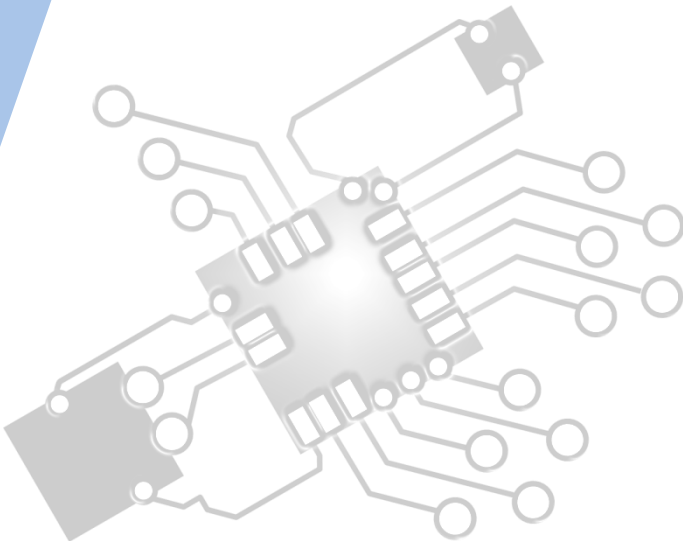




Features of OOP

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

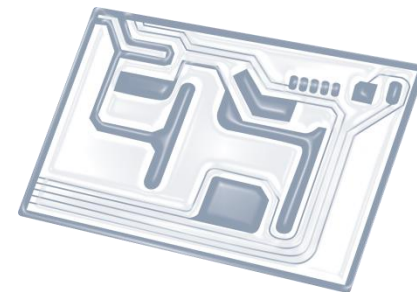


7: Control

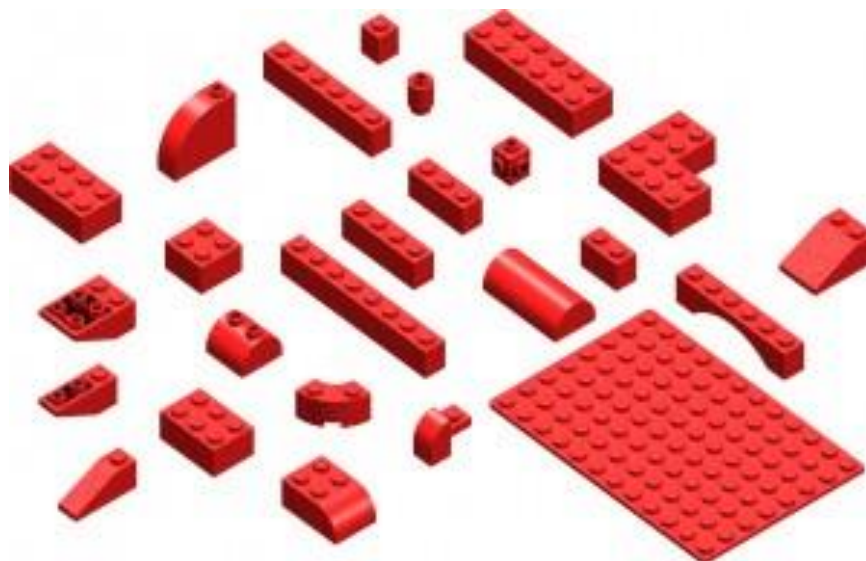
D: OOP



Topic D.2.10

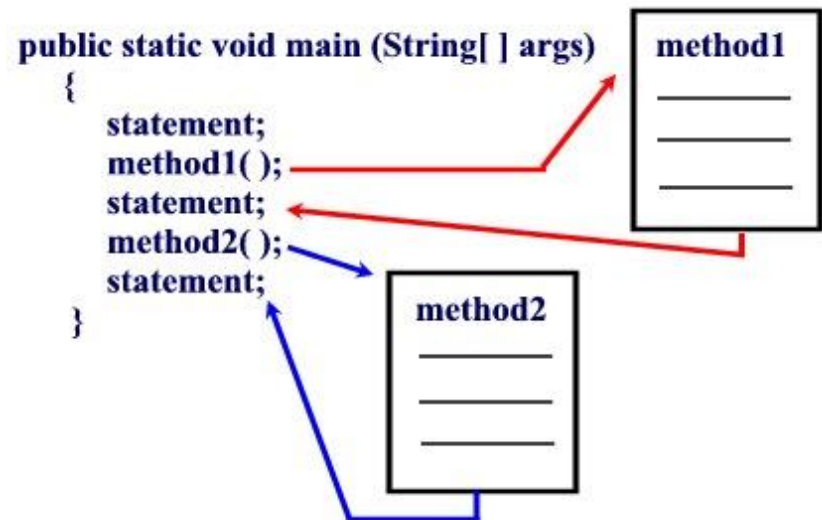


Explain the **advantages** of **modularity** in program development



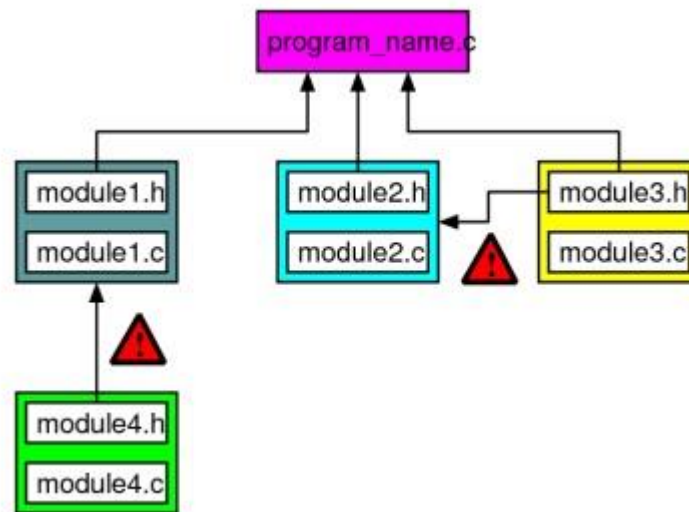
Advantages

- A. Easier **debugging** and **testing**
- B. **Speedier** completion
- C. Code blocks **reusable**



