

ASSIGNMENT – 3

Subject: CSW2 (CSE 2141)

Name: Arpit Kumar Mohanty

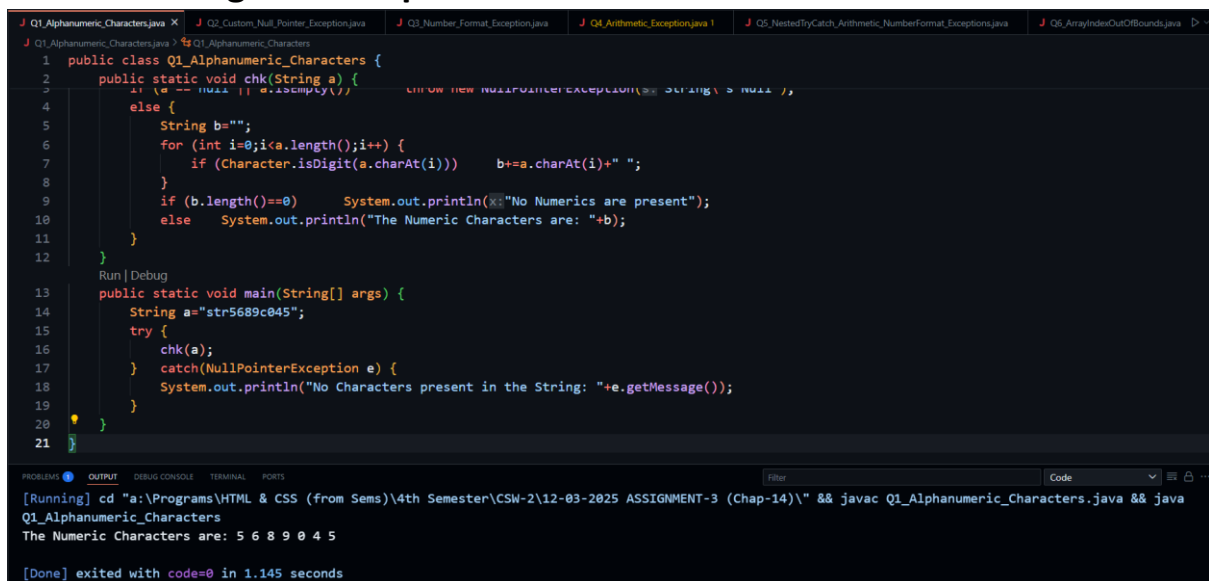
Registration Number: 2341013237

Section: 23412G1

Branch: CSE

Q1. You are given a string containing alphanumeric characters, and your task is to design a Java program that extracts and displays the numeric characters from the given string. If no numeric characters are present, the program should display an appropriate message indicating their absence. Additionally, if the input string is null or empty, the program must throw a `NullPointerException` with a meaningful error message.

Solution along with Output:



```
Q1_Alphanumeric_Characters.java
1 public class Q1_Alphanumeric_Characters {
2     public static void chk(String a) {
3         if (a == null || a.isEmpty()) throw new NullPointerException("String is null");
4         else {
5             String b="";
6             for (int i=0;i<a.length();i++) {
7                 if (Character.isDigit(a.charAt(i))) b+=a.charAt(i)+" ";
8             }
9             if (b.length()==0) System.out.println("No Numerics are present");
10            else System.out.println("The Numeric Characters are: "+b);
11        }
12    }
13    public static void main(String[] args) {
14        String a="str5689c045";
15        try {
16            chk(a);
17        } catch (NullPointerException e) {
18            System.out.println("No Characters present in the String: "+e.getMessage());
19        }
20    }
21 }
```

OUTPUT

```
[Running] cd "a:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)\\" && javac Q1_Alphanumeric_Characters.java && java Q1_Alphanumeric_Characters
The Numeric Characters are: 5 6 8 9 0 4 5
[Done] exited with code=0 in 1.145 seconds
```

Q2. Implement a custom exception class named `CustomNullPointerException` that replicates the behavior of the standard `NullPointerException`. However, instead of relying on default error messages or null references, this custom exception should accept a `String` message as a constructor argument. Your task is to create this

custom exception class and demonstrate its usage in a Java program.

