

ממשקים והורשה

אם ממשק A יורש ממשק B ומחלקה B מממשת את ממשק B אז המחלקה B צריכה לממש את כל הפעולות המוגדרות בשני הממשקים.

אם מחלקה C יורשת ממחלקה B (מחלקה C לא התחייבה לממש את הממשקים) אז מחלקה C יכולה להשתמש בפעולות הממשק שמומשו במחלקה B ממנה היא ירשה.

```
public interface InterA
{
    public void printA(String st);
}

public interface InterB extends InterA
{
    public void printB(String st);
}

public class ClassA implements InterA
{
    public ClassA(String st)
    {
        System.out.println(st + " Object of ClassA");
    }

    public void printA(String st)
    {
        System.out.println(st + " PrintA from ClassA");
    }
}

public class ClassB implements InterB
{
    public ClassB(String st)
    {
        System.out.println(st + " Object of ClassB");
    }

    public void printA(String st) { System.out.println(st + " PrintA from ClassB"); }
    public void printB(String st) { System.out.println(st + " PrintB from ClassB"); }
}

public class ClassC extends ClassB
{
    public ClassC (String st)
    {
        super(st);
        System.out.println(st + " Object of ClassC");
    }
}
```

```
public class TestInheritance
{
    public static void main(String[] args)
    {
        ClassA ca = new ClassA("ca");    System.out.println();
        ClassB cb = new ClassB("cb");    System.out.println();
        ClassC cc = new ClassC("cc");    System.out.println();

        System.out.println();
        ca.printA("ca");
        cb.printA("cb");
        cb.printB("cb");
        cc.printA("cc");
        cc.printB("cc");
    }
}
```

```
ca Object of ClassA
cb Object of ClassB
cc Object of ClassB
cc Object of ClassC

ca PrintA from ClassA
cb PrintA from ClassB
cb PrintB from ClassB
cc PrintA from ClassB
cc PrintB from ClassB
```

כלומר - קיים יחס הורשה בין ממשקים. המשמעות: ממשק-A מגדיר את הפעולות שלו. ממשק-B מרחיב (יורש) את ממשק-A. ממשק-C יורש ממשק-B. בכל אחד מהממשקים כתובות רק כותרות הפעולות שהם מגדירים. מחלקה המממשת את ממשק אחד תצטרך לממש רק את הפעולות של ממשק זה. מחלקה המממשת את ממשק שמרחיב ממשק אחר תצטרך לממש את כל הפעולות של ממשק-העל ושל תת-הממשק.

```
public interface InterC extends InterB
{
    public void printC(String st);
}

public class ClassD implements InterC
{
    public void printC(String st)
    {
        System.out.println(st + " PrintC from ClassD");
    }
}
```

אבל - עלולה להיווצר פה בעיה: מחלקה-C (מחלקה עצמאית, ללא ירשה) המתחייבת לממש את ממשק-C, רואה בממשק-C רק את הפעולות שהוגדרו בממשק זה, איך היא אמורה לדעת מהן הפעולות בממשק-B ושל ממשק-A?

