**What is inheritance and why is it important?**

* **Explain the meaning of Inheritance**

Inheritance is when a class can get attributes from another class.

* **Highlight a benefit of Inheritance**

As I was learning about inheritance this week, I learned that it has a lot of benefits one of the benefits is that you can call and use other member functions from the base class.

* **Provide an application of Inheritance**

A clear way of using it, perhaps one of the most important ways, is that when working in a project after dividing the tasks for each class (abstraction), it may occur that there will be functions or variable in every single class that will have the same behavior or will hold the same data, using inheritance you store that function or variable in the base class then, any other function in the program can use it.

* **Use a code example of Inheritance from the program you wrote**

For the program we created this week, there were 4 classes, 1 base class and 3 derived classes (the derived classes are the classes that inherit the attributes from the base class).

Class Activity (base)

Class BreathingActivity (derived)

Class ReflectingActivity (derived)

Class ListingActivity (derived)

* **Thoroughly explain these concepts (this likely cannot be done in less than 100 words)**

After following the principles of Abstraction and Encapsulation we notice that there are some behaviors in these classes that are similar and only with Abstraction and Encapsulation it is hard to accomplish these tasks with only a function or any other attributes.

In this case we have similar behaviors like showing the spinner when loading, showing a countdown before any activity, display the starting and the ending messages. Using inheritance, we can set all of these behavior in the base class Activity then have the derived classes using it.

To do that we only need to use : then follow by the name of the base class. Exe.

Class BreathingActivity : Activity

This way the Class BreathingActivity inherit the attributes from the base class, but there are some limitations, it won’t inherit the encapsulated attributes. To get the encapsulated attributes we can use getters and setters but there is the protected method that allows only the derived classes to inherit.

To finish, a derived class can also inherit the constructor parameters from the base class using in front of its constructor the ‘: base ()’ method. Inside of the parentheses we may define which parameters to inherit from the base class.