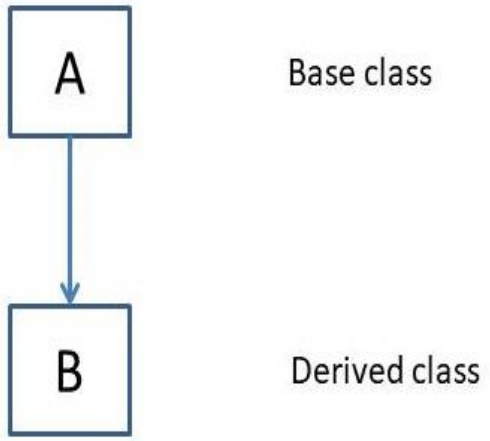
Training On Java

Lecture – 6
Concept Of Inheritance

Inheritance In Java

The Inheritance is a very important feature of object oriented programming. In Inheritance we can create a new class by using existing class. The existing class is called base/ super/ parent class and new created class is called derived/ sub/ child class. The concept of inheritance is also called ‘Reusability’.



Syntax Of Inheritance In Java

```
class A                    //Base Class
{
……….……;
……….……;
}
class B extends A //Derived Class
{
……….……;
……….……;
}
```

Types Of Inheritance In Java

There are following types of inheritance are supported in java:-

1. Single/ Simple Inheritance

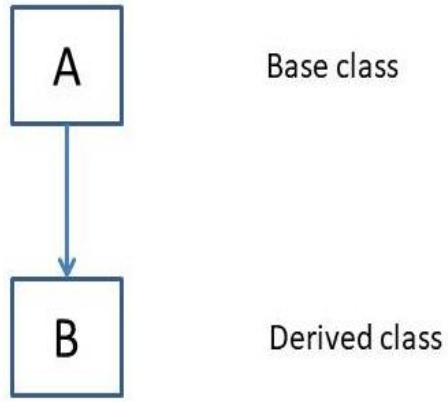
2. Hierarchical Inheritance

3. Multi – level Inheritance

4. Hybrid Inheritance

Single/ Simple Inheritance

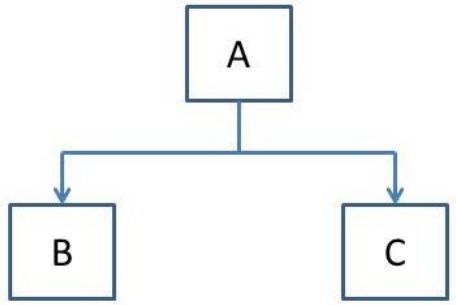
In Single Inheritance there is a single base class and single derived class.




```
class A
{
……;
……;
}
class B extends A
{
……;
……;
}
```

Hierarchical Inheritance

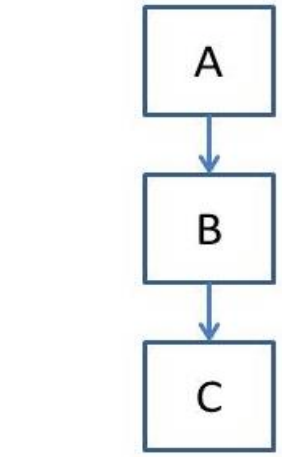
In Hierarchical Inheritance there is a single base class and multiple derived class.



Hierarchical Inheritance

```
class A
{
…….;
…….;
}
class B extends A
{
…….;
…….;
}
class C extends A
{
…….;
…….;
}
```

Multi-level Inheritance

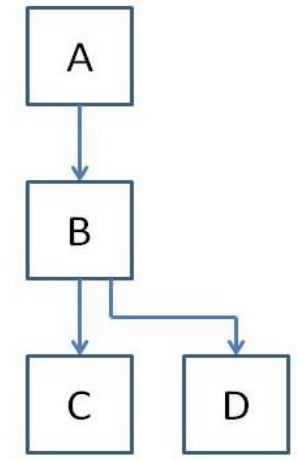


Multi-level Inheritance

```
class A
{
…….;
…….;
}
class B extends A
{
…….;
…….;
}
class C extends B
{
…….;
…….;
}
```

Hybrid Inheritance

If we combine more than one inheritance then resultant inheritance is called Hybrid Inheritance. Since Multiple Inheritance is not supported in java, so we can not include multiple inheritance to make Hybrid Inheritance.



