

```
public class C {

    float x = 6.25f;        // *** f nach 6.25 oder double statt float; immer double für Gleitkommazahlen
    double y;               // --- muss nicht initialisiert werden --> Default '= 0'
    private int i;           // *** kein ':' nach 'private' // --> default: 'public'

    C() {
        float f = -11;      // *** Deklaration hat fehlt
    }

    final void g(float x) {
        if (i != 0) {       // *** ohne '!=0' --> 'i' müsste sonst vom Typ boolean sein
            this.x = x;      // *** 'this.x' statt 'this->x'
        }
    }
    // for (int x = 0; x < 2; x++) {} // --- nicht erlaubt, da 'x' bereits verwendet wird in diesem Block

    public static void main(String args[]) {
        C c = new C();
        c.g(2);
        D d = new D(0);
        d.g("Test");
    }
}

class D extends C {

    D(int x) {               // *** keine Default-Parameter in Java
        super();            // *** super() muss erste Anweisung im Konstruktor sein
        C w[] = new C[2];   // *** [...] statt ();
        System.out.println(Integer.toString(x)); // *** toString existiert nicht für int
        // --> Wrapper-Klasse verwenden: Integer.toString(x);
        // System.out.println(x);                // = Alternative
    }

    void g(java.lang.String s) {
    //void g(float s) { // --- wäre nicht erlaubt, da überschreiben nur o.k., wenn nicht 'final' in Oberklasse

        C obj = new C();    // *** Initialisierung notwendig
        obj.y = java.lang.Math.sqrt(1.25); // *** nur 'sqrt' reicht nicht aus
    }
}
```