

Week 5 Research:

What are the differences between abstract classes and interfaces? When should you use one over the other?

<https://www.scaler.com/topics/java/difference-between-abstract-class-and-interface/>

An abstract class can declare constructs that interfaces cannot. Methods like non-static, final, private protected and/or public. An abstract class cannot be instantiated but may be extended from one class to another. If a class implements an abstract class, it can only extend for one abstract class. Interface is a very detailed list of expectations. Interface will list step by step what should be done. Like a list of rules that must be followed. Interface can use methods of "public" such as abstract, static or default interfaces. So interfaces are useful when we do not need implementing classes to be related and want to include many implementations.

What is the difference between checked and unchecked exceptions?

<https://rollbar.com/blog/how-to-handle-checked-unchecked-exceptions-in-java/#>

An unchecked exception is a programming error and are fatal. Something has gone wrong with the program and cannot be recovered. A checked exception is an exception condition within the code that can be recovered/ fixed. So while running a code in Java, as we have been learning, in the console area it tells us when there has been an error. Even where to find this error. This would be considered a checked exception. Whereas, when being taught the "for loop", if the i++ was forgotten. It would create an unchecked error that would make the program continue.