



Dr. M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY

University with Graded Autonomy Status
(An ISO 21001 : 2018 Certified Institution)

Periyar E.V.R. High Road, Maduravoyal, Chennai-95. Tamilnadu, India.



PROJECT REPORT

JAVA PROGRAMMING FUNDAMENTALS (EBDS22ET2)

2024-2025(EVEN SEMESTER)

DEPARTMENT OF B.Tech DS & AI (E&T)

COURSE : B-TECH CSE-DS (AI)

YEAR/SEM/SEC : I YEAR – II SEM – AB1

PROJECT TITLE : EXCEPTION HANDLING



Dr. M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY

University with Graded Autonomy Status

(An ISO 21001 : 2018 Certified Institution)

Periyar E.V.R. High Road, Maduravoyal, Chennai-95. Tamilnadu, India.



BONAFIDE CERTIFICATE

JAVA PROGRAMMING FUNDAMENTALS

DEPARTMENT OF B.Tech DS & AI (E&T)

Certified that this project report “ **EXCEPTION HANDLING**” is confirmed work of **S.chandru** I-year B-Tech CSE- DS(AI) in **JAVA PROGRAMMING FUNDAMENTALS (EBDS22ET2)** who carried out the project work under the supervision

Signature of Lab-in-Charge

Signature of Head of Dept

Submitted for the Practical Examination held on_____

Internal Examiner

External Examiner

ABSTRACT

This project focuses on the concept of exception handling in Java, a crucial mechanism that enables developers to handle runtime errors, ensuring smooth program execution. The primary aim is to understand the types of exceptions, how they are thrown and caught, and the use of key constructs like try, catch, throw, throws, and finally. By implementing real-time examples, this project showcases how exception handling enhances the robustness and reliability of Java applications.

INTRODUCTION TO JAVA:

Java is a high-level, object-oriented programming language developed by Sun Microsystems. It follows the principle of "Write Once, Run Anywhere", which means that code compiled in Java can run on any platform with a Java Virtual Machine (JVM). Java offers several key features including platform independence, memory management, multithreading, and robust security. One of its core strengths lies in exception handling, which ensures that the program continues its execution even in the face of unexpected errors or inputs.

PROJECT TITLE

Understanding and Implementing Exception Handling in Java

AIM:

To design and implement Java programs demonstrating how to handle exceptions using try, catch, finally, and user-defined exception classes.

ALGORITHMS:

1. Start the program.
2. Accept input from the user.
3. Try to perform an operation (e.g., division or file access).
4. If an exception occurs, catch it and display a message.
5. Optionally, use a finally block for cleanup.
6. For specific cases, throw a custom exception.
7. End the program.

.

SOURCE CODE:

```
import java.util.Scanner;

// Custom Exception Class
class NegativeNumberException extends Exception {
    public NegativeNumberException(String message) {
        super(message);
    }
}

public class ExceptionHandlingExample {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        try {
            System.out.print("Enter a number: ");
            int number = scanner.nextInt();

            if (number < 0) {
                throw new NegativeNumberException("Negative numbers are not
allowed.");
            }

            int result = 100 / number; // May cause ArithmeticException
            System.out.println("Result: " + result);
        } catch (ArithmeticException e) {
            System.out.println("Error: Cannot divide by zero.");
        } catch (NegativeNumberException e) {
            System.out.println("Error: " + e.getMessage());
        } finally {
            System.out.println("Program execution completed.");
            scanner.close();
        }
    }
}
```

OUTPUT

```
Case 1 – Input: 10
Enter a number: 10
Result: 10
Program execution completed.

Case 2 – Input: 0
Enter a number: 0
Error: Cannot divide by zero.
Program execution completed.

Case 3 – Input: -5
Enter a number: -5
Error: Negative numbers are not allowed.
Program execution completed.
```

RESULT:

The project was successfully implemented and tested. The program demonstrates how exception handling prevents the program from crashing due to runtime errors and how custom exceptions can be created for user-defined conditions.

GITHUB LINKS .

S.CHANDRU – AB1-35;

<https://github.com/chandruab1/EXCEPTION-HANDLING>