Solution along with Output:

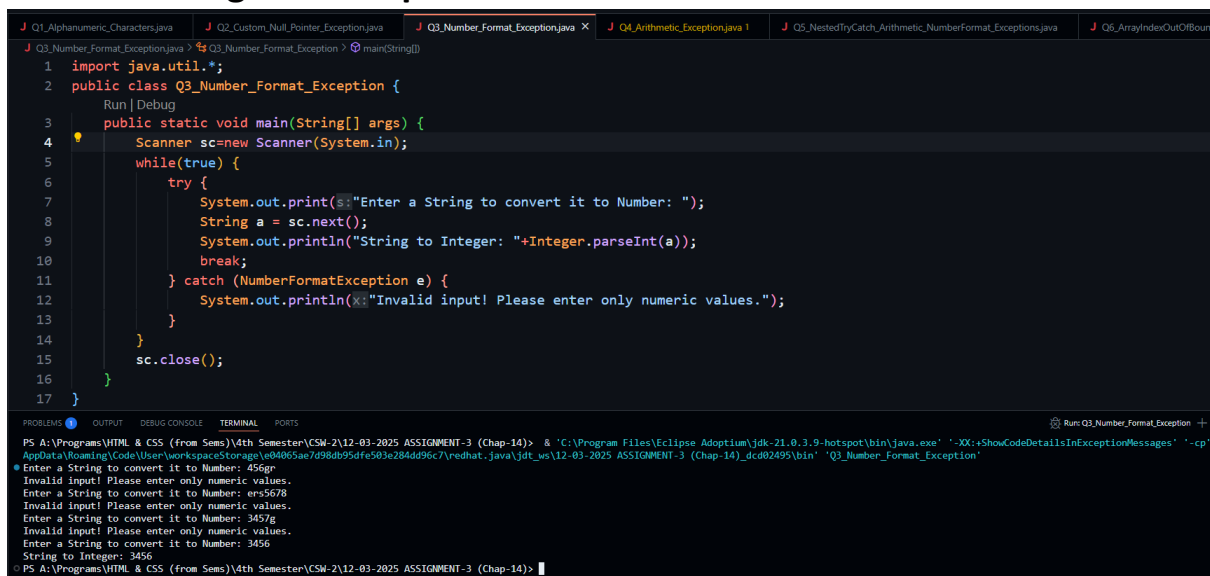


```
J Q1_Alphanumeric_Characters.java J Q2_Custom_Null_Pointer_Exception.java X J Q3_Number_Format_Exception.java J Q4_Arithmetic_Exception.java 1 J Q5_NestedTryCatch_Arithmetic_Nu
J Q2_Custom_Null_Pointer_Exception.java > Q2_Custom_Null_Pointer_Exception > main(String[])
1 class CustomNullPointerException extends NullPointerException {
2     public CustomNullPointerException(String message) {
3         super(message); // Call superclass constructor with message
4     }
5 }
6 public class Q2_Custom_Null_Pointer_Exception {
7     Run | Debug
8     public static void main(String[] args) {
9         try {
10             int[] ar = null; // {1,2,3};
11             if (ar==null) throw new CustomNullPointerException(message:"Array is Null");
12         } catch (NullPointerException e) {
13             System.out.println(e.getMessage());
14         }
15     }
16 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter Code
[Running] cd "a:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)\\" && javac
Q2_Custom_Null_Pointer_Exception.java && java Q2_Custom_Null_Pointer_Exception
Array is Null
```

Q3. Create a method that accepts a string input and converts it into an integer. Use a trycatch block to handle NumberFormatException, and if an exception occurs, prompt the user to enter a valid numeric value.

