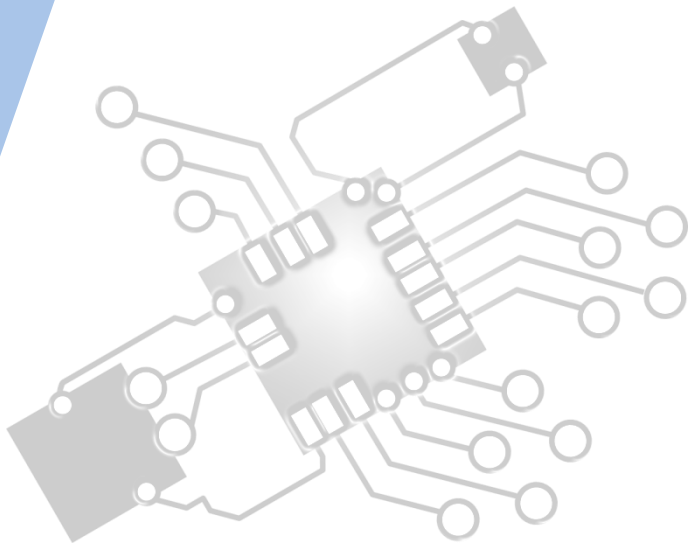
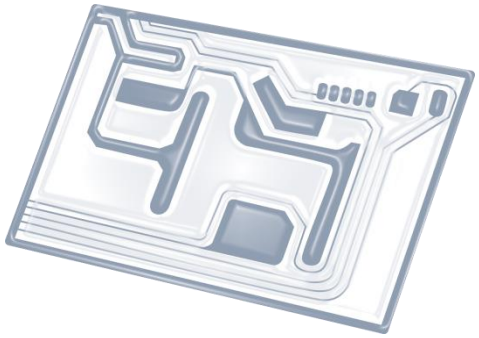
Objects as a programming concept