A. Easier debugging and testing

- By having smaller modules to test, it is easier to find and fix bugs
- The number of tests that have to be run to confirm that module is fully operational is also drastically reduced according to the number of functions in a particular module



B. Speedier completion

- By breaking the project into smaller modules, you could save time by finding modules that already exist in libraries that do the function you are looking for.
- By splitting the task into different modules, each module can be worked on concurrently (except for particular circumstances) thus leading to a speedier completion of the overall project

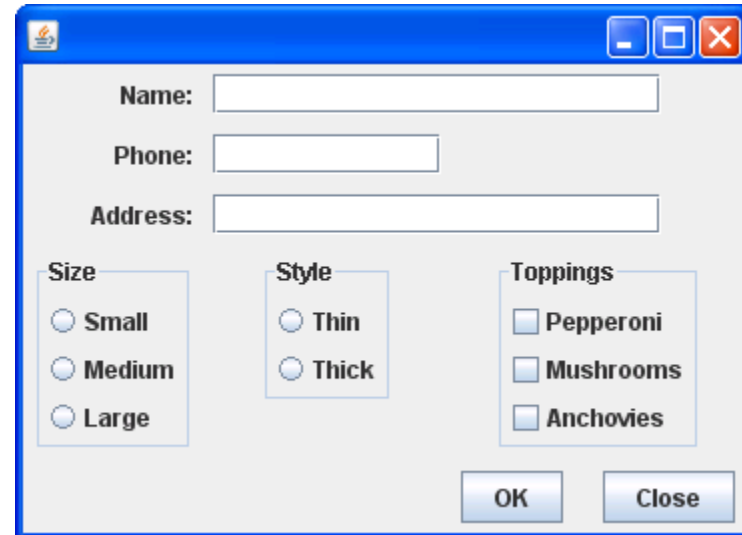
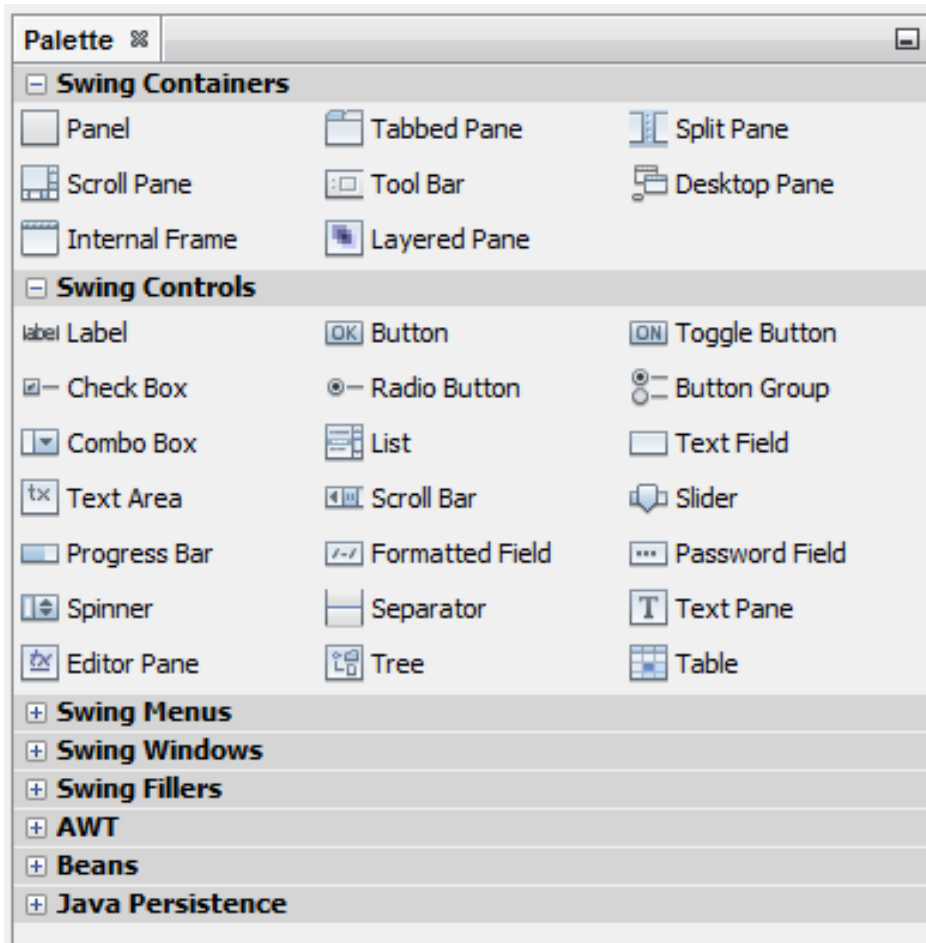


C. Reusable code blocks

- Some problems/functions are very common and possibly occur in multiple different programs (for example the need to make a text box or a clickable button to start a task)
- By reusing blocks of code, the development time is slashed and the project is completed sooner.



Classic example: Javax Swing



```
//Make sure we have nice window decorations.
JFrame.setDefaultLookAndFeelDecorated(true);

//Create and set up the window.
JFrame frame = new JFrame("HelloWorldSwing");
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

//Add the ubiquitous "Hello World" label.
JLabel label = new JLabel("Hello World");
frame.getContentPane().add(label);

//Display the window.
frame.pack();
frame.setVisible(true);
```