Objects and classes are related concepts, because to use a class, you need to create an object of the class. A class in java is an abstract concept. The class is the blueprint that defines the object. That means you can’t use any of the methods or variables without first creating an instance of the class. The instance of a class is called an object. Objects can be seen as complex entities that can possess different data types and have their own methods. The values an object can hold and the methods an object has access to are based on the class that the object is created from.

In the screenshot below I will illustrate an object and the class that the object represents.

A screen shot of a computer screen

Description automatically generated

Objects are said to be of the type of the class name. If you notice when I create the newExample object I first need to declare it of type “Example”. This means that newExample is an object of type Example, much like a variable can be an integer or some other data type.

In this sample program the class is Example, and it contains two member variables, a constructor, and a method to output to the terminal. To use the Example class, I first need to create an instance of the class, this instance newExample is an object. When I create an object, I need to define some characteristics so that the constructor can build the object. After I have an object created, I can use the method that outputs test by using the newExample.output() to access the method.

References

Eck, D. J. (2019). *Introduction to programming using Java*, version 8.1. Hobart and William Smith Colleges. <http://math.hws.edu/javanotes/>.