Hybrid Inheritance

```
class A {
……;
}
class B extends A {
……;
}
class C extends B {
……;
}
class D extends B {
……;
}
```

Example Application - 1

/*Develop a program in java to create a class Rundog. In Rundog class make a method bark(), in bark() method display the rundog name and voice. By extending Rundog class create a new class named Bulldog. In Bulldog class make a method grawl(), in grawl() method display bulldog name and voice.*/

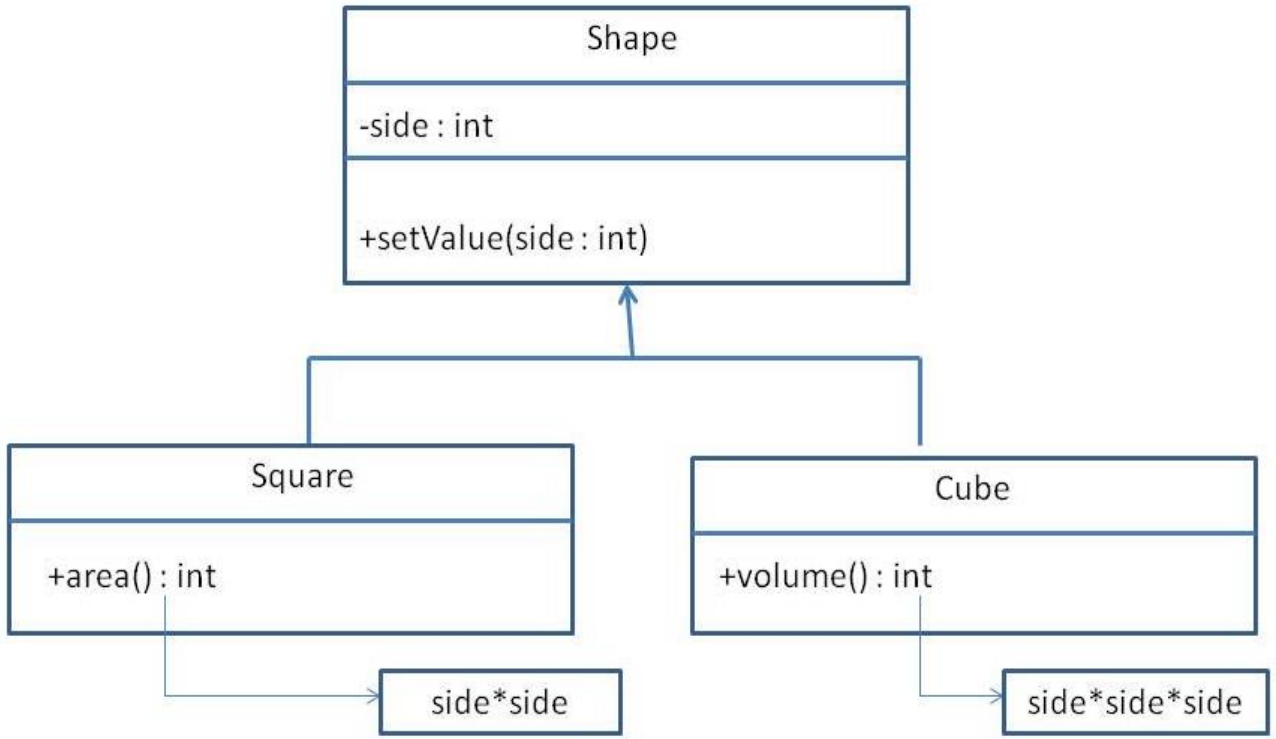
```
class Rundog
{
public void bark()
{
System.out.println("Tommy……");
System.out.println("Bho…….Bho……..");
}
}
class Bulldog extends Rundog
{
public void grawl()
{
System.out.println("Tuffy……..");
System.out.println("Gurr……Gurr……");
}
}
```

Example Application – 1 (cont..)

```
class Test
{
public static void main()
{
Bulldog dog=new Bulldog();
dog.bark();
dog.grawl();
}
}
```

Example Application – 2

Create the classes as following structure:-



Now test the classes..

Example Application – 2 (cont..)

```
import java.util.Scanner;
class Shape {
protected int s;        //protected data member
public void setValue(int x) //public method to initialize data member
{
s=x;
}
}
class Cube extends Shape {
public int volume() {
return (s*s*s);
}
}
class Square extends Shape {
public int area() {
return(s*s);
}
}
```

Example Application – 2 (cont..)

```
class Test
{
public static void main(String [] args)
{
Scanner sc=new Scanner(System.in);
int x;
System.out.print(“Enter side of cube : ”);
x=sc.nextInt();
Cube cu=new Cube();
cu.setValue(x);
System.out.println(“Volume of cube : ”+cu.volume());
System.out.print(“Enter side of square : ”);
x=sc.nextInt();
Square sq=new Square();
sq.setValue(x);
System.out.println(“Area of square : ”+sq.area());
}
}
```