PROGRAM – 1

```
class Player{
 String name;
 int age;
 Player(String n,int a){
    name=n;
    age=a;
 }
 void show()
 {
   System.out.println("Player name: "+name);
   System.out.println("Age: "+age);
 }
}
class cricket_player extends Player{
 String type;
 cricket_player(String n,String t,int a){
   super(n,a);
   type=t;
 }
 public void show()
 {
   super.show();
   System.out.println("Player type : "+type);
 }

}
```

```
class football_player extends Player{
    String type;
    football_player(String n,String t,int a)
 {
    super(n,a);
    type=t;
 }
  public void show()
  {
    super.show();
    System.out.println("Player type : "+type);
  }
}
class hockey_player extends Player{
  String type;
  hockey_player(String n,String t,int a)
  {
     super(n,a);
     type=t;
  }
  public void show()
  {
     super.show();
     System.out.println("Player type : "+type);
  }
```

```
}
public class Main{
  public static void main(String args[])
  {
    cricket_player c=new cricket_player("Rohit","Cricket",20);
    football_player f=new football_player("Rahul","Football",21);
    hockey_player h=new hockey_player("Aditya","Hockey",22);
    c.show();
    f.show();
    h.show();
  }
}
```

```
class Player{
  String name;
  int age;
  Player(String n,int a){
      name=n;
      age=a;

  }

  void show()
  {
     System.out.println("Player name: "+name);
     System.out.println("Age: "+age);
  }

}
```

input

```
Player name: Rohit
Age: 20
Player type : Cricket
Player name: Rahul
Age: 21
Player type : Football
Player name: Aditya
Age: 22
Player type : Hockey


...Program finished with exit code 0
Press ENTER to exit console.
```

PROGRAM - 2

```
class Rectangle{
   int length, breadth;
   Rectangle(int l, int b){
      length = l;
      breadth = b;
   }
   void area(){
      System.out.println("Area of Rectangle: "+(length*breadth));
   }
   void perimeter(){
      System.out.println("Perimeter of Rectangle: "+ (2*(length+breadth)));
   }
}
class Square extends Rectangle{
   int side;
   Square(int s, int l, int b){
      super(l, b);
      side = s;
   }
   public void area(){
      super.area();
      System.out.println("Area of Square: "+(side*side));
   }
   public void perimeter(){
      super.perimeter();
      System.out.println("Perimeter of Square: "+(4*side));
```

```
    }
}


public class Main
{
        public static void main(String[] args) {
                Square sq = new Square(8, 20, 15);
                sq.area();
                sq.perimeter();
        }
}
```

```
class Rectangle{
    int length, breadth;
    Rectangle(int l, int b){
        length = l;
        breadth = b;
    }

    void area(){
        System.out.println("Area of Rectangle: "+(length*breadth));
    }

    void perimeter(){
        System.out.println("Perimeter of Rectangle: "+ (2*(length+breadth)));
    }
}

class Square extends Rectangle{
    int side;
```

input

```
Area of Rectangle: 300
Area of Square: 64
Perimeter of Rectangle: 70
Perimeter of Square: 32


...Program finished with exit code 0
Press ENTER to exit console.
```

PROGRAM – 3

```
interface Printable  // creating Interface 1
{
   void print();
}
interface Showable  // creating Interface 2{
void show();
}
class Demo implements Printable,Showable{
  public void print()
  {
     System.out.println("Welcome");
  }
  public void show()
  {
   System.out.println("Interface Demo");
  }
}
public class Main{
   public static void main(String args[])
  {
     Demo obj = new Demo();
     obj.print();
     obj.show();
   }
}
```

PROGRAM – 4

```
interface Animal  // creating Interface 1
{
   int a=10; // implicitly static and final
   void eat(); //methods of interface are implicitly public
   void sleep();
}
class Demo implements Animal
{
  public void eat()
  {
    System.out.println("Animal Eats!");
  }

  public void sleep()
  {
```

```
    System.out.println("Animal Sleeps!");
    System.out.println("Static variable: "+a);
  }
}

public class Main{
   public static void main(String args[])
  {
    Demo obj = new Demo();
    obj.eat();
    obj.sleep();
  }
}
```

Main.java

```
interface Animal  // creating Interface 1
{
    int a=10; // implicitly static and final
    void eat(); //methods of interface are implicitly public
    void sleep();
}

class Demo implements Animal
{
   public void eat()
   {
      System.out.println("Animal Eats!");
   }

   public void sleep()
   {
      System.out.println("Animal Sleeps!");
      System.out.println("Static variable: "+a);
   }
```

input

```
Animal Eats!
Animal Sleeps!
Static variable: 10


...Program finished with exit code 0
Press ENTER to exit console.
```