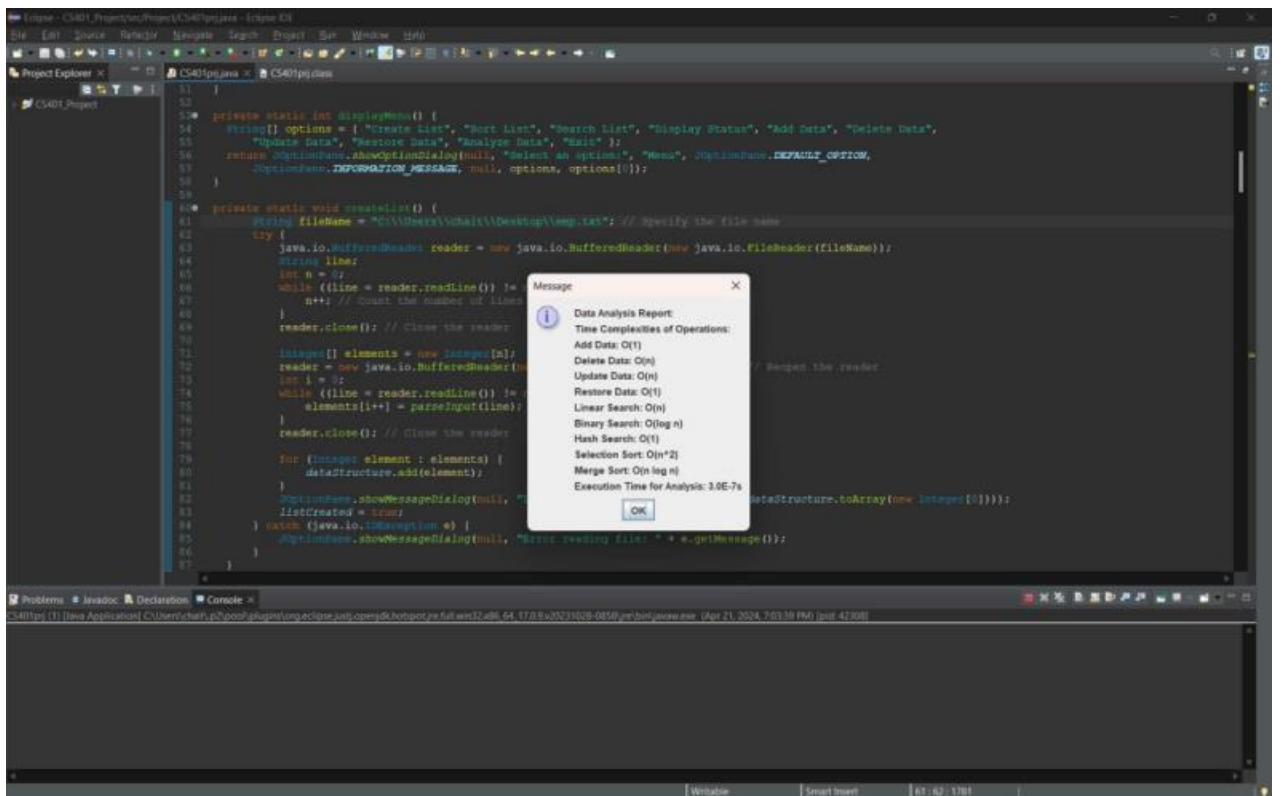
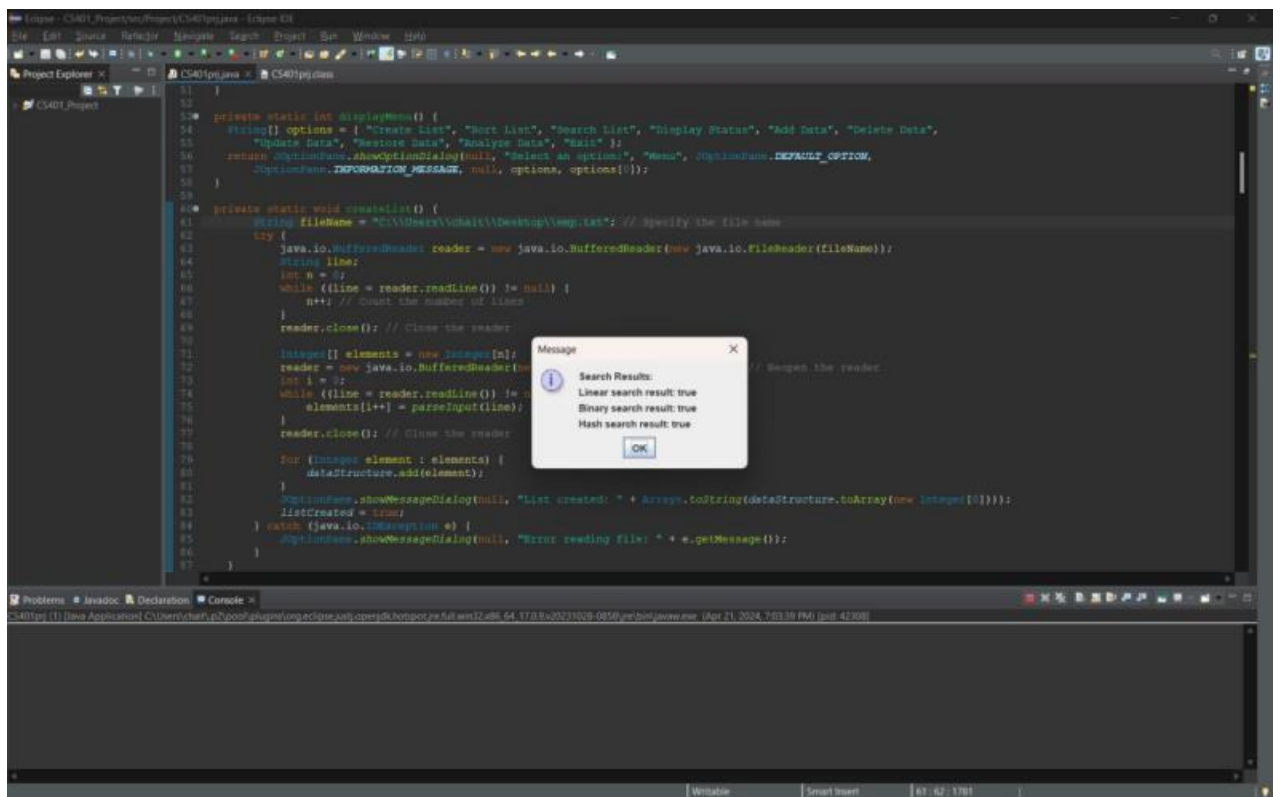