Solution along with Output:

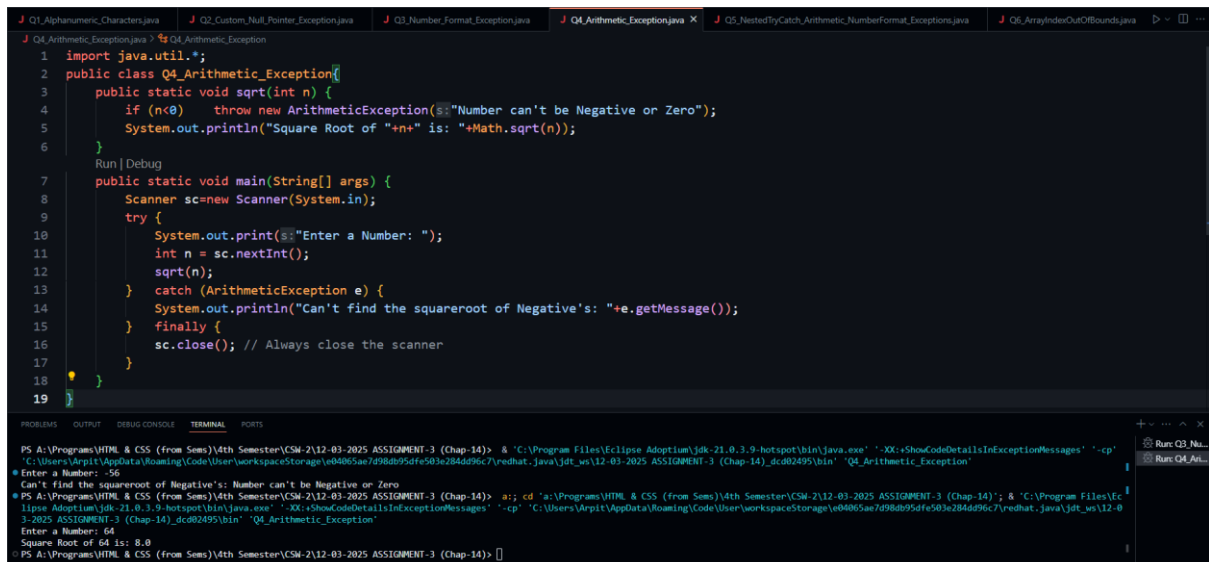


```
J Q1_Alphanumeric_Characters.java J Q2_Custom_Null_Pointer_Exception.java J Q3_Number_Format_Exception.java X J Q4_Arithmetic_Exception.java 1 J Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions.java J Q6_ArrayIndexOutOfBounds
J Q3_Number_Format_Exception.java > Q3_Number_Format_Exception > main(String[])
1 import java.util.*;
2 public class Q3_Number_Format_Exception {
3     Run | Debug
4     public static void main(String[] args) {
5         Scanner sc=new Scanner(System.in);
6         while(true) {
7             try {
8                 System.out.print(s:"Enter a String to convert it to Number: ");
9                 String a = sc.next();
10                System.out.println("String to Integer: "+Integer.parseInt(a));
11                break;
12            } catch (NumberFormatException e) {
13                System.out.println(k:"Invalid input! Please enter only numeric values.");
14            }
15        }
16        sc.close();
17    }
18 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter Code
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
AppData\Roaming\Code\User\workspaceStorage\e04065ae7d98db95dfe503e284dd96c7\redhat_java\jdk_ws\12-03-2025 ASSIGNMENT-3 (Chap-14)_dcd02495\bin' 'Q3_Number_Format_Exception'
Enter a String to convert it to Number: 456g
Invalid input! Please enter only numeric values.
Enter a String to convert it to Number: ers5678
Invalid input! Please enter only numeric values.
Enter a String to convert it to Number: 345/g
Invalid input! Please enter only numeric values.
Enter a String to convert it to Number: 3456
String to Integer: 3456
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)>
```

Q4. Write a Java program to find the square root of an integer number. Demonstrate the use of a try-catch block to handle ArithmeticException.

