**Hints: Program 1**

1. In Eclipse, **place your text file in the PROJECT folder, not in the src** .
2. Remember to  **provide the program the filename using command-line arguments**, so that it reads the filename from main's array of strings called args,  rather than asking the user to type the name of the file as the program runs.  
       As mentioned in class**, in Eclipse**, you provide **command line arguments** this way:  
       - pull down on RUN, choose RUN CONFIGURATIONS.  
       - in the left window pane, click on JAVA APPLICATION  
       - in the right window, choose the ARGUMENTS tab  
       - type your filename  (i.e. testScores.txt ) into the PROGRAM ARGUMENTS.  
    Now, in your program, grab the filename from args[0] rather than asking the user to type it in.
3. If you have a**text file delimited by a comma followed by whitespace**, you could use a regular expression as an argument to the useDelimiter method to make that work.  For example, to read from a textfile called testScores.txt, you could use a statement like this:

            sc = new Scanner(new BufferedReader(  
                      new FileReader("testScores.txt"))).useDelimiter(", \\s\*");

**I provided some information about EXCEPTIONS in our class Moodle**

**Here's a short example that illustrates those concepts:**

//reads from Scanner stream -- delimited by whitespace  
**import** java.io.IOException;  
**import** java.util.Scanner;  
**public** **class** ExampleThrowsIOException {  
 **public** **static** **void** main(String[] args) **throws** IOException{  
 String lastName, firstName; **float** gpa;  
 Scanner sc = **null**;  
 **try** { sc = **new** Scanner("John Talbot 4.0 Mary Jones 3.5 Steven Jackson 3.8");  
 **while** (sc.hasNext())  
 { firstName=sc.next(); lastName=sc.next(); gpa=sc.nextFloat();  
 System.*out*.printf("%s, %s %4.1f \n", lastName, firstName, gpa);  
 }//end while  
 }//end try  
 **finally**  
 { **if** (sc != **null**){ sc.close(); }  
 }//end finally

}//end main

}//end class