***Lab 4 – Exception Handling***

Exercise:

1. Write a program that meets the following requirements:
   1. Create an array with one hundred randomly chosen integers.
   2. Cause an exception, *ArrayIndexOutOfBoundsException*, display the message “Out Of Bound”. You can display all the array elements using looping.
2. Given the Loan class below:

|  |
| --- |
| 1. **package** loan; 2. **import** java.util.Date; 3. **public** **class** Loan { 4. **private** **double** annualInterestRate; 5. **private** **int** numberOfYears; 6. **private** **double** loanAmount; 7. **private** java.util.Date loanDate; 9. **public** Loan() { 10. // **TODO** Auto-generated constructor stub 11. } 12. **public** Loan(**double** annualInterestRate, **int** numberOfYears, 13. **double** loanAmount) { 14. **super**(); 15. **this**.annualInterestRate = annualInterestRate; 16. **this**.numberOfYears = numberOfYears; 17. **this**.loanAmount = loanAmount; 18. **this**.loanDate = **new** java.util.Date(); 19. } 20. **public** **double** getAnnualInterestRate() { 21. **return** annualInterestRate; 22. } 23. **public** **void** setAnnualInterestRate(**double** annualInterestRate) { 24. **this**.annualInterestRate = annualInterestRate; 25. } 26. **public** **int** getNumberOfYears() { 27. **return** numberOfYears; 28. } 29. **public** **void** setNumberOfYears(**int** numberOfYears) { 30. **this**.numberOfYears = numberOfYears; 31. } 32. **public** **double** getLoanAmount() { 33. **return** loanAmount; 34. } 35. **public** **void** setLoanAmount(**double** loanAmount) { 36. **this**.loanAmount = loanAmount; 37. } 38. **public** java.util.Date getLoanDate() { 39. **return** loanDate; 40. } 41. **public** **double** monthlyPayment(){ 42. **return** 0.0;//return actual monthly payment 43. } 45. **public** **double** totalPayment(){ 46. **return** 0.0;//return total payment 47. } 49. } |

Modify the Loan class to throw IllegalArgumentException if the loan amount, interest rate or number of years is less than or equal to zero.

1. Consider a Calculator program, note that number 1 and number 2 were a non-numeric string, the program would report exceptions. Modify the program with an exception handler to catch ArithmeticException (e.g., divided by 0) and NumberFormatException (e.g., input is not an integer), and display the errors in a message dialog box.