Solution along with Output:



```
1 import java.util.*;
2 public class Q4_Arithmetic_Exception {
3     public static void sqrt(int n) {
4         if (n<0) throw new ArithmeticException("Number can't be Negative or Zero");
5         System.out.println("Square Root of "+n+" is: "+Math.sqrt(n));
6     }
7
8     public static void main(String[] args) {
9         Scanner sc=new Scanner(System.in);
10        try {
11            System.out.print("Enter a Number: ");
12            int n = sc.nextInt();
13            sqrt(n);
14        } catch (ArithmeticException e) {
15            System.out.println("Can't find the squareroot of Negative's: "+e.getMessage());
16        } finally {
17            sc.close(); // Always close the scanner
18        }
19    }
20 }
```

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14) > & "C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe" ^-XX:+ShowCodeDetailsInExceptionMessages^ ^-cp "C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\eb4065ae7d98db95dfe503e284d496c7\redhat-java\jdk_ws12-03-2025 ASSIGNMENT-3 (Chap-14)_dc02495\bin" "Q4_Arithmetic_Exception"

Enter a Number: -56
Can't find the squareroot of Negative's: Number can't be Negative or Zero

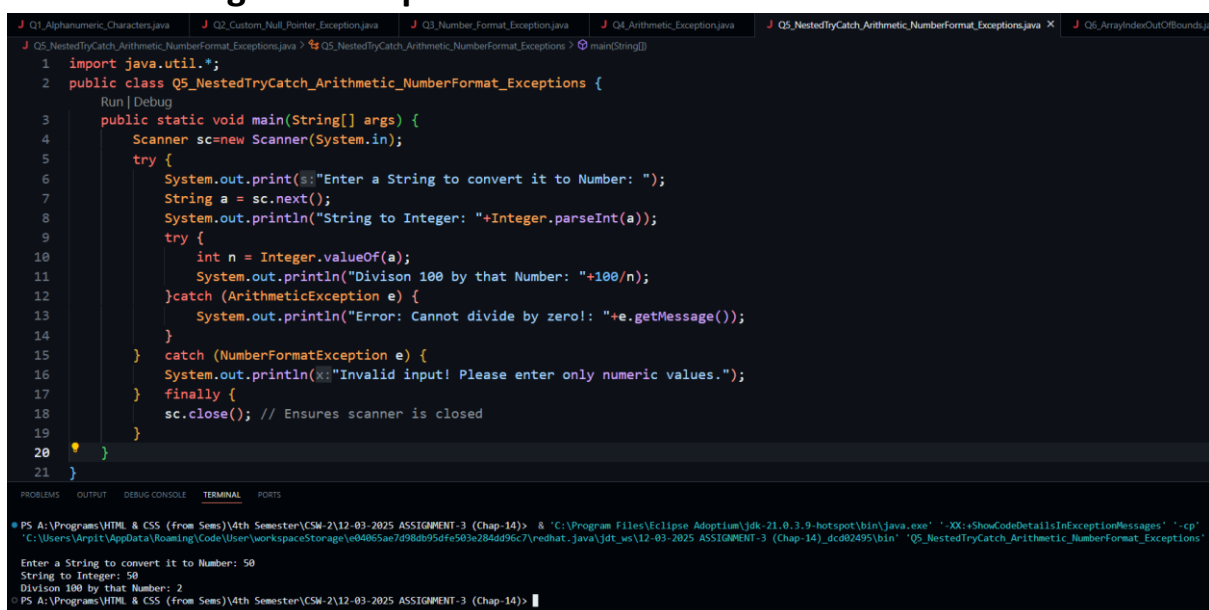
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14) > a;; cd 'e:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14)'; & "C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe" ^-XX:+ShowCodeDetailsInExceptionMessages^ ^-cp "C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\eb4065ae7d98db95dfe503e284d496c7\redhat-java\jdk_ws12-03-2025 ASSIGNMENT-3 (Chap-14)_dc02495\bin" "Q4_Arithmetic_Exception"

Enter a Number: 64
Square Root of 64 is: 8.0

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14) > |

Q5. Demonstrate the use of a nested try-catch block. Write a Java program where the outer try-catch block handles a NumberFormatException, while the inner try-catch block handles an ArithmeticException.

Solution along with Output:



```
1 import java.util.*;
2 public class Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions {
3     public static void main(String[] args) {
4         Scanner sc=new Scanner(System.in);
5         try {
6             System.out.print("Enter a String to convert it to Number: ");
7             String a = sc.next();
8             System.out.println("String to Integer: "+Integer.parseInt(a));
9             try {
10                int n = Integer.valueOf(a);
11                System.out.println("Division 100 by that Number: "+100/n);
12            } catch (ArithmeticException e) {
13                System.out.println("Error: Cannot divide by zero!: "+e.getMessage());
14            }
15        } catch (NumberFormatException e) {
16            System.out.println("Invalid input! Please enter only numeric values.");
17        } finally {
18            sc.close(); // Ensures scanner is closed
19        }
20    }
21 }
```

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14) > & "C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe" ^-XX:+ShowCodeDetailsInExceptionMessages^ ^-cp "C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\eb4065ae7d98db95dfe503e284d496c7\redhat-java\jdk_ws12-03-2025 ASSIGNMENT-3 (Chap-14)_dc02495\bin" "Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions"

Enter a String to convert it to Number: 50
String to Integer: 50
Division 100 by that Number: 2

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14) > |

Q6. Implement a Java program that performs complex manipulations on an array of integers, including operations such as sorting, searching, and accessing elements at various indices. Introduce scenarios where accessing elements beyond the array bounds leads to an `ArrayIndexOutOfBoundsException`. Handle these exceptions gracefully to ensure the program continues execution without crashing.

Solution along with Output:

```
Q1_Alphanumeric_Characters.java Q2_Custom_Null_Pointer_Exception.java Q3_Number_Format_Exception.java Q4_Arithmetic_Exception.java Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions.java Q6_ArrayIndexOutOfBounds.java X
1 import java.util.*;
2 public class Q6_ArrayIndexOutOfBounds {
3     Run | Debug
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int[] ar = {19, 47, 2, 56, 90, 100, 1, 45};
7
8         System.out.print("\nArray before Sorting: ");
9         for (int e : ar) {
10             System.out.print(e + " ");
11         }
12
13         Arrays.sort(ar);
14
15         System.out.print("\nArray after Sorting: ");
16         for (int e : ar) {
17             System.out.print(e + " ");
18         }
19
20         System.out.print("\nEnter a number to search for: ");
21         if (sc.hasNextInt()) {
22             int target = sc.nextInt();
23             int index = Arrays.binarySearch(ar, target);
24
25             if (index >= 0) {
26                 System.out.println("Number found at index: " + index);
27             } else {
28                 System.out.println("Number not found in the array.");
29             }
30         } else {
31             System.out.println("Invalid input! Please enter a valid integer.");
32             sc.next(); // Clear invalid input
33         }
34     }
35 }
```

```
Q1_Alphanumeric_Characters.java Q2_Custom_Null_Pointer_Exception.java Q3_Number_Format_Exception.java Q4_Arithmetic_Exception.java Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions.java Q6_ArrayIndexOutOfBounds.java X
2 public class Q6_ArrayIndexOutOfBounds {
3     public static void main(String[] args) {
4
5         try {
6             System.out.print("\nEnter an index to get an element: ");
7             if (sc.hasNextInt()) {
8                 int index = sc.nextInt();
9                 System.out.println("Number at index " + index + " is: " + ar[index]);
10             } else {
11                 System.out.println("Invalid input! Please enter a valid integer.");
12             }
13         } catch (ArrayIndexOutOfBoundsException e) {
14             System.out.println("Error: Invalid index! Please enter a value between 0 and " + (ar.length - 1));
15         }
16         sc.close();
17     }
18 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS A:\Programs\HTML & CSS (from Sem5)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & "C:\Program Files\Eclipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe" "-XG:+ShowCodeDetailsInExceptionMessages" "-cp"
"C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\eb4065ae7d98db95dfe503e284d96c7\redhat.java\jdk_ws\12-03-2025 ASSIGNMENT-3 (Chap-14)_dcd02495\bin\" "Q6_ArrayIndexOutOfBounds"
Array before Sorting: 19 47 2 56 90 100 1 45
Array after Sorting: 1 2 19 45 47 56 90 100
Enter a number to search for: 19
Number found at index: 2
Enter an index to get an element: 30
Error: Invalid index! Please enter a value between 0 and 7
PS A:\Programs\HTML & CSS (from Sem5)\4th Semester\CSM-2\12-03-2025 ASSIGNMENT-3 (Chap-14)>
```

Q7. Design a Java program to perform matrix operations such as addition, multiplication, and transpose. Introduce scenarios where accessing elements beyond the matrix bounds results in an

ArrayIndexOutOfBoundsException. Handle these exceptions effectively and provide meaningful error messages that clearly indicate the nature of the exception

Solution along with Output:

```
J Q7_MatrixOperations_ArrayIndexOutOfBoundsException.java > Q7_MatrixOperations_ArrayIndexOutOfBoundsException > main(String[])
1 import java.util.*;
2 public class Q7_MatrixOperations_ArrayIndexOutOfBoundsException {
3
4     public static int[][] addMatrices(int[][] m1, int[][] m2) {
5         int rows = m1.length, cols = m1[0].length;
6         int[][] result = new int[rows][cols];
7
8         for (int i = 0; i < rows; i++) {
9             for (int j = 0; j < cols; j++) {
10                 result[i][j] = m1[i][j] + m2[i][j];
11             }
12         }
13         return result;
14     }
15
16     public static int[][] multiplyMatrices(int[][] m1, int[][] m2) {
17         int rows = m1.length, cols = m2[0].length, common = m2.length;
18         int[][] result = new int[rows][cols];
19
20         for (int i = 0; i < rows; i++) {
21             for (int j = 0; j < cols; j++) {
22                 for (int k = 0; k < common; k++) {
23                     result[i][j] += m1[i][k] * m2[k][j];
24                 }
25             }
26         }
27         return result;
28     }
}
```

```
public class Q7_MatrixOperations_ArrayIndexOutOfBoundsException {
    public static int[][] transposeMatrix(int[][] matrix) {
        int rows = matrix.length, cols = matrix[0].length;
        int[][] transposed = new int[cols][rows];

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                transposed[j][i] = matrix[i][j];
            }
        }
        return transposed;
    }

    public static void printMatrix(int[][] matrix) {
        for (int[] row : matrix) {
            for (int value : row) {
                System.out.print(value + " ");
            }
            System.out.println();
        }
    }
}

Run | Debug
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    int[][] m1 = {{1, 2, 3}, {3, 4, 7}, {31, 45, 70}};
    int[][] m2 = {{10, 31, 34}, {32, 47, 73}, {38, 90, 69}};

    System.out.println("%s", "Matrix 1:");
    printMatrix(m1);

    System.out.println("%s", "\nMatrix 2:");
    printMatrix(m2);

    System.out.println("%s", "\nMatrix Addition Result:");
    printMatrix(addMatrices(m1, m2));
}
```

```

J_Q7_MatrixOperations_ArrayIndexOutOfBounds.java > J_Q7_MatrixOperations_ArrayIndexOutOfBounds.java > main(String[])
2 public class Q7_MatrixOperations_ArrayIndexOutOfBounds {
49 public static void main(String[] args) {
64     System.out.println("\nMatrix Multiplication Result:");
65     printMatrix(multiplyMatrices(m1, m2));
66
67     System.out.println("\nTranspose of Matrix 1:");
68     printMatrix(transposeMatrix(m1));
69
70     try {
71         System.out.print("\nEnter row and column index to access an element: ");
72         int r = sc.nextInt();          int c = sc.nextInt();
73         System.out.println("Matrix-1 Element: " + m1[r][c] + ", Matrix-2 Element: " + m2[r][c]);
74     } catch (ArrayIndexOutOfBoundsException e) {
75         System.out.println("Invalid index! Please enter values between 0 and " + (m1.length - 1));
76     }
77     sc.close();
78 }
79 }

```

Matrix 1:

```

1 2 3
3 4 7
31 45 70

```

Matrix 2:

```

10 31 34
32 47 73
38 90 69

```

Matrix Addition Result:

```

11 33 37
35 51 80
69 135 139

```

Matrix Multiplication Result:

```

188 395 387
424 911 877
4410 9376 9169

```

Transpose of Matrix 1:

```

1 3 31
2 4 45
3 7 70

```

Enter row and column index to access an element: 5 3

Invalid index! Please enter values between 0 and 2

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> |

Q8. Create a custom-checked exception class named CustomCheckedException. Use this exception in your program to handle a specific error condition and demonstrate its usage with a try-catch block.

