An Introduction to **Object-Orientation**

What do we mean by an 'Object'?

- An entity that encapsulates both data and behaviour

Programming paradigms

- Pre-procedural days
  - Single procedure
  - GOTO command
  - Spaghetti Code

```
1 i=0
2 i=i+1
3 PRINT i; "squared=";i*i
4 IF i>=100 THEN GOTO 6
5 GOTO 2
6 PRINT "Program Completed."
7 END
```

Procedural Programming Paradigm

- The C programming language
  - Designed by Dennis Ritchie in 1972
  - CLion IDE supports C
  - Modular
  - Global or local variables

```c
#include <stdio.h>

int Square(int n)
{
    return n * n;
}

int main()
{
    int num = 6;
    num = Square(num);
    printf("%d\n", num);
    return 0;
}
```

Object-Oriented Programming Paradigm

- The C++ programming language
  - Designed by Bjarne Stroustrup in 1985
  - "C with Classes"
- Data can be associated with a class.
  - Known as **attributes** or **member variables**
- Behaviour can be associated with a class.
  - Implementations are called **methods** or **member functions**
- A **class** is a template or blueprint for creating **objects**
  - An object is a concrete instance of the class type.
  - We use the term **instantiating** a class to create an object.
- Classes have a specific method called a **constructor** that will set the initial state of an object.

Attributes, States and Behaviours

- 5 Minutes in pairs to identify possible attributes of the following objects.
    - Dog
    - Hotel booking
    - Circle
- 5 Minutes in pairs to identify possible methods of the following objects.
    - Hotel booking
    - Circle

Abstraction and Encapsulation

- Objects are representations of things that exist in the real world that we wish to model in a computer system.
- Offers us a new level of abstraction.
- Objects do not share data.
- Encapsulation binds together the attributes and the methods that manipulate the data.
- Data hiding is a feature of object-orientation, an objects data should be private. Access is controlled through public getter and setter methods.