Brian Walsh

CPSC 240 Lab 3

**Use Case 1**

Title: Quit

Description: Quits the program

Main Flow

1. User selects that they would like to Quit
2. System saves last user ID used to a file
3. System quits

There is no alternate flow

**Use Case 2**

Title: Print Patients sorted by ID

Description: Prints patients sorted by ID, in ascending order

Main Flow

1. User selects they would like to print the patients sorted by ID
2. System gets the list of patients sorted by ID
3. System prints the list.

Alternate flow

1. The list is empty
2. System prints that the list is empty

**Questions**

1. An exception is an event that disrupts the normal flow of how a program is supposed to work.
2. To handle an exception, you must try a section of code for exceptions, and if an exception is caught there must be code in place to handle that situation.
3. The three kinds of exceptions are: checked exceptions, errors, and runtime exceptions.
4. When I try to compile the ListOfNumbers class I get a message that says “Unhandled Exception Type IOException”. This means that I need to have a try/catch block for this FileIO.
5. The purpose of the try block is to tell the program where to look for exceptions.
6. The purpose of the catch block is to tell the code what to do if an exception occurs.
7. Yes, you can include code to handle multiple exceptions associated with a single try block. For example, you could write ***catch(IOException|SQLException ex){ System.out.println(“EXCEPTION!”);}***.
8. The purpose of a finally block is to execute regardless of whether an exception occurs.
9. The possible exit scenarios are:
   1. If there are no exceptions, the try block executes and prints the list, followed by the finally block executing and the PrintWriter is closed.
   2. If there is an Index Out of Bounds exception, the first catch block executes and prints an error message, followed by the finally block executing and the PrintWriter closing.
   3. If there is an IOException, an error message is printed, and a message stating the PrintWriter was not opened
10. To indicate exceptions that may be thrown by a method you put the “throws” tag in the method title, followed by the Exceptions it may throw.
11. A stack trace is a list of the method calls by the program until an Exception was thrown. You can access it by using .getStackTrace() on an exception. It is useful to show where the program was it when the exception occurred and helps with debugging.
12. If a client can reasonably be expected to recover from an exception, make it a checked exception. If a client cannot do anything to recover from the exception, make it an unchecked reception.
13. The advantages of using exceptions in your program include:
    1. Separates error handling from the rest of the code
    2. Propagates errors up the call stack
    3. Grouping and differentiating error types