

Term 2

Lesson 3:

Classes - The Basics

Classes The Basics

Object

Has variables and methods

Variables: the state (condition) of the object

Methods: behavior (actions) of the object

Classes The Basics

Class

Template for an object

The **data type**

Classes The Basics

Analogy

A blueprint for a house describes what a house can be, but is not an actual house.

One blueprint can be used to make many houses.

Classes The Basics

Basic Idea:

An **object** is an instance of a **class**.

```
Book b = new Book ("Harry Potter",  
                  "JK Rowling", 1997);
```

Classes The Basics

Instantiation

Create a variable in memory

Ex:

```
Book b = new Book ("Harry Potter",  
                  "JK Rowling", 1997);
```

This creates a reference to a Book object.

The reference is stored in b.

Classes The Basics

Constructor

Method in a class that builds class in memory

Sets values in variables

Has same name as class

Can have more than one
(we'll go over this later)

Classes The Basics

How it happens:

Constructors build objects

```
Book b = new Book ("Harry Potter",  
                  "JK Rowling", 1997);
```

new: returns a reference to a newly created object

Calls the **constructor** method in the class

Classes The Basics

Now you try it:

Write a class to hold a coin

Have an int variable to store whether it is on heads or tails.

Classes The Basics

Public and Private

Public – means it can be used by a program outside this file

Private - means it can only be used by this class

ALL variables are private

We'll talk about why we do this later

Classes The Basics

Get and Set

These are called your **accessors** and **mutators**

We'll talk more about why we do this in a few lessons

For now - it controls how the user can get to your values