Q1) Write a program that tries to access an element outside the bounds of an array and handles the ArrayIndexOutOfBoundsException by printing a user-friendly message.

Program:

**package** Home;

**public** **class** ArrayExceptionHandling {

**public** **static** **void** main(String[] args) {

// Define an array with 5 elements

**int**[] numbers = {1, 2, 3, 4, 5};

// Print the array elements

System.***out***.println("Array elements:");

**for** (**int** i = 0; i < numbers.length; i++) {

System.***out***.println("Index " + i + ": " + numbers[i]);

}

**try** {

// Attempt to access an element outside the bounds of the array

// This will cause an ArrayIndexOutOfBoundsException

System.***out***.println("\nAttempting to access index 10:");

**int** outOfBoundsElement = numbers[10];

System.***out***.println("Element at index 10: " + outOfBoundsElement);

} **catch** (ArrayIndexOutOfBoundsException e) {

// Handle the exception by printing a user-friendly message

System.***out***.println("\nError: Tried to access an element outside the bounds of the array.");

System.***out***.println("Exception message: " + e.getMessage());

}

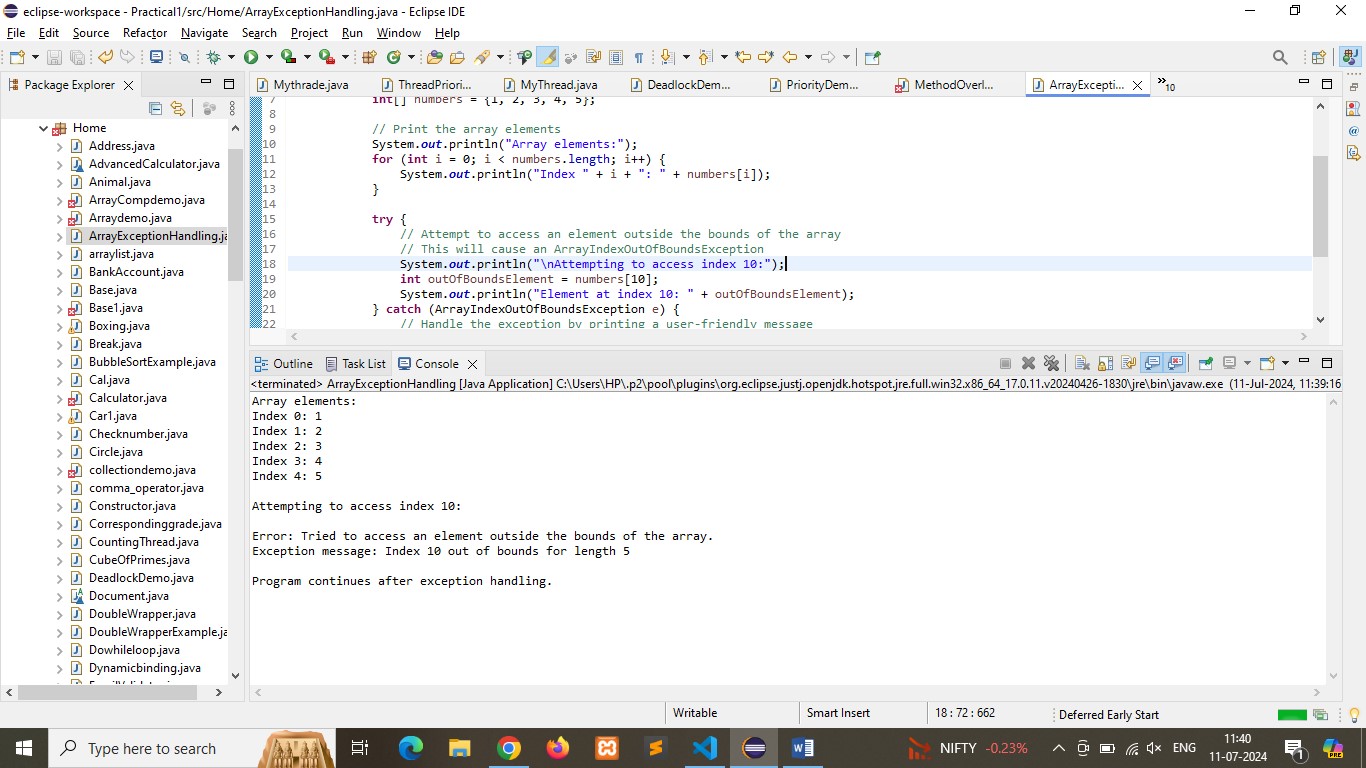
// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:



Q2) Write a program that attempts to divide a number by zero and handles the ArithmeticException by printing a message that division by zero is not allowed.

