JAVA PBL

Problem 1:

Write a program to input marks of students in 5 subjects, calculate total, average, and grade using methods and handle invalid marks using exception handling.

```
import java.util.Scanner;

public class StudentMarks {

  public static int calculateTotal(int[] marks) {
    int total = 0;
    for (int mark : marks) {
      total += mark;
    }
    return total;
  }


  public static double calculateAverage(int total, int subjects) {
    return (double) total / subjects;
  }

  public static String calculateGrade(double average) {
    if (average >= 90) return "A+";
    else if (average >= 80) return "A";
    else if (average >= 70) return "B";
    else if (average >= 60) return "C";
```

```
        else if (average >= 50) return "D";
        else return "F";
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] marks = new int[5];

        try {
            System.out.println("Enter marks for 5 subjects (0–100):");
            for (int i = 0; i < 5; i++) {
                System.out.print("Subject " + (i + 1) + ": ");
                marks[i] = sc.nextInt();

                if (marks[i] < 0 || marks[i] > 100) {
                    throw new IllegalArgumentException("Invalid mark entered! Marks must be between 0 and 100.");
                }
            }

            int total = calculateTotal(marks);
            double average = calculateAverage(total, marks.length);
            String grade = calculateGrade(average);

            System.out.println("\n--- Result ---");
            System.out.println("Total Marks: " + total);
            System.out.printf("Average: %.2f\n", average);
            System.out.println("Grade: " + grade);

        } catch (IllegalArgumentException e) {
            System.out.println("Error: " + e.getMessage());
        } catch (Exception e) {
            System.out.println("Error: Invalid input! Please enter numeric values for marks.");
        } finally {
            sc.close();
        }
    }
}
```

Output:

```
Enter marks for 5 subjects (0–100):
Subject 1: 85
Subject 2: 90
Subject 3: 78
Subject 4: 88
Subject 5: 92

--- Result ---
Total Marks: 433
Average: 86.60
Grade: A
```

Problem 2:

Accept item names, price, and quantity. Calculate total, apply a discount if total > 2000, and display formatted bill using methods.

```
import java.util.Scanner;
```

```java
public class SimpleBill {
    public static double getTotal(double price, int quantity) {
        return price * quantity;
    }


    public static double getDiscountedTotal(double total) {
        if (total > 2000) {
            total = total - (total * 0.10); // 10% discount
            System.out.println("10% discount applied!");
        }
        return total;
    }

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

        try {
            System.out.print("Enter item name: ");
            String name = sc.nextLine();

            System.out.print("Enter price: ");
            double price = sc.nextDouble();

            System.out.print("Enter quantity: ");
            int quantity = sc.nextInt();

            double total = getTotal(price, quantity);
            total = getDiscountedTotal(total);
            System.out.println("\n--- BILL ---");
            System.out.println("Item Name : " + name);
            System.out.println("Price     : " + price);
            System.out.println("Quantity  : " + quantity);
            System.out.println("Total     : " + total);

        } catch (Exception e) {
            System.out.println("Error: Please enter valid input!");
        }

        sc.close();
    }
}
```

Output:

```
Enter item name: Shoes
Enter price: 1500
Enter quantity: 2
10% discount applied!

--- BILL ---
Item Name : Shoes
Price     : 1500.0
Quantity  : 2
Total     : 2700.0
```

Problem 3:

Take a sentence and count the number of words and occurrences of a specific word using arrays and string methods.

```
import java.util.Scanner;
```

```
public class WordCount {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);


        System.out.print("Enter a sentence: ");
        String sentence = sc.nextLine();


        String[] words = sentence.split(" ");
        int totalWords = words.length;




        System.out.print("Enter a word to find its occurrence: ");
        String searchWord = sc.next();


        int count = 0;




        for (String w : words) {
            if (w.equalsIgnoreCase(searchWord)) {
                count++;
            }
        }


        System.out.println("\n--- RESULT ---");
        System.out.println("Total number of words: " + totalWords);
        System.out.println("Occurrences of '" + searchWord + "': " + count);


        sc.close();
    }
}
```