IB Computer Science



Content developed by
Dartford Grammar School
Computer Science Department

HL Topics 1-7, D1-4



1: System design

2: Computer Organisation

3: Networks

4: Computational thinking

5: Abstract data structures

6: Resource management

7: Control

D: OOP

HL & SL D.2 Overview

D.2 Features of OOP

D.2.1 Define the term encapsulation

D.2.2 Define the term inheritance

D.2.3 Define the term polymorphism

D.2.4 Explain the advantages of encapsulation

D.2.5 Explain the advantages of inheritance

D.2.6 Explain the advantages of polymorphism

D.2.7 Describe the advantages of libraries of objects

D.2.8 Describe the disadvantages of OOP

D.2.9 Discuss the use of programming teams

D.2.10 Explain the advantages of modularity in program development

1: System design

2: Computer Organisation

3: Networks

4: Computational thinking

5: Abstract data structures

6: Resource management

7: Control

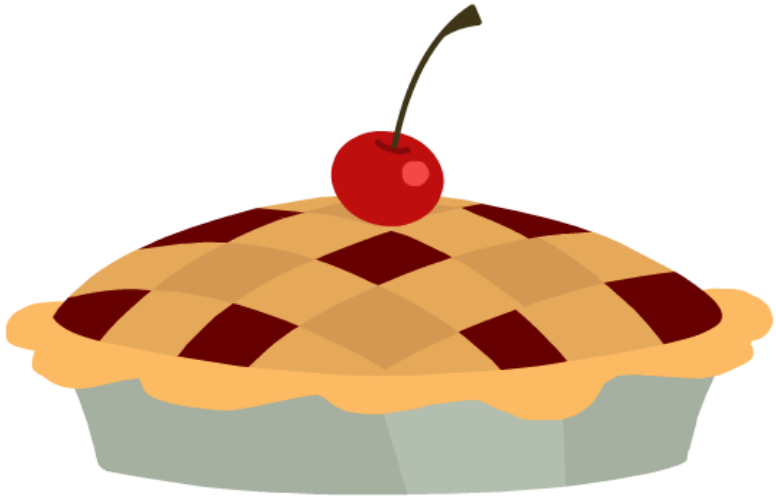
D: OOP

Topic D.2.5

Explain the advantages of **inheritance**

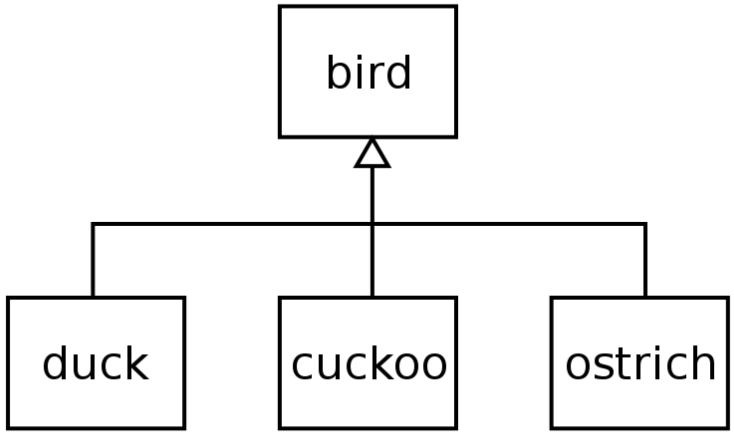
Four OOP fundamentals:

- **A**bstraction
- **P**olymorphism
- **I**nheritance
- **E**ncapsulation



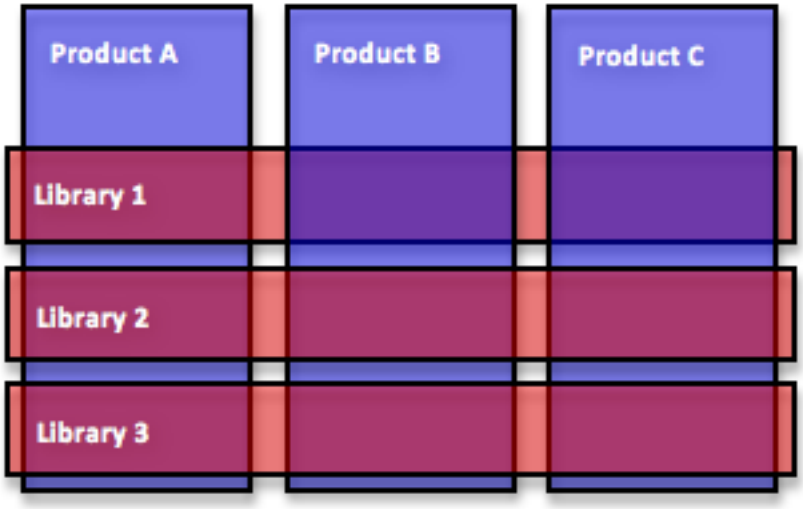
Definition: Inheritance

- Process whereby one object acquires the **properties** (states and behaviours) of another
- The most commonly used keyword would be **extends** and **implements**