Program:

**package** Home;

**public** **class** DivisionByZeroHandling {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

// Define two integers for division

**int** numerator = 10;

**int** denominator = 0;

// Print the values of numerator and denominator

System.***out***.println("Numerator: " + numerator);

System.***out***.println("Denominator: " + denominator);

**try** {

// Attempt to divide by zero

// This will cause an ArithmeticException

System.***out***.println("\nAttempting to divide " + numerator + " by " + denominator + ":");

**int** result = numerator / denominator;

System.***out***.println("Result: " + result);

} **catch** (ArithmeticException e) {

// Handle the exception by printing a user-friendly message

System.***out***.println("\nError: Division by zero is not allowed.");

System.***out***.println("Exception message: " + e.getMessage());

}

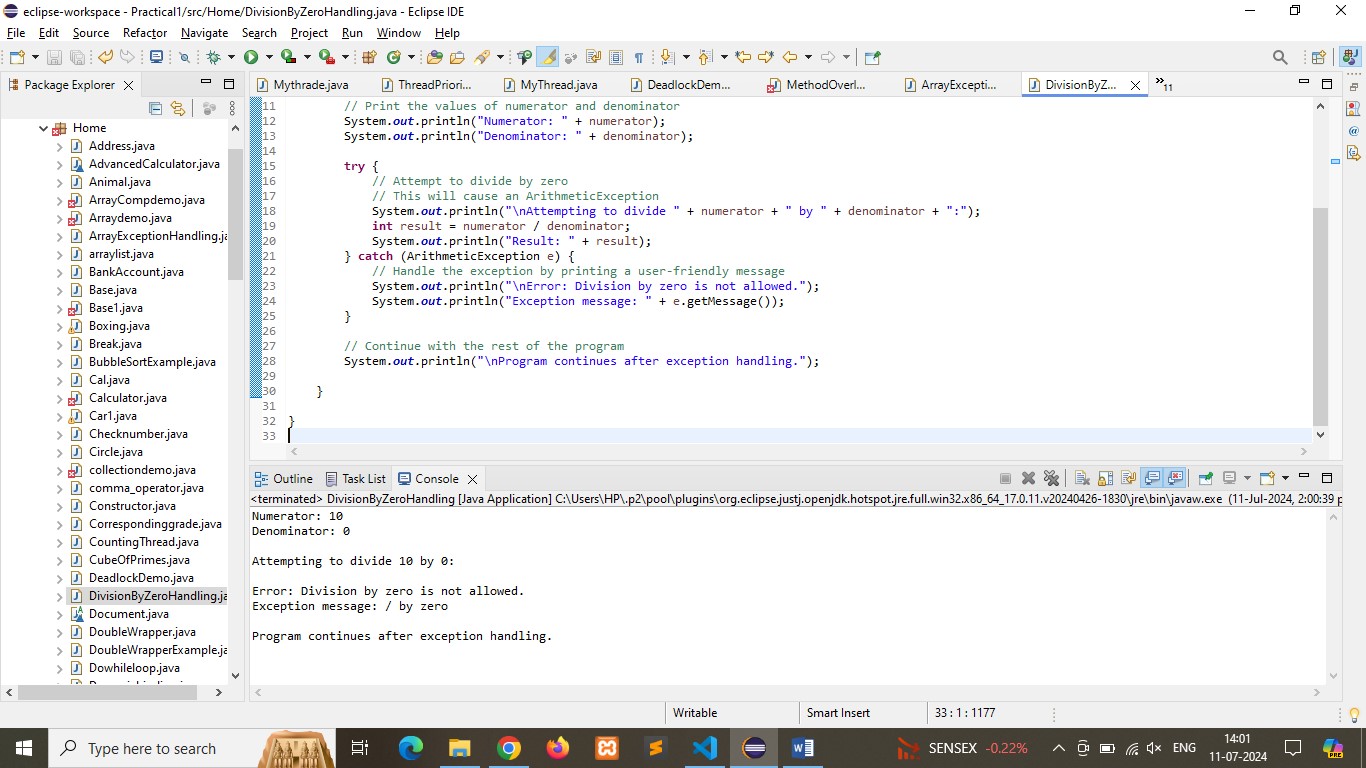
// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:



Q3) Write a Java program that reads an integer input from the user and throws an IllegalArgumentException if the input is negative. Display an appropriate message when the exception is caught.

Program:

**package** Home;

**import** java.util.Scanner;

**public** **class** NegativeNumberExceptionHandling {

**public** **static** **void** main(String[] args) {

// Create a Scanner object for reading input

Scanner scanner = **new** Scanner(System.***in***);

**try** {

// Prompt the user to enter an integer

System.***out***.print("Enter an integer: ");

**int** userInput = scanner.nextInt();

// Check if the input is negative

**if** (userInput < 0) {

// Throw an IllegalArgumentException if the input is negative

**throw** **new** IllegalArgumentException("Negative numbers are not allowed.");

}

// Print the input if it is valid (non-negative)

System.***out***.println("You entered: " + userInput);

} **catch** (IllegalArgumentException e) {

// Handle the exception by printing a user-friendly message

System.***out***.println("\nError: " + e.getMessage());

} **catch** (Exception e) {

// Handle any other exceptions that may occur

System.***out***.println("\nError: Invalid input. Please enter a valid integer.");

} **finally** {

// Close the scanner

scanner.close();

}

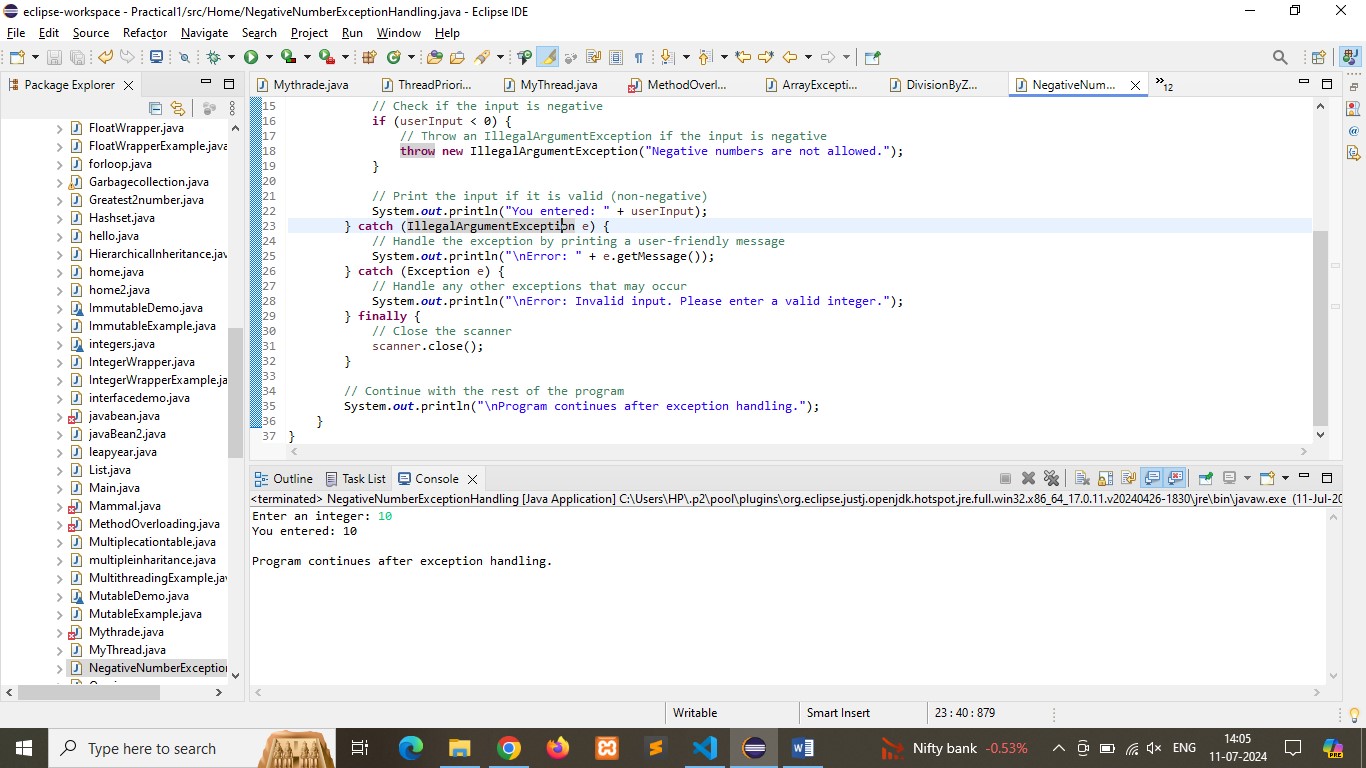
// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:



Q4) Create a Java method that divides two numbers and declares that it throws an ArithmeticException. Handle the exception in the main method.

Program:

**package** Home;

**public** **class** DivisionMethodHandling {

// Method to divide two numbers, declares that it throws ArithmeticException

**public** **static** **int** divide(**int** numerator, **int** denominator) **throws** ArithmeticException {

**if** (denominator == 0) {

// Throw an ArithmeticException if denominator is zero

**throw** **new** ArithmeticException("Division by zero is not allowed.");

}

**return** numerator / denominator;

}

**public** **static** **void** main(String[] args) {

// Define two integers for division

**int** numerator = 10;

**int** denominator = 0;

// Print the values of numerator and denominator

System.***out***.println("Numerator: " + numerator);

System.***out***.println("Denominator: " + denominator);

**try** {

// Attempt to divide the numbers using the divide method

System.***out***.println("\nAttempting to divide " + numerator + " by " + denominator + ":");

**int** result = *divide*(numerator, denominator);

System.***out***.println("Result: " + result);

} **catch** (ArithmeticException e) {

// Handle the exception by printing a user-friendly message

System.***out***.println("\nError: " + e.getMessage());

}

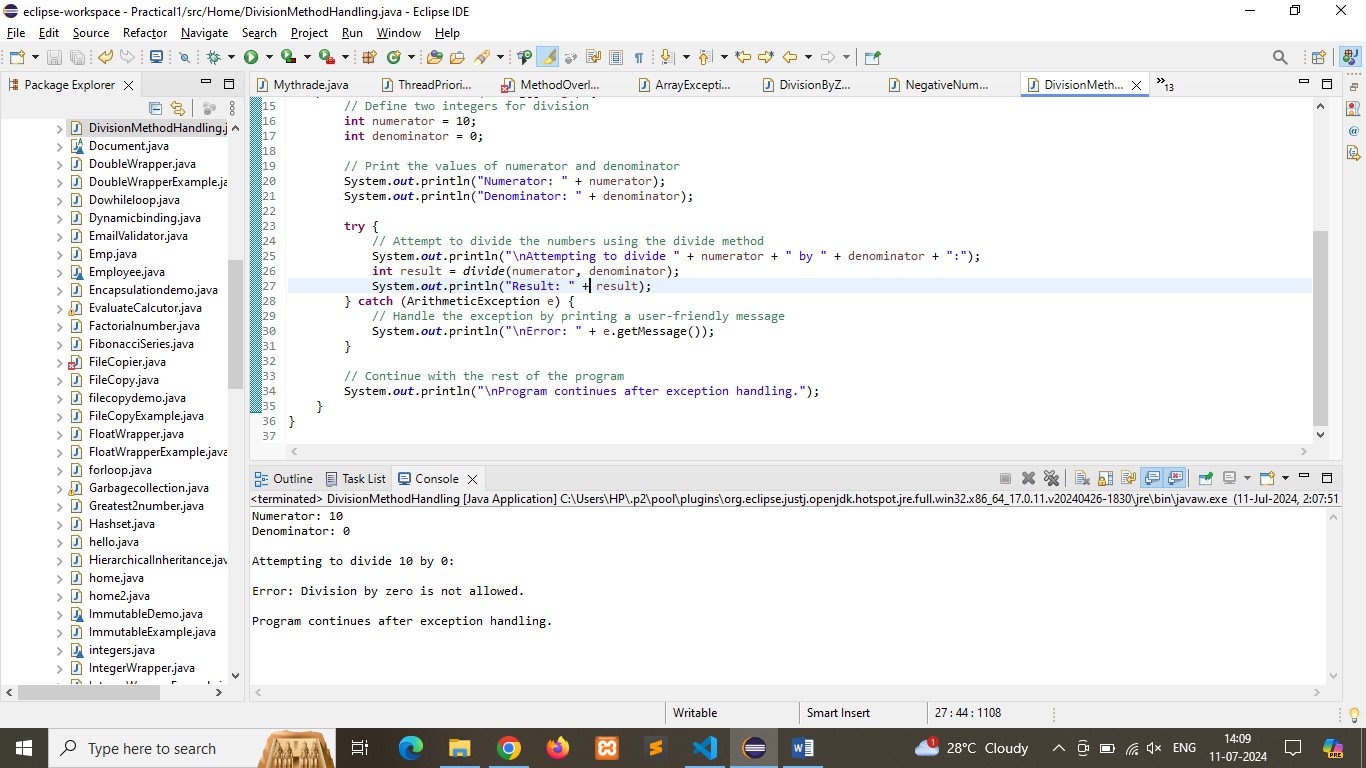
// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:



Q5) Define a custom exception called InvalidAgeException. Write a Java program that throws this exception if the age provided is less than 18. Handle the exception and display an appropriate message.

Program:

**package** Home;

**import** java.util.Scanner;

// Define the custom exception class

**class** InvalidAgeException **extends** Exception {

// Constructor that accepts a message

**public** InvalidAgeException(String message) {

**super**(message);

}

}

**public** **class** CustomExceptionHandling {

// Method to check age and throw InvalidAgeException if age is less than 18

**public** **static** **void** checkAge(**int** age) **throws** InvalidAgeException {

**if** (age < 18) {

// Throw an InvalidAgeException with a custom message

**throw** **new** InvalidAgeException("Age must be 18 or older.");

}

System.***out***.println("Age is valid: " + age);

}

**public** **static** **void** main(String[] args) {

// Create a Scanner object for reading input

Scanner scanner = **new** Scanner(System.***in***);

**try** {

// Prompt the user to enter their age

System.***out***.print("Enter your age: ");

**int** age = scanner.nextInt();

// Check the age using the checkAge method

*checkAge*(age);

} **catch** (InvalidAgeException e) {

// Handle the InvalidAgeException

System.***out***.println("\nError: " + e.getMessage());

} **catch** (Exception e) {

// Handle any other exceptions that may occur

System.***out***.println("\nError: Invalid input. Please enter a valid age.");

} **finally** {

// Close the scanner

scanner.close();

