     System.out.println(" Number is not Armstrong");
     }}
 }
```

Output:-

```
F:\full Stack>javac Armstrong.java
F:\full Stack>java Armstrong 153
 Number is Armstrong
```

3. Write a java program to check the given no is palindrome or not

Program:-

```
public class Pallindrome{
public static void main(String args[]){
 int n=Integer.parseInt(args[0]);
 int s=0;
 int c=n;
 int r;
 while(c>0){
     r=c%10;
     s=s*10+r;
     c=c/10;
     }
     if(s==n){
     System.out.println(" Number is Pallindrome");
     }
     else{
     System.out.println(" Number is not Pallindrome");
     }
 }
 }
```

Output:-

```
F:\full Stack>javac Pallindrome.java

F:\full Stack>java Pallindrome 151

Number is Pallindrome
```

4.Write a java program to generate first N prime numbers

Program:-

```
public class NPrime{
public static void main(String args[]){
 int n=Integer.parseInt(args[0]);
 int i=1;
 int c=0;
 while(c<n){
   //System.out.print(c);
     int j=1;
     int count=0;
     while(i>j){
               if(i%j==0)
                        count++;
               j++;
     }
     if(count==1){
```

```
                c++;
                System.out.print(i);
          }
        i++;

     }
     }
     }
```

Output:-

```
F:\full Stack>javac Nprime.java

F:\full Stack>java NPrime 5
235711
```

5. Write a java program to print even numbers in between given two numbers.

Program:-

```
public class GEven{
public static void main(String args[]){
int n1=Integer.parseInt(args[0]);
int n2=Integer.parseInt(args[1]);
int i;
for(i=n1+1;i<n2;i++){
  if(i%2==0){
  System.out.print(i);
  }
 }
}
}
```

Output:-

