**Unit 9 cheat sheet:**

Inheritance is a feature of object oriented programming languages. It is a way to derive one class from another.

**Documentation for inheritance:**

<https://docs.oracle.com/javase/tutorial/java/IandI/subclasses.html>

For inheriting from a class in Java, we use the extends keyword:

Public class subclass extends superclass{

}

Where subclass is the child of the superclass (parent).

The subclass will inherit all features for the superclass.

Overriding methods:

Use @Override to highlight in your code that you are overriding a method. Not necessary but will keep things neet.

To Override a method is to use the same name of a method from the superclass in the subclass.

Super keyword:

The super ketword lets you call any method or instance variable to the subclass form the superclass.

This can be used just like the normal object.instanceVariable or object.method() syntax we use in Java.

Example:

public class subclass extends superclass{

public int variable;

public subclass (int variable, int variabl2, variable3)

{

This.variable = variable; //int variable in the subclass gets the first value passed through the //constructor

super(variable2, variable3); //variable2 and variable3 get passed to the constructor in the superclass

}

@Override

public void methodexample()

{

Super.methodexample();//calling methodexample in the main from the subclass will execute the //code in methodexample from the superclass

}

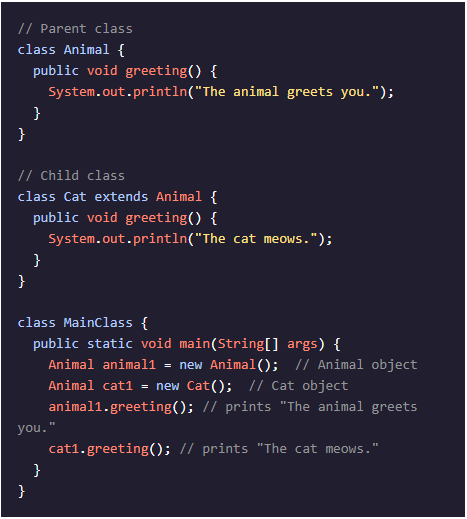
}

Polymorphism:

You can declare a variable of type superclass but create a new subclass:

MySuperClass x = new MySubClass();

* Variable x is of type MySuperClass
* Implemantation will be of class MySubClass
* Example:



\*code from https://www.codecademy.com/learn/learn-java/modules/learn-java-inheritance-and-polymorphism/cheatsheet