

ערות על תרגול 10 בקורס תוכנה 1 – ליאור שפירא

הרחבת/צמצום נראות של שירותים

כפי שאמרתי בכיתה (וצדקתי לרווחתי), אסור למחלקה מרחיבה לדרוס שירות ולצמצם את הנראות שלו. קחו לדוגמה את המצב הבא:

```
public interface I {  
    public void foo();  
}  
  
public class A implements I {  
    public void foo() {  
        System.out.println("A.foo");  
    }  
}  
  
Public class B extends A {  
    ???????? void foo() {  
        System.out.println("B.foo");  
    }  
}
```

אם נגדיר את הנראות של B.foo להיות יותר מצומצמת משל A (למשל private) אזי B כבר לא מקיים את הממשק I. גם אם לא היה מוגדר הממשק I עדיין אסור לצמצם נראות, כיוון שניתן לכתוב:

```
A a = new B();  
  
a.foo();
```

החוזה שמגדיר הטיפוס A כולל שירות ציבורי בשם foo, אך הוא לא קיים במימוש B. לכן תהיה שגיאת קומפילציה כשנסה לקמפל את B

מחלקות פנימיות בתוך שירותים

1. מחלקה פנימית המוגדרת בתוך שירות יכולה לגשת לערכם של משתנים מקומיים שהם `final`. זה לא משנה אם המשתנה `static` או לא, העיקר שיהיה `final`.
2. מחלקה פנימית בתוך שירות היא בעלת גישה מלאה לכל השדות והשירותים של המחלקה העוטפת (כמו כל מחלקה פנימית). עם זאת, מחלקה זו תוכל לגשת לשדות ושירותי מופע של המחלקה העוטפת רק אם היא מוגדרת בתוך שירות מופע לדוגמא:

```
public class Outer {
    private int x = 5;

    public void foo() {
        class InFoo {
            public String toString() {
                return "" + x;
            }
        }

        System.out.println(new InFoo());
    }

    public static void main(String[] args) {
        Outer o = new Outer();
        o.foo();
    }
}
```