```
F:\full Stack>javac GEven.java
F:\full Stack>java GEven 3 9
468
```

1. What is Abstraction?

Ans:-Abstraction is a process of hiding the implementation details from the user, only the functionality will be provided to the user. In other words, the user will have the information on what the object does instead of how it does it.

In Java, abstraction is achieved using Abstract classes and interfaces.

Abstract class:-

A class which contains the abstract keyword in its declaration is known as

Abstract class

- Abstract classes may or may not contains abstract methods, i.e., methods without body (public void get () ;)
- But ,if a class has at least one abstract method, then the class must be declared abstract.
- If a class is declared abstract, it cannot be instantiated.
- To use an abstract class, you have to inherit it from another class, provide implementation to the abstract methods in it.
- If you inherit an abstract class, you have to provide implementations to all the abstract methods in it

Abstract Methods:-

If you want a class to contain a particular method but you want the actual implementation of that method to be determined by child classes, you can declare the method in the parent class as an abstract.

- abstract keyword is used to declare the method as abstract.
- You have to place the abstract keyword before the method name in the method declaration.
- An abstract method contains a method signature, but no method body.
- Instead of curly braces, an abstract method will have a semi colon (;) At the end.

2. What is Encapsulation?

Ans: - Encapsulation is one of the four fundamental OOP concepts. The other three are inheritance, polymorphism, and abstraction.

Encapsulation in Java is a mechanism of wrapping the data (variables) and code acting on the data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class. Therefore, it also known as data hiding.

To achieve encapsulation in Java-

- Declare the variables of a class as private.
- Provide public setter and getter methods to modify and view the variables values.

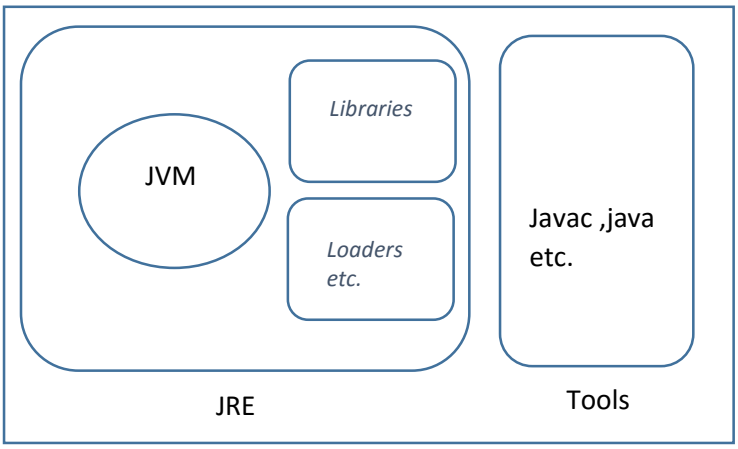
Benefits of Encapsulation:-

- The fields of a class can be made read-only or write-only.
- A class can have total control over what is stored in its fields.

3. What is JDK?

Ans: - Java Development Kit (JDK) is a software development kit that contains tools needed to develop Java applications, and JRE to run the program .We can think of JDK as a set of JRE and Development Tools.

When we download JDK, JRE is also downloaded, and we don't need to download it separately.



Jdk

4. What is JVM?

Ans: - Java Virtual Machine (JVM) is a specification to provide the runtime environment on which a byte code can be executed. JVMs are prepared platform specific and are available for almost all the hardware and machine.

Features of JVM:-

- It provides class loader to load a class.
- It provides bytecode verifier to verify the legality of the bytecode.
- It provides runtime.
- It executes the bytecode.

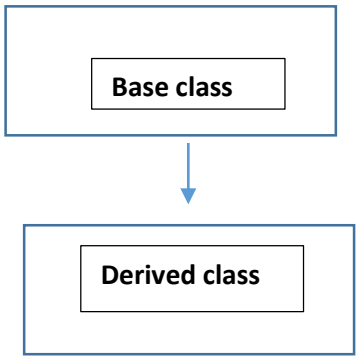
5. Define Inheritance?

