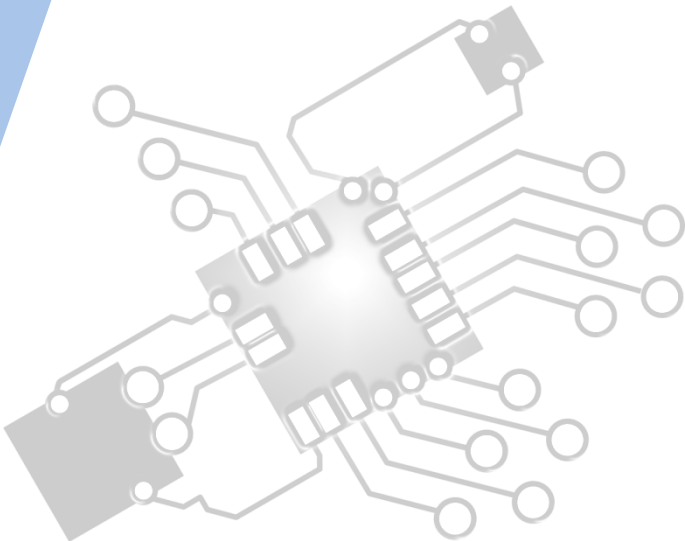




Features of OOP

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

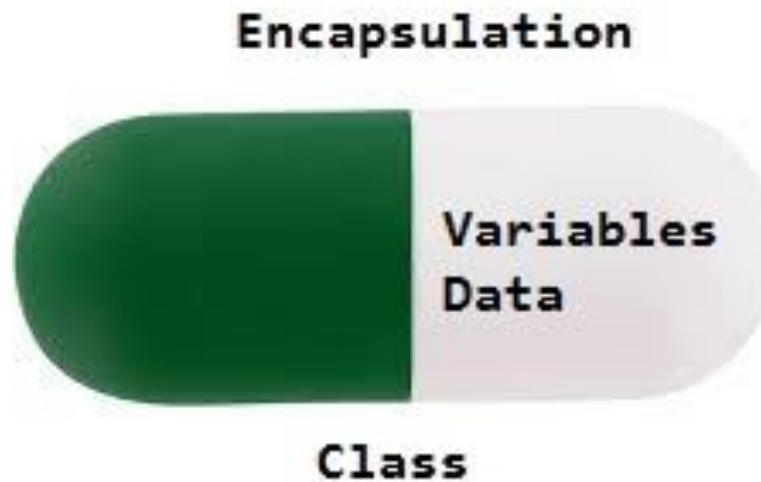
D: OOP





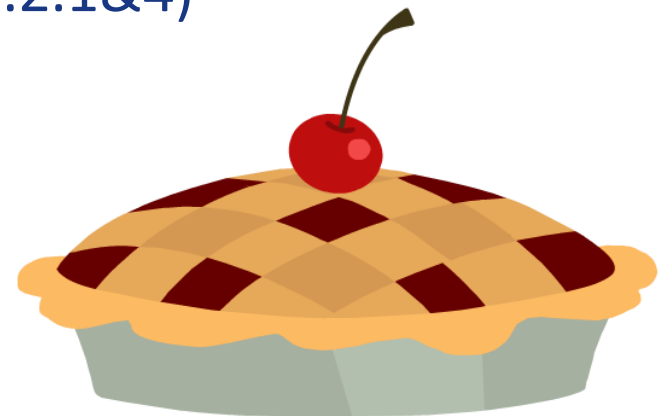
Topic D.2.4

Explain the **advantages** of encapsulation



Four **OOP** fundamentals:

- **A**bstraction (See Topic 4.1.17-20)
- **P**olymorphism (See Topic D.2.3&6)
- **I**nheritance (See Topic D.2.2&5)
- **E**ncapsulation (See Topic D.2.1&4)

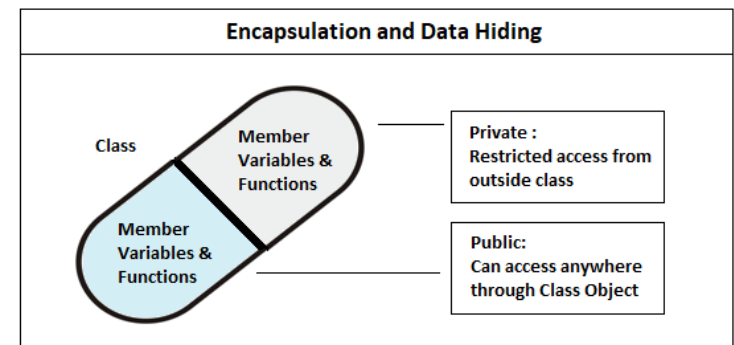


Advantages of Encapsulation

- A. Data Hiding
- B. Increased Flexibility
- C. Reusability
- D. Testing code is easy

A. Data Hiding

- The user will have no idea about the inner implementation of the class (implies security).
- It will not be visible to the user that how the class is storing values in the variables.
- He/she only knows that we are passing the values to a setter method and variables are getting initialized with that value.



B. Reusability

- Encapsulation also improves the reusability of code (write once/use many times).
- Methods can be copied to different/new classes and help meet new requirements.



C. Testing code is easy

- Encapsulated code is easy to test with unit testing (a type of automated testing that tests many different types of data quickly).
- It is easier to fix larger programs if you know which method is returning the wrong response.



D. Increased Flexibility

- We can make the variables of the class as read-only or write-only depending on our requirement.
- If we wish to make the variables as read-only then we can omit the setter methods
- Or if we wish to make the variables as write-only then we have to omit the get methods



Exam note!

It is important to keep in mind that the scope of data should be **confined to the object** in which it is defined as far as possible in order to **limit side effects** and dependencies.

WARNING