Output:

```
Enter a sentence: Java is fun and Java is powerful

Enter a word to find its occurrence: Java


--- RESULT ---

Total number of words: 7

Occurrences of 'Java': 2
```

Problem 4:

Check password strength: Length ≥ 8, contains uppercase, lowercase, digit, and symbol, throw exception if invalid.

```
import java.util.Scanner;


public class PasswordCheck {


    public static void checkPassword(String password) throws Exception {

        if (password.length() < 8)

            throw new Exception("Password must be at least 8 characters long!");


        boolean hasUpper = false, hasLower = false, hasDigit = false, hasSymbol = false;


        for (char ch : password.toCharArray()) {

            if (Character.isUpperCase(ch))

                hasUpper = true;

            else if (Character.isLowerCase(ch))

                hasLower = true;

            else if (Character.isDigit(ch))

                hasDigit = true;

            else
```

```
                hasSymbol = true;
            }

            if (!hasUpper)
                throw new Exception("Password must contain at least one uppercase letter!");
            if (!hasLower)
                throw new Exception("Password must contain at least one lowercase letter!");
            if (!hasDigit)
                throw new Exception("Password must contain at least one digit!");
            if (!hasSymbol)
                throw new Exception("Password must contain at least one symbol!");

            System.out.println("✅ Password is strong!");
        }

        public static void main(String[] args) {
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter your password: ");
            String password = sc.nextLine();

            try {
                checkPassword(password);
            } catch (Exception e) {
                System.out.println("❌ Error: " + e.getMessage());
            }

            sc.close();
        }
}
```

Output:

```
Case 1 – Strong password

Enter your password: Abc@1234

✅ Password is strong!

Case 2 – Weak password

Enter your password: abc123

❌ Error: Password must be at least 8 characters long!
```

Problem 5:

Simulate ATM operations like deposit, withdraw, and check balance. Use methods for each operation and handle insufficient balance with exception handling.

```java
import java.util.Scanner;


public class SimpleATMProgram {
    static double balance = 1000; // starting balance


    public static void checkBalance() {
        System.out.println("Current Balance: ₹" + balance);
    }


    public static void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("₹" + amount + " deposited successfully.");
        } else {
            System.out.println("Invalid amount! Deposit must be greater than 0.");
        }
    }


    public static void withdraw(double amount) {
        try {
            if (amount <= 0) {
                throw new Exception("Withdrawal amount must be greater than 0!");
            }
            if (amount > balance) {
                throw new Exception("Insufficient balance!");
            }
            balance -= amount;
            System.out.println("₹" + amount + " withdrawn successfully.");
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int choice;

        do {
            System.out.println("\n===== ATM MENU =====");
            System.out.println("1. Check Balance");
            System.out.println("2. Deposit");
            System.out.println("3. Withdraw");
            System.out.println("4. Exit");
            System.out.print("Enter your choice: ");
            choice = sc.nextInt();

            switch (choice) {
                case 1:
                    checkBalance();
                    break;
                case 2:
                    System.out.print("Enter amount to deposit: ");
                    double dep = sc.nextDouble();
                    deposit(dep);
                    break;
                case 3:
                    System.out.print("Enter amount to withdraw: ");
                    double wd = sc.nextDouble();
                    withdraw(wd);
                    break;
                case 4:
                    System.out.println("Thank you for using the ATM!");
                    break;
                default:
                    System.out.println("Invalid choice! Please try again.");
            }
        } while (choice != 4);

        sc.close();
    }
}
```

Output:

```
===== ATM MENU =====
1. Check Balance
2. Deposit
3. Withdraw
4. Exit
Enter your choice: 2
Enter amount to deposit: 500
₹500.0 deposited successfully.

===== ATM MENU =====
1. Check Balance
2. Deposit
3. Withdraw
4. Exit
Enter your choice: 3
Enter amount to withdraw: 2000
Error: Insufficient balance!

===== ATM MENU =====
1. Check Balance
2. Deposit
3. Withdraw
4. Exit
Enter your choice: 4
Thank you for using the ATM!
```

Problem 6:

Accept basic salary and compute HRA, DA, PF, and gross salary. Display results using methods and handle invalid inputs with exceptions.

```
import java.util.Scanner;

public class SalaryCalculator {



    public static double calculateHRA(double basic) {
        return 0.20 * basic;
```

```
}

public static double calculateDA(double basic) {
    return 0.10 * basic;
}

public static double calculatePF(double basic) {
    return 0.08 * basic;
}


public static double calculateGross(double basic, double hra, double da, double pf) {
    return basic + hra + da - pf;
}

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

    try {
        System.out.print("Enter Basic Salary: ");
        double basic = sc.nextDouble();

        if (basic <= 0) {
            throw new Exception("Basic salary must be greater than 0!");
        }

        double hra = calculateHRA(basic);
        double da = calculateDA(basic);
        double pf = calculatePF(basic);
        double gross = calculateGross(basic, hra, da, pf);
```

```
            System.out.println("\n--- SALARY DETAILS ---");
            System.out.println("Basic Salary : ₹" + basic);
            System.out.println("HRA (20%)    : ₹" + hra);
            System.out.println("DA (10%)     : ₹" + da);
            System.out.println("PF (8%)      : ₹" + pf);
            System.out.println("--------------------------");
            System.out.println("Gross Salary : ₹" + gross);

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

Output:

```
Enter Basic Salary: 30000

--- SALARY DETAILS ---
Basic Salary : ₹30000.0
HRA (20%)    : ₹6000.0
DA (10%)     : ₹3000.0
PF (8%)      : ₹2400.0
--------------------------
Gross Salary : ₹36600.0
```

Problem 7:

Accept total bill and membership type (Silver/Gold/Platinum) and apply discounts accordingly using if-else and methods.

```java
import java.util.Scanner;

public class MembershipDiscount {

    public static double applyDiscount(double total, String membership) {
        double discount = 0;

        if (membership.equalsIgnoreCase("Silver")) {
            discount = 0.05; // 5% discount
        } else if (membership.equalsIgnoreCase("Gold")) {
            discount = 0.10; // 10% discount
        } else if (membership.equalsIgnoreCase("Platinum")) {
            discount = 0.15; // 15% discount
        } else {
            System.out.println("Invalid membership type! No discount applied.");
        }

        return total - (total * discount);
    }

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

        try {
            System.out.print("Enter total bill amount: ");
            double totalBill = sc.nextDouble();

            if (totalBill <= 0) {
                throw new Exception("Total bill must be greater than 0!");
            }
            sc.nextLine(); // consume newline
            System.out.print("Enter membership type (Silver/Gold/Platinum): ");
            String membership = sc.nextLine();

            double finalAmount = applyDiscount(totalBill, membership);

            System.out.println("\n--- BILL DETAILS ---");
            System.out.println("Original Bill : ₹" + totalBill);
            System.out.println("Membership    : " + membership);
            System.out.println("Final Amount  : ₹" + finalAmount);

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

Output:

```
Enter total bill amount: 2000
Enter membership type (Silver/Gold/Platinum): Gold

--- BILL DETAILS ---
Original Bill : ₹2000.0
Membership    : Gold
Final Amount  : ₹1800.0
```

Problem 8:

For ‘n’ products, store product name, price, and quantity in arrays. Calculate total stock value and handle out-of-stock errors via exception handling.

```
import java.util.Scanner;


public class ProductStock {


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


        try {
            System.out.print("Enter number of products: ");
            int n = sc.nextInt();
            sc.nextLine(); // consume newline


            String[] productNames = new String[n];
            double[] prices = new double[n];
            int[] quantities = new int[n];


            for (int i = 0; i < n; i++) {
                System.out.print("\nEnter name of product " + (i + 1) + ": ");
                productNames[i] = sc.nextLine();


                System.out.print("Enter price of " + productNames[i] + ": ");
                prices[i] = sc.nextDouble();


                System.out.print("Enter quantity of " + productNames[i] + ": ");
                quantities[i] = sc.nextInt();
                sc.nextLine(); // consume newline


                if (quantities[i] <= 0) {
                    throw new Exception("Product '" + productNames[i] + "' is out of stock!");
                }
            }

            double totalValue = 0;
            for (int i = 0; i < n; i++) {
                totalValue += prices[i] * quantities[i];
            }

            System.out.println("\n--- STOCK DETAILS ---");
            for (int i = 0; i < n; i++) {
                System.out.println(productNames[i] + " | Price: ₹" + prices[i] + " | Quantity: " + quantities[i]);
            }
            System.out.println("---------------------------");
            System.out.println("Total Stock Value: ₹" + totalValue);

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

Output:

```
Enter number of products: 3


Enter name of product 1: Pen
Enter price of Pen: 10
Enter quantity of Pen: 50
Enter name of product 2: Notebook
Enter price of Notebook: 50
Enter quantity of Notebook: 0
Error: Product 'Notebook' is out of stock!
```

Problem 1:

Process a coffee order: take customer size choice, calculate total price based on size and add-ons, and handle a list of 5 drink types.

```
import java.util.Scanner;

public class CoffeeOrder {

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


        String[] drinks = {"Espresso", "Latte", "Cappuccino", "Mocha", "Americano"};
        double[] basePrices = {100, 120, 130, 140, 110}; // base prices for each drink

        System.out.println("Available drinks:");
        for (int i = 0; i < drinks.length; i++) {
            System.out.println((i + 1) + ". " + drinks[i] + " - ₹" + basePrices[i]);
        }

        try {

            System.out.print("\nEnter drink number (1-5): ");
            int drinkChoice = sc.nextInt();
```

```
if (drinkChoice < 1 || drinkChoice > 5) {
    throw new Exception("Invalid drink choice!");
}

String selectedDrink = drinks[drinkChoice - 1];
double price = basePrices[drinkChoice - 1];


System.out.println("Choose size: 1. Small (+₹0) 2. Medium (+₹20) 3. Large (+₹40)");
System.out.print("Enter size number: ");
int sizeChoice = sc.nextInt();

switch (sizeChoice) {
    case 1: price += 0; break;
    case 2: price += 20; break;
    case 3: price += 40; break;
    default: throw new Exception("Invalid size choice!");
}

sc.nextLine(); // consume newline


System.out.print("Do you want extra milk? (yes/no): ");
String milk = sc.nextLine();
if (milk.equalsIgnoreCase("yes")) price += 10;

System.out.print("Do you want extra sugar? (yes/no): ");
String sugar = sc.nextLine();
if (sugar.equalsIgnoreCase("yes")) price += 5;
```

```
            System.out.println("\n--- ORDER SUMMARY ---");
            System.out.println("Drink: " + selectedDrink);
            System.out.println("Size: " + (sizeChoice == 1 ? "Small" : sizeChoice == 2 ? "Medium" : "Large"));
            System.out.println("Extra Milk: " + milk);
            System.out.println("Extra Sugar: " + sugar);
            System.out.println("Total Price: ₹" + price);

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

Output:

```
Available drinks:
1. Espresso - ₹100
2. Latte - ₹120
3. Cappuccino - ₹130
4. Mocha - ₹140
5. Americano - ₹110

Enter drink number (1-5): 2
Choose size: 1. Small (+₹0) 2. Medium (+₹20) 3. Large (+₹40)
Enter size number: 3
Do you want extra milk? (yes/no): yes
Do you want extra sugar? (yes/no): no

--- ORDER SUMMARY ---
```

```
Drink: Latte
Size: Large
Extra Milk: yes
Extra Sugar: no
Total Price: ₹170
```

Problem 2:

Create a method that accepts two numbers and an operation symbol. Use a switch to perform and return the result of addition, subtraction, multiplication, or division.

```
import java.util.Scanner;

public class Calculator {

    public static double calculate(double num1, double num2, char op) throws Exception {
        double result;
        switch (op) {
            case '+':
                result = num1 + num2;
                break;
            case '-':
                result = num1 - num2;
                break;
            case '*':
                result = num1 * num2;
                break;
            case '/':
                if (num2 == 0) throw new Exception("Cannot divide by zero!");
                result = num1 / num2;
                break;
            default:
                throw new Exception("Invalid operation!");
        }
        return result;
    }

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

        try {
            System.out.print("Enter first number: ");
            double num1 = sc.nextDouble();

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

            System.out.print("Enter operation (+, -, *, /): ");
            char op = sc.next().charAt(0);

            double result = calculate(num1, num2, op);
            System.out.println("Result: " + result);

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }

        sc.close();
    }
}
```

