

# Feedback Comment Bank

Write the **code** on the paper, or paste the full comment into your LMS. Every comment ends with the fix.

---

## super (S)

Code	Comment
<b>S1</b>	super(...) must be the FIRST statement in the subclass constructor. Move it to the top.

## Override (O)

Code	Comment
<b>O1</b>	That's overloading (different parameters), not overriding. To override, match the superclass signature exactly.
<b>O2</b>	To reuse the parent's behavior inside your override, call super.method() — you reimplemented it instead.

## Polymorphism (P)

Code	Comment
<b>P1</b>	The ACTUAL (runtime) type decides which overridden method runs, not the declared type. Re-trace.

## Types (T)

Code	Comment
<b>T1</b>	You called a subclass-only method through a superclass reference — that won't compile without a cast.

## Positive (G)

Code	Comment
<b>G1</b>	Clean subclass — super() first, correct override signature. That's the inheritance FRQ pattern.
<b>G2</b>	You correctly reasoned that the actual type drives dynamic dispatch. Exactly the polymorphism concept tested.