TASK 10

The purpose of the @Override annotation in Java is to implement certain methods in a sublcass that is already used by its parent class, so when you add @Override as a line of code before a method containing the same name as its parent class' method, it overrides the method in the parent class that would have been called if we had not used the @Override notation, so as to avoid typographical errors and give more readability to the overall code.

For example, if a method message() prints a message in a parent class, and its subclass wants to use a different method message() to print a different method then we can utilize the @Override annotation to make sure that they remain separate functions but are stil easily readable as “message()” methods that print a line of message as output.

An abstract class cannot be instantiated itself, however it can contain methods to be implemented. For example an abstract class may contain certain variables or parameters to be shared by all of its subclasses to use, so that its subclasses can be implemented. However the abstract class itself merely provides the data for its subclasses to use and it itself is not implemented.