



Vidyavardhini's College of Engineering & Technology
Department of Computer Science And Engineering (Data Science)

Experiment No. 9
Implement a program on Exception handling.
Date of Performance:
Date of Submission:



Vidyavardhini's College of Engineering & Technology

Department of Computer Science And Engineering (Data Science)

Aim: Implement a program on Exception handling.

Objective: To able handle exceptions occurred and handle them using appropriate keyword

Theory:

The Exception Handling in Java is one of the powerful mechanisms to handle the runtime errors so that the normal flow of the application can be maintained.

Exception Handling is a mechanism to handle runtime errors such as ClassNotFoundException, IOException, SQLException, RemoteException, etc.

Java Exception Keywords

Java provides five keywords that are used to handle the exception. The following table describes each.

Keyword	Description
try	The "try" keyword is used to specify a block where we should place an exception code. It means we can't use try block alone. The try block must be followed by either catch or finally.
catch	The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later.
finally	The "finally" block is used to execute the necessary code of the program. It is executed whether an exception is handled or not.
throw	The "throw" keyword is used to throw an exception.
throws	The "throws" keyword is used to declare exceptions. It specifies that there may occur an exception in the method. It doesn't throw an exception. It is always used with method signature.

```
public class JavaExceptionExample{  
  
    public static void main(String args[]){
```



Vidyavardhini's College of Engineering & Technology

Department of Computer Science And Engineering (Data Science)

```
try{

    //code that may raise exception

    int data=100/0;

}catch(ArithmeticException e){System.out.println(e);}

//rest code of the program

System.out.println("rest of the code...");

}

}
```

Output:

```
Exception in thread main java.lang.ArithmeticException:/ by zero
rest of the code...
```

Code:

```
public class ExceptionHandlingExample {

    public static void main(String[] args) {
        try {
            int number = Integer.parseInt("hello");
        } catch (NumberFormatException e) {
            System.out.println("The string \"hello\" is not a valid number.");
        } finally {
            System.out.println("The program is finished.");
        }
    }
}
```



Vidyavardhini's College of Engineering & Technology

Department of Computer Science And Engineering (Data Science)

}

Output:

Output

```
java -cp /tmp/L40PU56o05 ExceptionHandlingExample
The string "hello" is not a valid number.
The program is finished.
```

Conclusion:

Exceptions are handled in Java using the try-catch statement. The try-catch statement allows you to handle errors that occur in your code.

The try block contains the code that you want to run. If an error occurs in the try block, the Java runtime environment (JRE) will throw an exception.

The catch block is used to catch the exception that is thrown by the JRE. The catch block contains the code that you want to execute when an exception occurs.

You can have multiple catch blocks to catch different types of exceptions. If an exception is not caught by any of the catch blocks, the JRE will terminate the program.