

PROGRAMMING IABSTRACT CLASSES and INTERFACES

Designing for Polymorphism

Logical thinking needed to ensure that there is no redundant data in the sub-classes and nothing in the super-classes that does not apply to all ~~first~~ subclasses

Abstract class may/may not contain abstract methods.

Any class that contains abstract methods must be abstract (not concrete).

Various issues trying to fit "play" into the hierarchy - perhaps best solution is to create a new superclass.

Java provides 'Interfaces' with behaviour similar to 100% abstract classes. - can be used anywhere we might use a class.

A class can extend one other class but can implement many interfaces.

See Powerpoint slides for the lecture notes - there are many complex slides moving too quickly to make reliable notes!