**Intro to OOP (Worksheet 1)**

Classes & Objects

A **class** is a template or blueprint from which objects will be instantiated. It's like the mould from which objects can be created/instantiated from.

A class consists of a collection of states (**attributes** or **properties**) and behaviours (**methods**).

An **object** is an instance of a class. The **attributes** of an object are stored using variables and the behaviour/methods of an object are implemented as functions or procedures and can be used to perform operations on the data.

Most classes have one method called a **constructor**. It is a function that is automatically called when an object is instantiated from a class. It is mainly used to initialise the default values for each **attribute** of the object.

Superhero Class

Superhero
name
superpower
strength
walk()
run()
jump()
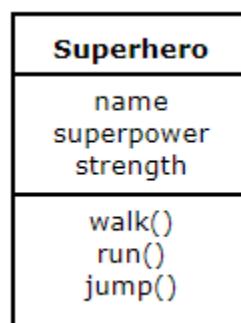
Let's consider a class called *Superhero*.

The attributes of a superhero are:
- *name* (String)
- *superpower* (String)
- *strength* (Integer)

Some of the methods of a superhero are:
- *Superhero()* – This is the constructor which may take some parameters used to initialise the default value of the superhero to be created.
- *walk()*
- *run()*
- *jump()*

The diagram above represents the *Superhero* Class with its attributes and its methods.

Create the Superhero class in C#.

1. Create a new Console project (.NET framework) in Visual Studio called *VideoGame*

2. Click *Project, Add Class* to add a new class called *Superhero*

3. Add the three attributes listed above to the class – remember that these should be **private** so that they cannot be accessed directly by other parts of the program.

4. Add a method called *Superhero* which accepts one argument (a name). This will be the **constructor** for this class. Use the argument to set the value of the name attribute

5. Add a method called *setPower* which accepts one argument (a superpower) and returns no values. In this method, set the value of the *superpower* attribute to the argument that was passed in*.*

6. Add a method called *getPower* which accepts no arguments and **returns** a string which is the *superpower* attribute.

7. Repeat steps 5 & 6 for strength

8. Create a *getName* method which returns the superhero name

9. Save your class and return to the *Program.cs* window

A video game might have 3 superheroes called Spiderman, Superman and Batman, so we will instantiate 3 objects from our *Superhero* Class.

"Spiderman",   "Wrist web-shooters to shoot spider web material", 10

"Batman",   "Night vision",   10

"Superman",   "Can fly",   20

> **Naming Convention**
>
> When coding, the identifier of a class starts with an uppercase letter, e.g. *Superhero*.
> The identifier of an object starts with a lowercase letter, e.g. *spiderman*.

10. In the *Main* procedure, add code which creates a **new** Superhero object called spiderman. Make sure that you pass in data for the appropriate parameter(s).

11. Using the *set* methods, set spiderman's superpower and strength

12. Repeat for the other two superheroes

13. Use the *get* methods to write out to the console output which looks like this:

    Spiderman's strength is 10 and his superpower is that he has wrist web-shooters to shoot spider web material

14. Repeat for the other two superheroes.
15. Use the *get* methods to tell the user which superhero is the strongest

Extension

- Create a collection(5 element array) of Superheros called *Supers*

- Add your existing superheros (Spiderman, Batman and superman) to the array and add two more of your own choice(5 superheros in total)

- Use a for loop to produce console output for all superheros which looks like this:

  Spiderman's strength is 10 and his superpower is that he has wrist web-shooters to shoot spider web material

- Sort the array and print the superheros in order of their strength

- Change the size of the array to 7 and ask the user to tell you the details for another 2 superheros – make sure that you validate their input

- Create a menu screen for the user which allows them to choose to add or delete a superhero - shouldn't allow them to have more than 7 in total.

- They should also be able to choose to see the powers of an individual superhero - maybe they type in a name and you display their details