}

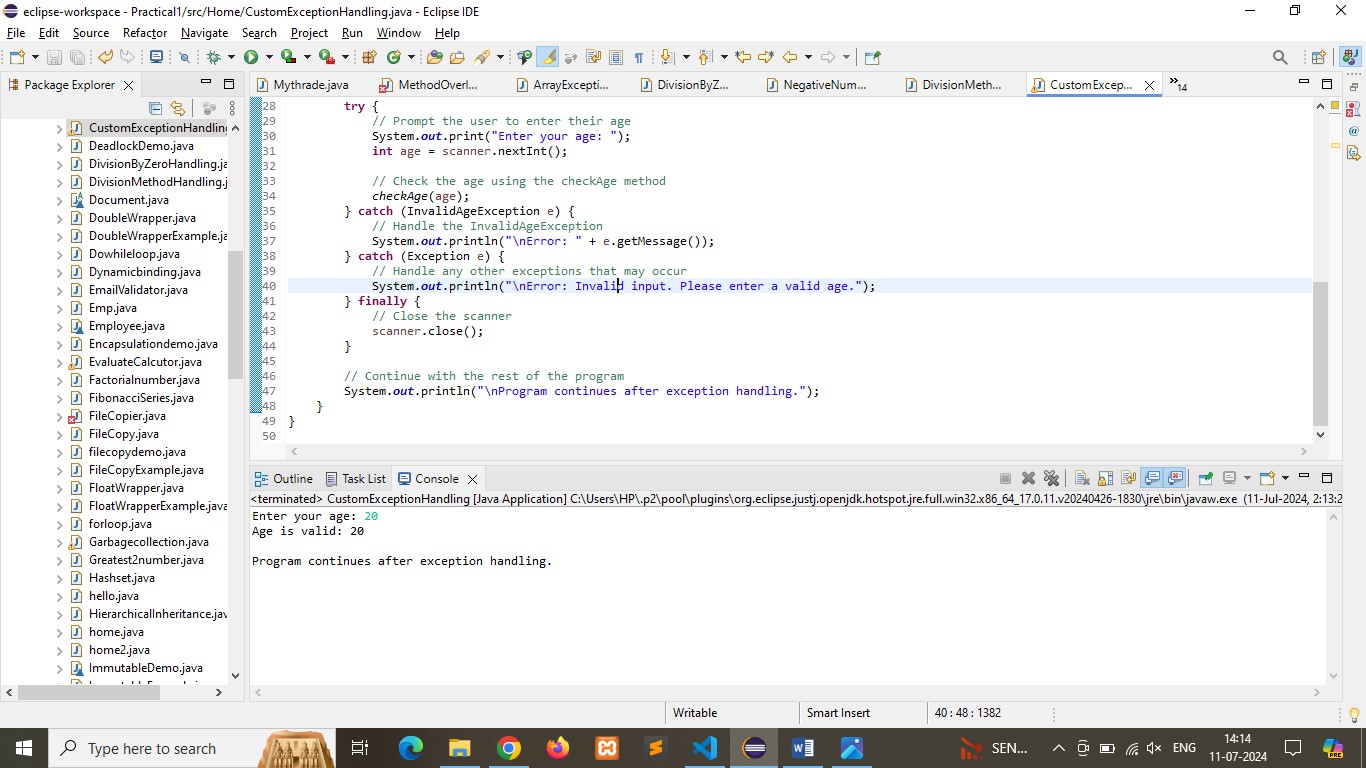
// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:



Q6) Write a Java program that has a method to validate a user's email address. The method should throw a custom exception InvalidEmailException if the email does not contain @ and .. Handle the exception in the main method.

Program:

**package** Home;

**import** java.util.Scanner;

// Define the custom exception class

**class** InvalidEmailException **extends** Exception {

// Constructor that accepts a message

**public** InvalidEmailException(String message) {

**super**(message);

}

}

**public** **class** EmailValidation {

// Method to validate the email address

**public** **static** **void** validateEmail(String email) **throws** InvalidEmailException {

// Check if the email contains "@" and "."

**if** (!email.contains("@") || !email.contains(".")) {

// Throw an InvalidEmailException with a custom message

**throw** **new** InvalidEmailException("Invalid email address. Email must contain '@' and '.'.");

}

System.***out***.println("Email is valid: " + email);

}

**public** **static** **void** main(String[] args) {

// Create a Scanner object for reading input

Scanner scanner = **new** Scanner(System.***in***);

**try** {

// Prompt the user to enter their email address

System.***out***.print("Enter your email address: ");

String email = scanner.nextLine();

// Validate the email using the validateEmail method

*validateEmail*(email);

} **catch** (InvalidEmailException e) {

// Handle the InvalidEmailException

System.***out***.println("\nError: " + e.getMessage());

} **catch** (Exception e) {

// Handle any other exceptions that may occur

System.***out***.println("\nError: Invalid input. Please enter a valid email address.");

} **finally** {

// Close the scanner

scanner.close();

}

// Continue with the rest of the program

System.***out***.println("\nProgram continues after exception handling.");

}

}

Output:

