

Suggestions from *Effective Java*

- ▶ 13: Minimize the accessibility of classes and members.
- ▶ 14: In public classes, use accessor methods, not public fields.
- ▶ 15: Minimize mutability.
- ▶ 16: Favor composition over inheritance.
- ▶ 17: Design and document for inheritance or else prohibit it.
- ▶ 18: Prefer interfaces to abstract classes
- ▶ 19: Use interfaces only to define types.
- ▶ 20: Prefer class hierarchies to tagged classes
- ▶ 21: Use function objects to represent strategies
- ▶ 22: Favor static member classes over nonstatic.

Joshua Bloch, *Effective Java*, Addison-Wesley, 2008. Pg 6 7–108