Exception Handling

1. Create a Java program that acts as a simple calculator. ● The program should prompt the user to enter two numbers and an operator (+, -, *, /). ● Perform the corresponding calculation based on the operator. ● Handle potential exceptions, such as division by zero or invalid operator input. ● Display the result or an appropriate error message.

```
import java.util.Scanner;

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

        try {
            // Prompt user for input
            System.out.print("Enter the first number: ");
            double num1 = scanner.nextDouble();

            System.out.print("Enter the second number: ");
            double num2 = scanner.nextDouble();

            System.out.print("Enter the operator (+, -, *, /): ");
            char operator = scanner.next().charAt(0);

            // Perform calculation based on the operator
            double result;
            switch (operator) {
                case '+':
                    result = num1 + num2;
                    break;
                case '-':
                    result = num1 - num2;
                    break;
                case '*':
                    result = num1 * num2;
                    break;
                case '/':
                    // Handle division by zero
                    if (num2 == 0) {
                        System.out.println("Error: Division by zero is not allowed.");
                        return;
                    }
                    result = num1 / num2;
                    break;
                default:
                    System.out.println("Error: Invalid operator entered.");
                    return;
            }

            // Display the result
            System.out.println("Result: " + result);

        } catch (java.util.InputMismatchException e) {
            System.out.println("Error: Invalid input. Please enter valid numbers.");
        } finally {
            // Close the scanner
            scanner.close();
        }
    }
}
```

2. Write a Java program to simulate a simple banking application. ● Create a class BankAccount with a balance and methods for deposit and withdrawal. ● Implement exception handling for withdrawal operations to prevent overdrawing. ● Handle the scenario when the withdrawal amount is greater than the balance.

```
import java.util.Scanner;

class BankAccount {
    private double balance;

    public BankAccount(double initialBalance) {
        this.balance = initialBalance;
    }

    public double getBalance() {
        return balance;
    }

    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("Deposit successful. New balance: " + balance);
        } else {
            System.out.println("Error: Invalid deposit amount.");
        }
    }

    public void withdraw(double amount) {
        try {
            if (amount > 0) {
                if (amount <= balance) {
                    balance -= amount;
                    System.out.println("Withdrawal successful. New balance: " + balance);
                } else {
                    throw new InsufficientFundsException("Error: Insufficient funds. Cannot withdraw
more than the balance.");
                }
            } else {
                throw new IllegalArgumentException("Error: Invalid withdrawal amount.");
            }
        } catch (IllegalArgumentException | InsufficientFundsException e) {
            System.out.println(e.getMessage());
        }
    }
}

class InsufficientFundsException extends Exception {
    public InsufficientFundsException(String message) {
        super(message);
    }
}

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

        System.out.print("Enter initial balance: ");
        double initialBalance = scanner.nextDouble();

        BankAccount account = new BankAccount(initialBalance);

        while (true) {
            System.out.println("\n1. Deposit");
            System.out.println("2. Withdraw");
            System.out.println("3. Check Balance");
            System.out.println("4. Exit");
            System.out.print("Select an option: ");
            int choice = scanner.nextInt();

            switch (choice) {
                case 1:
                    System.out.print("Enter deposit amount: ");
                    double depositAmount = scanner.nextDouble();
                    account.deposit(depositAmount);
                    break;
                case 2:
                    System.out.print("Enter withdrawal amount: ");
                    double withdrawalAmount = scanner.nextDouble();
                    account.withdraw(withdrawalAmount);
                    break;
                case 3:
                    System.out.println("Current Balance: " + account.getBalance());
                    break;
                case 4:
                    System.out.println("Exiting the program. Thank you!");
                    scanner.close();
                    System.exit(0);
                default:
                    System.out.println("Invalid choice. Please enter a valid option.");
            }
        }
    }
}
```

By - Divya Parihar