# Glossary of Object-Oriented terms

|  |  |  |
| --- | --- | --- |
| **Item** | **Term** | **Definition** |
| 1 | Class | A user-defined data type that has specific data and includes the methods/functions that operate on that data |
| 2 | Object | An instance of a class which has been created, usually by calling its ‘constructor’ |
| 3 | Encapsulation | The term applied to enclosing the data and functions of a class within that class definition in programming |
| 4 | Information Hiding | Another name for ‘encapsulation’ |
| 5 | Field | A variable that is part of a class; an ‘attribute’ of that ‘thing’ |
| 6 | Method/Function | A piece of code that can operate on a field in an object of a class once that object has been created |
| 7 | Constructor | A special method in a class that creates instances of a class |
| 8 | Inheritance | A mechanism that passes fields and methods from one class to another |
| 9 | “is-a” | A reference to the “direction” of inheritance |
| 10 | Sub-class/subclass | A class which has inherited the data and methods from another class |
| 11 | Superclass | A class which provides data and methods to its subclasses |
| 12 | Visibility | Term applied to the ability of a field or method in a class to be ‘seen’ by another class in the program |
| 13 | Instance | An instantiation of a class; another word for ‘object’ |
| 14 | Instance variable | A variable which ‘belongs’ to each object of a class after it has been instantiated |
| 15 | Class variable | A variable which ‘belongs’ to **the class**, NOT to the objects of that class |
| 16 | Singleton | A class which has only one instance in a program. Ever. |
| 17 | Attribute | Parts of a class that are captured as ‘fields’ or ‘instance variables’ |