Solution along with Output:

```
J Q8_CustomCheckedException.java > Q8_CustomCheckedException > main(String[])
1  import java.util.*;
2  class CustomCheckedException extends Exception {
3      public CustomCheckedException(String message) {
4          super(message);
5      }
6  }
7  public class Q8_CustomCheckedException {
8      public static void main(String[] args) {
9          Scanner sc=new Scanner(System.in);
10         System.out.print(s:"Enter the Age: ");
11         int age = sc.nextInt();
12         try {
13             if (age < 18) {
14                 throw new CustomCheckedException(message:"Age must be 18 or above to proceed.");
15             }
16             System.out.println(x:"Access granted! Age is valid.");
17         } catch (CustomCheckedException e) {
18             System.out.println("Caught Exception: " + e.getMessage());
19         } finally {
20             sc.close();
21         }
22     }
23 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\04065ae7d98db95dfe503e284dd96c7\redhat.java\jdt_ws\12-03-2025 ASSIGNMENT-3 (Chap-14)_dcd02495\bin' 'Q8_CustomCheckedException'
Enter the Age: 15
Caught Exception: Age must be 18 or above to proceed.
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> a;; cd 'a:\Programs\HTML & CSS (from Sems)\4th Semester\Eclipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\04065ae7d98db95dfe503e284dd96c7\redhat.java\jdt_ws\12-03-2025 ASSIGNMENT-3 (Chap-14)_dcd02495\bin' 'Q8_CustomCheckedException'
Enter the Age: 26
Access granted! Age is valid.
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)>
```

Q9. . Implement a method that reads an integer from the user and handles InputMismatchException using a try-catch block.

Solution along with Output:

```
etic_Exception.java  J Q5_NestedTryCatch_Arithmetic_NumberFormat_Exceptions.java  J Q6_ArrayIndexOutOfBounds.java  J Q7_MatrixOperations_Arr
J Q9_InputMismatchException.java
1  import java.util.*;
2  public class Q9_InputMismatchException {
    Run | Debug
3      public static void main(String[] args) {
4          Scanner sc = new Scanner(System.in);
5          try {
6              System.out.print(s:"Enter an integer: ");
7              int n = sc.nextInt(); // May throw InputMismatchException
8              System.out.println("You entered: " + n);
9          } catch (InputMismatchException e) {
10             System.out.println(x:"Invalid input! Please enter a valid integer.");
11          } finally {
12              sc.close();
13          }
14      }
15  }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & 'C:\Program Files\Eclipse Adopt
'C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\e04065ae7d98db95dfe503e284dd96c7\redhat.java\jdt_ws\12-03-2025 ASSI
• Enter an integer: 56ht
Invalid input! Please enter a valid integer.
• PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> a;; cd 'a:\Programs\HTML & CSS (f
lipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Arpit\AppData\Roam
3-2025 ASSIGNMENT-3 (Chap-14)_dcd02495\bin' 'Q9_InputMismatchException'
Enter an integer: 345
You entered: 345
• PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> |
```

Q10. Implement a Java program that reads a file path from the command-line argument and attempts to read its contents. If the file path is null or points to a non-existent file, throw a custom `FileNotFoundException`. If the file exists but cannot be read due to permission issues, throw a custom `FileReadPermissionException`. Your task is to create these custom exception classes and handle them appropriately in your program.