Ans:-Inheritance can be defined as the process where one class acquires the properties (Methods and fields) of another. With the use of inheritance the information is made manageable in a hierarchical

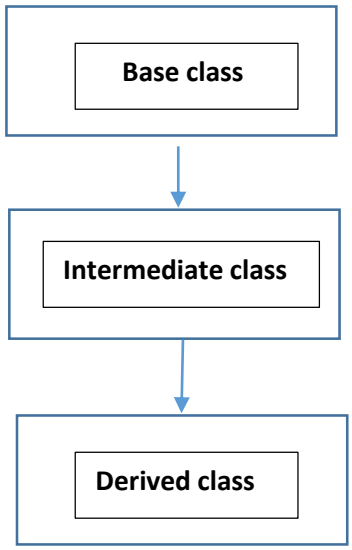
The class which inherits the properties of other is known as subclass (derived class, child class) whose properties are inherited is known as superclass (base class, parent class).extends is the keyword used to inherit the properties of a class.

Types of Inheritance

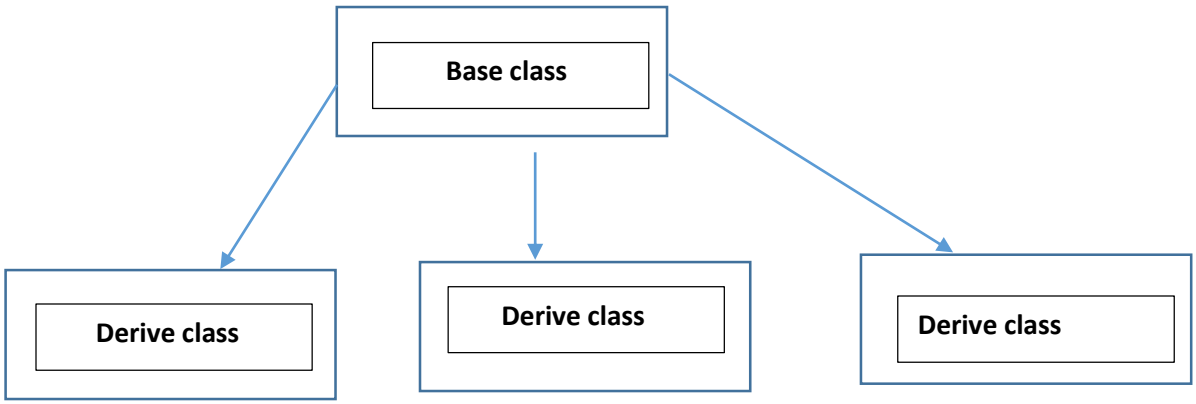
1. **Single Inheritance**: - In single inheritance, subclasses inherit the features of one superclass



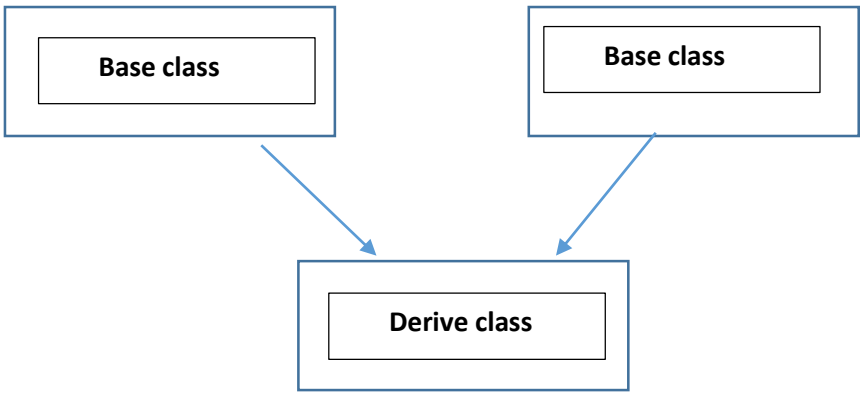
2. **Multilevel Inheritance:** In Multilevel Inheritance, a derived class will be inheriting a base class and as well as the derived class also act as the base class to other class



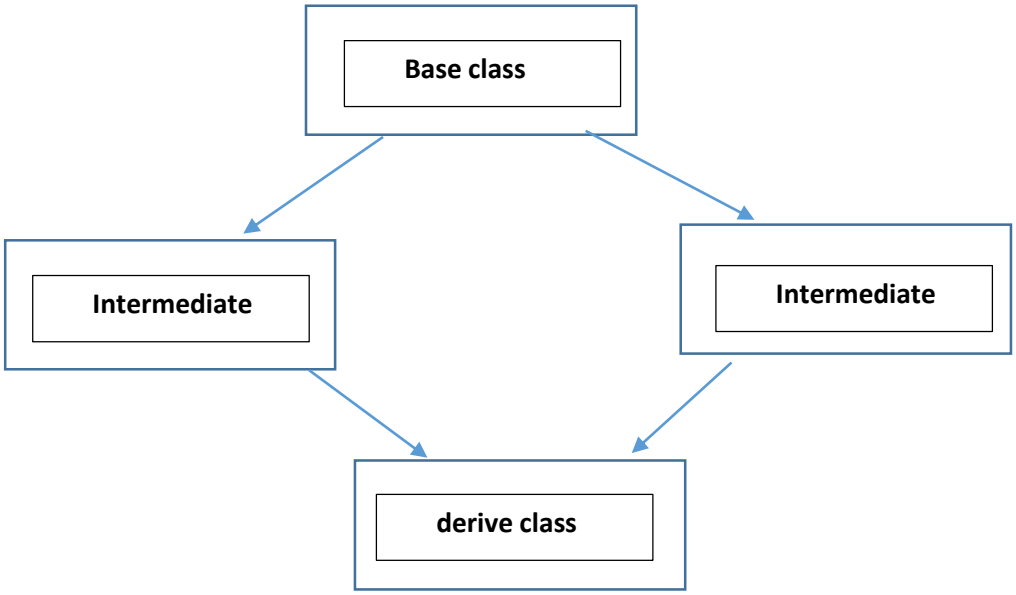
3. **Hierarchical Inheritance:** In Hierarchical Inheritance, one class serves as a superclass (base class) for more than one sub class



4. **Multiple Inheritance** (**Through Interfaces):-** In Multiple inheritance, one class can have more than one superclass and inherit features from all parent classes.
   Java does not support Multiple inheritance.



5. **Hybrid Inheritance (Through Interfaces):** It is a mix of two or more of the above types of inheritance. Since java doesn't support multiple inheritance with classes, the hybrid inheritance is also not possible with classes. In java, we can achieve hybrid inheritance only through Interfaces.



6. How java achieved platform independence?

Ans:-Platform independent language means once compiled you can execute the program on any platform (OS). Java is platform independent. Because the Java compiler converts the source code to bytecode, which is Intermediate Language. Bytecode can be executed on any platform (OS) using JVM (Java Virtual Machine).

7. Write the syntax of main function.

Ans:- Java main method is the entry point of any java program.

Its syntax is always public static void main (String [] args)

8. What is conditional operator?

Ans:-The ternary operator is also known as the conditional operator. This operator consists of three operands and is used to evaluate Boolean expressions. The goal of the operator is to decide, which value should be assigned to the variable.

The operator is written as:-variable x= (expression)? Value if true: Value if False

9. How many data types in java?

Ans:- There are two data types available in Java −

- Primitive Data Types
- Reference/Object Data Types

Primitive Data Types: - There are eight primitive data types supported by Java. Primitive data types are predefined by the language and named by a keyword. Let us now look into the eight primitive data types in detail.

1. Byte:-

- Byte data type is an 8-bit signed two's complement integer.
- Minimum value is -128 (-2^7)
- Default value is 0
- Maximum value is 127 (inclusive)(2^7 -1)
- Byte data type is used to save space in large arrays, mainly in place of integers, since a byte is four times smaller than an integer.
