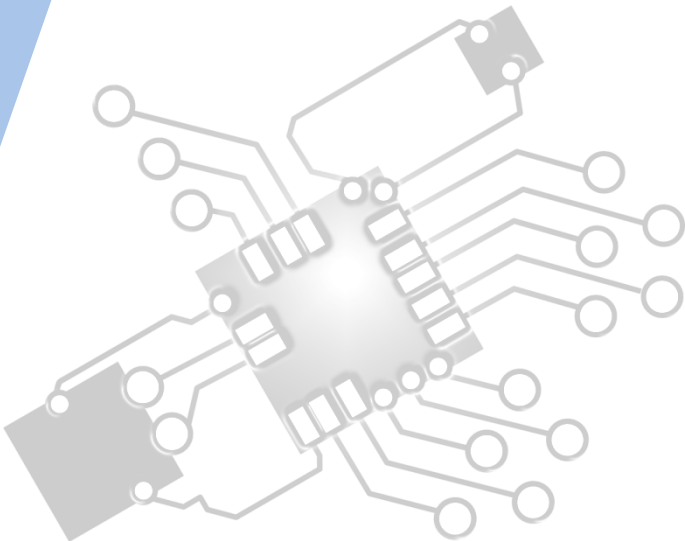




Objects as a programming concept

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.1 Overview

D.1 Objects as a programming concept

D.1.1 Outline the general nature of an object

D.1.2 Distinguish between an object (definition, template or class) and instantiation

D.1.3 Construct unified modelling language (UML) diagrams to represent object designs

D.1.4 Interpret UML diagrams

D.1.5 Describe the process of decomposition into several related objects

D.1.6 Describe the relationships between objects for a given problem

D.1.7 Outline the need to reduce dependencies between objects in a given problem

D.1.8 Construct related objects for a given problem

D.1.9 Explain the need for different data types to represent data items

D.1.10 Describe how data items can be passed to and from actions as parameters



1: System design

2: Computer Organisation



3: Networks

4: Computational thinking



5: Abstract data structures

6: Resource management



7: Control

D: OOP





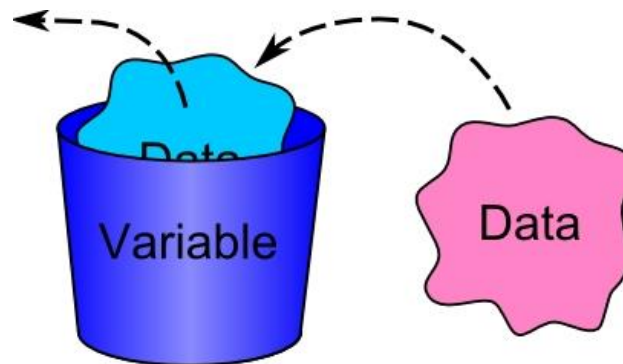
Topic D.1.9

Explain the need for **different data types** to represent data items



Four key data types

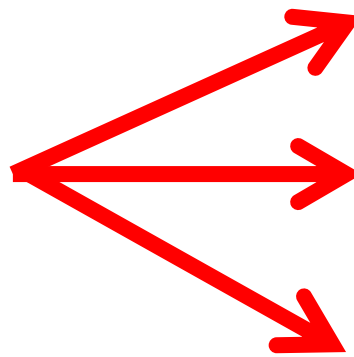
- **integer** (Java: **int**), e.g. **3**, **-4**, **999**, **23**
- **real** (Java: **double**), e.g. **-3.1415**, **9.999**
- **String** (Java: **String**) e.g. **“strange”**
- **Boolean** (Java: **boolean**) e.g. **true** / **false**



Why do we need different types?

- ✓ Data is stored as a combination of **binary values** in the computer.
- ✓ Data types are used to **store different kinds** of data.
- ✓ They are needed because they specify to the computer **how to interpret** the binary values in the storage.

10100011
00010101



Double? **143.21**?

Integers? **143** & **21**?

String? **"£2"**?

Is one type better than another?

- Each data type (in Java) takes up a different amount of RAM:
 - **Boolean** = **1 byte**
 - **Integer** = **4 bytes**
 - **Real** = **8 bytes**
 - **String** = ... a lot!
 - multiply the number of characters of the String by two;
 - add 38;
 - if the result is not a multiple of 8, round up to the next multiple of 8;
 - E.g. **"I love cheese"** = 12 character x 2 = 24 + 38 = 62 → **64 bytes**

The **less RAM** a program uses, the **more efficient** it is likely to be.

Efficient use of:

power



memory



processing

