CSY2030 Exceptions Lab

1. The *Integer.parseInt()* method requires a String argument, but fails if the string cannot be converted to an integer. Write an application in which you try to parse a String that does not represent an integer value. Catch the *NumberFormatException* that is thrown and then display an appropriate error message. Save the file as *ParseIntError.java*

2. Write a class named *TestScores*. The class constructor should accept an array of test scores as its argument. The class should have a method that returns the average of the test scores. If any test score in the array is negative or greater than 100, the class should throw an *IllegalArgumentException*. Demonstrate the class in a program called *TestScoresDemo.java*.

3. Create an *EmployeeException* class whose constructor receives a String that consists of an *employee's ID* and *payrate*. Save the file as *EmployeeException.java*. Create an *Employee* class with two fields, *idNum* and *hourlyWag*e. The Employee constructor requires values for both fields. Upon construction, throw an *EmployeeException* if the *hourlyWage* is less than £6.00 or over £50.00. Save the class as *Employee.java*. Write an application that establishes at least three Employees with hourlyWages that are above, below, and within the allowed range. Display an appropriate message when an Employee is successfully created and when one is not. Save the file as *ThrowEmployee.java*.

4. Create a *DataEntryException* class whose *getMessage()* method returns information about invalid integer data. Write a program named *GetIDAndAge* that continually prompts the user for an ID number and an age until a terminal 0 is entered for both. Throw a *DataEntryException* if the ID is not in the range of valid ID numbers (0 through 999), or if the age is not in the range of valid ages (0 through 119). Catch any *DataEntryException* or *InputMisMatchException* that is thrown, and display an appropriate message. Save the files as *DataEntryException.java* and *GetIDAndAge.java*.