Key benefit

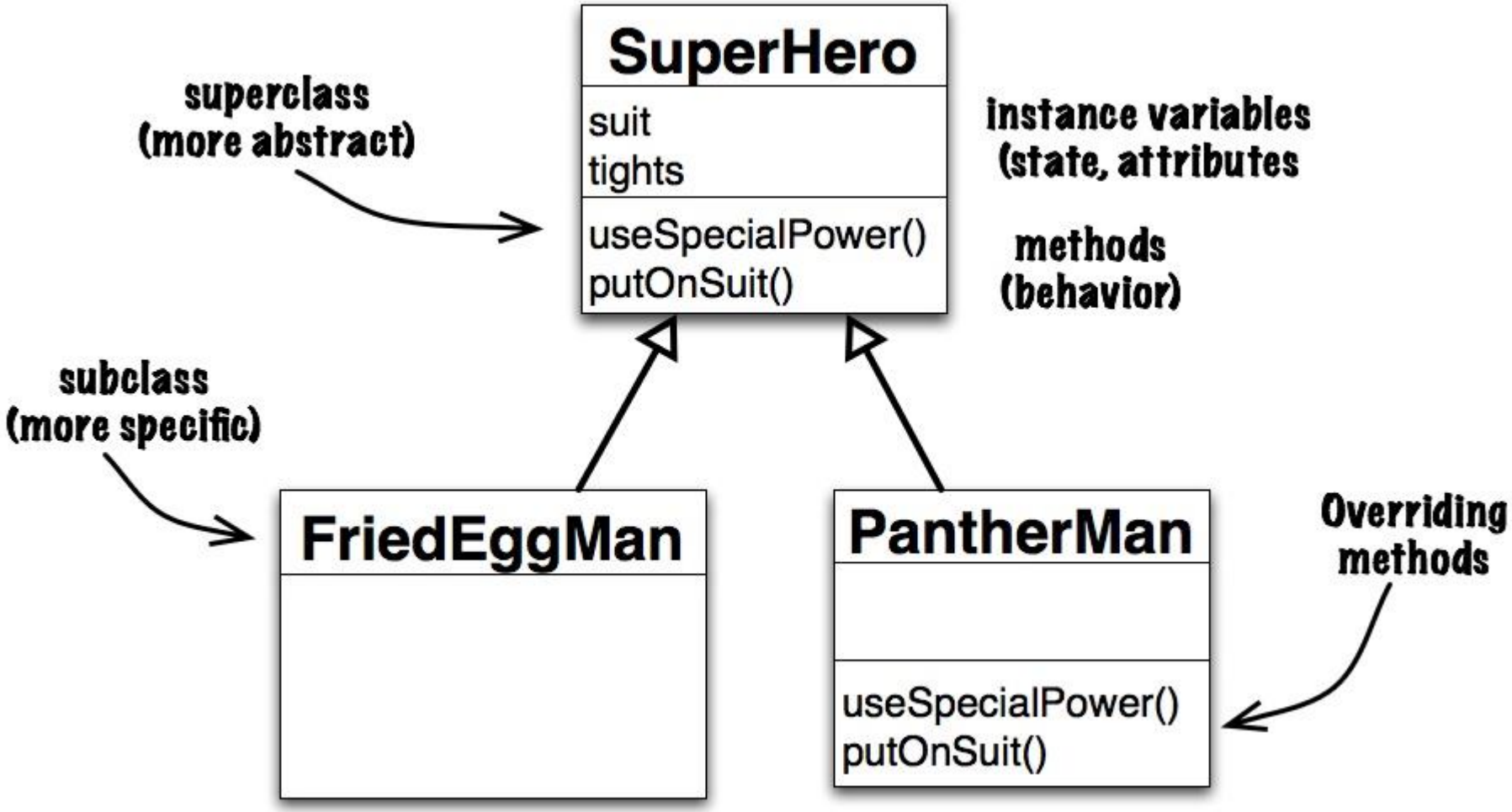
Minimizing the amount of **duplicate code** in an application by **sharing common code** amongst several subclasses.



Other advantages

- Where equivalent code exists in two related classes, the hierarchy can usually be refactored to **move the common code up to a mutual superclass**.
- This also tends to result in a **better organization** of code and **smaller, simpler compilation units**.
- Inheritance can also make application code **more flexible** to change because classes that inherit from a common superclass can be used interchangeably.

Making it modular



Additional advantages

- **Reusability** -- facility to use public methods of base class without rewriting the same
- **Extensibility** -- extending the base class logic as per business logic of the derived class
- **Data hiding** -- base class can decide to keep some data private so that it cannot be altered by the derived class
- **Overriding-**--With inheritance, we will be able to override the methods of the base class so that meaningful implementation of the base class method can be designed in the derived class.