Practical 8 – Week 8
File Input/Output & Exception Handling

1. Write a program that carries out the following tasks:

   Open a file with the name hello.txt.
   Store the message "Hello, World!" in the file.
   Close the file.
   Open the same file again.
   Read the content of the file and print it.

2. Write a program that reads a file containing text. Read each line and send it to the output file, preceded by line numbers. If the input file is:

   Mary had a little lamb
   Whose fleece was white as snow.
   And everywhere that Mary went,
   The lamb was sure to go!

   then the program produces the output file:

   /* 1 */ Mary had a little lamb
   /* 2 */ Whose fleece was white as snow.
   /* 3 */ And everywhere that Mary went,
   /* 4 */ The lamb was sure to go!

   The line numbers are enclosed in /* */ delimiters so that the program can be used for numbering Java source files.

3. Key in the following program that prompts user to input two integers: the numerator and denominator. Run the program with the user inputs as follows:

```
import java.util.Scanner;

public class ExceptionApp {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter an integer numerator: ");
      int numerator = sc.nextInt();
      System.out.print("Enter an integer denominator: ");
      int denominator = sc.nextInt();
      int result = numerator / denominator;
      System.out.printf("Result of %d / %d is %d\n",
                  Numerator, denominator, result);
   }
}
```

   1st run:

```
Enter an integer numerator: 10
Enter an integer denominator: 3
Result of 10 / 3 is 3
```

2nd run:

```
Enter an integer numerator: 10
Enter an integer denominator: abc
Exception in thread "main" java.util.InputMismatchException
    at java.util.Scanner.throwFor(Scanner.java:840)
    at java.util.Scanner.next(Scanner.java:1461)
    at java.util.Scanner.nextInt(Scanner.java:2091)
    at java.util.Scanner.nextInt(Scanner.java:2050)
    at Topic9Q1.main(Topic9Q1.java:11)
```

3rd run:

```
Enter an integer numerator: 10
Enter an integer denominator: 0
Exception in thread "main" java.lang.ArithmeticException: / by zero
    at Topic9Q1.main(Topic9Q1.java:13)
```

Add the "`try_catch` blocks" to handle two exceptions: `InputMismatchException` and `ArithmeticException`.

4. Using the same program, `ExceptionApp`, add in the **`finally`** block that displays this message "Cleaning up resources…" .

   Run the program with the input data as shown in the above 3 run samples. Is the message in the **finally** block displayed in all the 3 situations?

Optional

5. Write a program that reads a file containing two columns of floating-point numbers. Prompt the user for the file name. Print the average of each column. Assuming that the file is stored under the resources directory of project.

6. Write a program to

   i. Save a list of Friend (each Friend has name, contact and likes, refer to previous practical) objects to a text file.
   ii. Read from the text file and display the list of Friend objects to standard output.

- End -