- **Example** − byte a = 100, byte b = -50

2. short:-

- Short data type is a 16-bit signed two's complement integer
- Minimum value is -32,768 (-2^15)
- Maximum value is 32,767 (inclusive) (2^15 -1)
- The short data type can also be used to save memory as byte data type. A short is 2 times smaller than an integer
- The default value is 0.
- **Example** − short s = 10000, short r = -20000

3.Int:-

- Int data type is a 32-bit signed two's complement integer.
- Minimum value is - 2,147,483,648 (-2^31)
- Maximum value is 2,147,483,647(inclusive) (2^31 -1)

- Integer is generally used as the default data type for integral values unless there is a concern about memory.
- The default value is 0
- **Example** − int a = 100000, int b = -200000

4. Long

- Long data type is a 64-bit signed two's complement integer
- Minimum value is -9,223,372,036,854,775,808(-2^63)
- Maximum value is 9,223,372,036,854,775,807 (inclusive)(2^63 -1)
- This type is used when a wider range than int is needed
- Default value is 0L
- **Example** − long a = 100000L, long b = -200000L

5. Float

- Float data type is a single-precision 32-bit IEEE 754 floating point
- Float is mainly used to save memory in large arrays of floating point numbers
- Default value is 0.0f
- Float data type is never used for precise values such as currency
- **Example** − float f1 = 234.5f

6. Double

- The double data type is a double-precision 64-bit IEEE 754 floating point
- This data type is generally used as the default data type for decimal values, generally the default choice
- Double data type should never be used for precise values such as currency
- Default value is 0.0d **Example** − double d1 = 123.4

7. Boolean

- Boolean data type represents one bit of information
- There are only two possible values: true and false
- This data type is used for simple flags that track true/false conditions
- Default value is false
- **Example** − boolean one = true

8. Char

- char data type is a single 16-bit Unicode character
- Minimum value is '\u0000' (or 0)
- Maximum value is '\uffff' (or 65,535 inclusive)
- Char data type is used to store any character
- **Example** − char letter A = 'A'

Reference Data types:-

- Reference variables are created using defined constructors of the classes. They are used to access objects. These variables are declared to be of a specific type that cannot be changed. For example, Employee, Puppy, etc.
- Class objects and various type of array variables come under reference data type.
- The default value of any reference variable is null.
- A reference variable can be used to refer any object of the declared type or any compatible type.
- **Example** − Animal animal = new Animal("giraffe");

10. What is constant? How it is declared?

Ans:-A **constant** is a variable whose value cannot change once it has been assigned. **Java** doesn't have built-in support for **constants**, but the variable modifiers static and final can be used to effectively create one. **Constants** can make your program more easily read and understood by other.