



OO Terms and Good Practice

hannes@ru.is

OO Terms

- Class
 - Attributes / Data
 - Behavior / Code
 - Encapsulation vs. Interface
 - Specifies how to create an Instance / Object
- Inheritance / Extension / „is-a“
 - child / derived class inherits properties of parent / base class

OO Terms cont.

- Multiple-Inheritance
 - Possible in some languages, but normally avoided
 - Danger: Tree → Graph („deadly diamond“)
 - Mix-In Class: Parent-less. Just for introducing features

OO Terms cont.

- Polymorphism

- Classes that encapsulate different things but share an interface
- „insulates code from the knowledge of *types of objects*“
- Makes heterogenous classes appear homogenous
- Implemented with virtual functions

OO Terms cont.

- Extensions
 - Inheritance („is-a“)
- Compositions
 - Objects are composed of other objects that all live/die together
- Aggregations
 - Objects can contain other objects that may have another life elsewhere

OO Terms cont.

- Design Patterns
 - Common solutions
 - Some general ones and some domain specific
- Examples
 - Singleton
 - Iterator
 - Abstract Factory

Coding Standards

- Seek to maintain...
 - „slick“ interfaces
 - „good“ names
 - „clear“ namespaces
 - „absolute“ consistency
- And „reveal“ the errors
 - E.g. The original Hungarian notation