e) Testing document (your own created one) with input data file

- Predefined input is taken from the emp.txt file.



- This is the output for Test.java where the comparisons are displayed.

The screenshot shows the Eclipse IDE with the following components:

- Project Explorer:** Shows a project named 'CS401' with sub-projects like 'BinarySearch.java', 'BubbleSort.java', 'QuickSort.java', 'LinearSearch.java', 'HashSearch.java', and 'DataManipulator.java'.
- Source Editor:** Displays the code for 'Test.java'. The code includes a 'main' method that tests various sorting and searching algorithms. The code is as follows:

```
public class Test {  
    public static void main(String[] args) {  
        // Test data setup  
        Integer[] numbers = {69, 75, 23, 38, 15, 13};  
        Integer[] sortedNumbers = Arrays.copyOf(numbers, numbers.length);  
        Arrays.sort(sortedNumbers); // Sorted array for binary search and to verify sort algorithms  
  
        // Instantiate and test the sorting algorithms  
        StringBuilder sortingResults = new StringBuilder();  
        sortingResults.append("Bubble Sort Result: ").append(Arrays.toString(testBubbleSort(numbers))).append("\n");  
        sortingResults.append("Quick Sort Result: ").append(Arrays.toString(testQuickSort(numbers))).append("\n");  
        sortingResults.append("Linear Search Result: ").append(testLinearSearch(numbers, 23)).append("\n");  
        sortingResults.append("Binary Search Result: ").append(testBinarySearch(numbers, 23)).append("\n");  
        sortingResults.append("Hash Search Result: ").append(testHashSearch(numbers, 23)).append("\n");  
  
        // Instantiate and test the data manipulation algorithms  
        testDataManipulation();  
  
        // Display the results using JOptionPane  
        JOptionPane.showMessageDialog(null, sortingResults.toString());  
    }  
  
    private static Integer[] testBubbleSort(Integer[] numbers) {  
        Integer[] bubbleSort = new Integer[numbers.length];  
        bubbleSort = bubbleSort(numbers);  
        return bubbleSort;  
    }  
}
```
- Test Results Dialog:** A modal dialog box titled 'Test Results' with an information icon. It contains the following text:

```
Bubble Sort Result: [13, 15, 23, 38, 69, 75]  
Quick Sort Result: [13, 15, 23, 38, 69, 75]  
  
Linear Search Result: Element found at index 2  
Linear Search Comparisons: 3  
  
Binary Search Result: Element found at index 2  
Binary Search Comparisons: 1  
  
Hash Search Result: Element found in bucket index 0  
Hash Search Comparisons: 1
```
- Console:** Shows the output of the program. It includes the path to the Java application, the last action performed ('Restored data: 3'), and the updated data list:

```
Test (1) [Java Application] C:\Users\chait\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.9.v20231028-0850\jre\bin\javaw.exe (Apr 21, 2024, 7:46:34 PM) [pid: 72944]  
Last action: Restored data: 3  
Deleted elements: []  
  
Updated data list:  
10  
2  
4  
5  
6  
3
```