Task 1

**Encapsulation:**  This is the process of combining data and functions into a single class. The data is not accessed directly but is accessed through functions that present inside the specific class. Meaning that attributes of the class are kept private and public getters and setters method are provided in order to manipulate these attributes. For example in a real life scenario we can use a car. A car can start, accelerate, break and reverse. In order to perform these actions we need a mechanism. An ignition to start the car and stop the car. The gearbox, accelerator and the breaks in order to decrease/increase the speed of the car.

**Abstraction:** This is the process of where we only show the relevance of the data and ‘hide’ the unnecessary information of the object from the user. For example in a mobile phone we can press buttons such as the home button or lock screen and we can see the action being performed. However how the actions are performed like the process behind locking it and how we are straight away directed to the home screen when we press the home screen is all ‘hidden’ from the user. This is done to reduce complexity of the object but increase its efficiency.

**Polymorphism:** In programming this is the ability to present the same interface for different data types. For example if we use the class “Shape”. Other classes can inherit from the “shape” class such as the classes like Square, circle and polygon etc. With polymorphism, each of those classes will have different underlying data. The circle requires a centre and radius whereas the square requires 2 co-ordinates from the top left and bottom right corners.

**Inheritance:** This is the mechanism in which one object can acquire all the properties and behaviour of the parent object. We do this in order to create new classes that are built upon existing classes. This is used for method overriding and code reusability. In inheritance when we inherit from an existing class we can reuse the methods and fields of the parent class and we can also add new methods and fields in the class. In programming the syntax for java inheritance is “extends” this, indicates that we are making a new class that derives from an existing class. The terminology of Java for inheritance is Parent/Super class and Child/Subclass. For example we have Animals in this world. In respective of what I just said we would say Animals is the parent class, which has the function of eating. The class Dog will extend Animals. The class dog will have its own fields and function such as barking. But seeing as the dog is the child class of Animals, the dog will inherit the function eating from the parent class Animals. So the output will be that the dog will be able to bark and eat. We can create multilevel of inheritance. So in order to extend this example we can have Pup, which is the sub class of Dog and a pub can weep. So the pup will be able to bark, eat and weep.