Solution along with Output:


```

J Q10_FileExceptionHandling.java > Q10_FileExceptionHandling > main(String[])
1 import java.util.*; import java.io.*;
2
3 class CustomFileNotFoundException extends Exception {
4     public CustomFileNotFoundException(String message) {
5         super(message);
6     }
7 }
8
9 class CustomFileReadPermissionException extends Exception {
10    public CustomFileReadPermissionException(String message) {
11        super(message);
12    }
13 }
14
15 public class Q10_FileExceptionHandling {
16     Run | Debug
17     public static void main(String[] args) {
18         try {
19             File file = new File(args[0]); // Directly accessing args[0]
20
21             if (!file.exists()) {
22                 throw new CustomFileNotFoundException(message:"Error: File does not exist!");
23             }
24             if (!file.canRead()) {
25                 throw new CustomFileReadPermissionException(message:"Error: No read permission for the file!");
26             }
27             System.out.print(s:"File Contents: ");
28             Scanner sc = new Scanner(file);
29             while (sc.hasNextLine()) {
30                 System.out.println(sc.nextLine());
31             }
32             sc.close();
33         }
34     }
35 }

```

```

J Q10_FileExceptionHandling.java > Q10_FileExceptionHandling > main(String[])
15 public class Q10_FileExceptionHandling {
16     public static void main(String[] args) {
32
33     } catch (ArrayIndexOutOfBoundsException e) {
34         System.out.println(x:"Error: File path not provided!");
35     } catch (CustomFileNotFoundException | CustomFileReadPermissionException e) {
36         System.out.println(e.getMessage());
37     } catch (FileNotFoundException e) {
38         System.out.println(x:"Unexpected error: Unable to open the file.");
39     }
40 }
41 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSN-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
'C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\1e04065ae7d98db95dfe503e284dd96c7\redhat.java\jdk_ws\12-03-2025 ASSIGNMENT-3 (Chap-14)_dc02495\bin' 'Q10_FileExceptionHandling'
Error: File path not provided!
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSN-2\12-03-2025 ASSIGNMENT-3 (Chap-14)>

```

Q11. Write a program that reads data from a file and performs some processing. Handle checked IOException by using try-catch block to catch and handle the exception.

Solution along with Output:

```

J Q11_FileProcessingWithIOException.java > ...
1  import java.io.*;                import java.util.*;
2  public class Q11_FileProcessingWithIOException {
    Run | Debug
3      public static void main(String[] args) {
4          Scanner userInput = new Scanner(System.in);
5
6          // Taking file path as user input
7          System.out.print(s:"Enter the file path: ");
8          String filePath = userInput.nextLine();
9
10         // Try-Catch block to handle IOException
11         try {
12             File file = new File(filePath);
13
14             // Checking if file exists
15             if (!file.exists()) {
16                 throw new FileNotFoundException(s:"Error: File does not exist!");
17             }
18
19             Scanner fileScanner = new Scanner(file);
20             int lineCount = 0, wordCount = 0;
21
22             System.out.println(x:"\nFile Contents:");
23             while (fileScanner.hasNextLine()) {
24                 String line = fileScanner.nextLine();
25                 System.out.println(line);
26                 lineCount++;
27                 wordCount += line.split(regex:"\\s+").length; // Counting words in each line
28             }
29

```

```

J Q11_FileProcessingWithIOException.java > ...
2  public class Q11_FileProcessingWithIOException {
3      public static void main(String[] args) {
49
30         // Display file processing results
31         System.out.println(x:"\nFile Processing Results:");
32         System.out.println("Total Lines: " + lineCount);
33         System.out.println("Total Words: " + wordCount);
34
35         // Closing scanners
36         fileScanner.close();
37     } catch (FileNotFoundException e) {
38         System.out.println("Exception: " + e.getMessage());
39     } catch (IOException e) {
40         System.out.println(x:"IOException occurred while reading the file.");
41     } finally {
42         userInput.close(); // Close user input scanner
43     }
44 }
45 }

```

PROBLEMS 1 OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```

PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)> & 'C:\Program Files\Eclipse Adopti
'C:\Users\Arpit\AppData\Roaming\Code\User\workspaceStorage\e04065ae7d98db95dfe503e284dd96c7\redhat.java\jdt_ws\12-03-2025 ASSIG
Enter the file path: Hello.txt
Exception: Error: File does not exist!
PS A:\Programs\HTML & CSS (from Sems)\4th Semester\CSW-2\12-03-2025 ASSIGNMENT-3 (Chap-14)>

```