Output:

```
Enter first number: 10
Enter second number: 5
Enter operation (+, -, *, /): *
Result: 50.0
```

Problem 3:

Input a string and count vowels, consonants, digits, and special characters using loops and conditionals.

```
import java.util.Scanner;

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

        System.out.print("Enter a string: ");
        String str = sc.nextLine();

        int vowels = 0, consonants = 0, digits = 0, special = 0;

        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);

            if (Character.isLetter(ch)) {
                char lower = Character.toLowerCase(ch);
                if (lower == 'a' || lower == 'e' || lower == 'i' || lower == 'o' || lower == 'u') {
                    vowels++;
                } else {
                    consonants++;
                }
            } else if (Character.isDigit(ch)) {
                digits++;
            } else if (!Character.isWhitespace(ch)) {
                special++;
            }
        }

        System.out.println("\n--- COUNT ---");
        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
        System.out.println("Digits: " + digits);
        System.out.println("Special Characters: " + special);

        sc.close();
    }
}
```

Output:

```
Enter a string: Hello World! 123

--- COUNT ---
Vowels: 3
Consonants: 7
Digits: 3
Special Characters: 1
```

Problem 4:

For n customers, input name, account type, and balance. Apply 4% interest for savings and 6% for fixed accounts, then display updated balances.

```
import java.util.Scanner;
```

```
public class SimpleCustomerInterest {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter number of customers: ");
        int n = sc.nextInt();
        sc.nextLine(); // consume newline

        String[] names = new String[n];
        String[] types = new String[n];
        double[] balances = new double[n];

        for (int i = 0; i < n; i++) {
            System.out.print("\nEnter name of customer " + (i + 1) + ": ");
            names[i] = sc.nextLine();

            System.out.print("Enter account type (Savings/Fixed): ");
            types[i] = sc.nextLine();

            System.out.print("Enter balance: ");
            balances[i] = sc.nextDouble();
            sc.nextLine(); // consume newline


            if (types[i].equalsIgnoreCase("Savings")) {
                balances[i] += balances[i] * 0.04; // 4% interest
            } else if (types[i].equalsIgnoreCase("Fixed")) {
                balances[i] += balances[i] * 0.06; // 6% interest
            }
        }
        System.out.println("\n--- UPDATED BALANCES ---");
        for (int i = 0; i < n; i++) {
            System.out.println(names[i] + " | " + types[i] + " | Balance: ₹" + balances[i]);
        }

        sc.close();
    }
}
```

Output:

```
Enter number of customers: 2

Enter name of customer 1: Alice
Enter account type (Savings/Fixed): Savings
Enter balance: 10000

Enter name of customer 2: Bob
Enter account type (Savings/Fixed): Fixed
Enter balance: 20000

--- UPDATED BALANCES ---
Alice | Savings | Balance: ₹10400.0
Bob | Fixed | Balance: ₹21200.0
```

Problem 5:

Read 5 daily temperatures into an array. Use a loop and a method to convert each temperature from Celsius to Fahrenheit, displaying both.

```
import java.util.Scanner;
```

```
public class TemperatureConverter {

    public static double celsiusToFahrenheit(double celsius) {
        return (celsius * 9 / 5) + 32;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        double[] celsiusTemps = new double[5];

        for (int i = 0; i < 5; i++) {
            System.out.print("Enter temperature for day " + (i + 1) + " in Celsius: ");
            celsiusTemps[i] = sc.nextDouble();
        }

        System.out.println("\n--- TEMPERATURES ---");
        for (int i = 0; i < 5; i++) {
            double fahrenheit = celsiusToFahrenheit(celsiusTemps[i]);
            System.out.printf("Day %d: %.2f°C = %.2f°F%n", i + 1, celsiusTemps[i], fahrenheit);
        }

        sc.close();
    }
}
```

Output:

```
Enter temperature for day 1 in Celsius: 25
Enter temperature for day 2 in Celsius: 30
Enter temperature for day 3 in Celsius: 28
Enter temperature for day 4 in Celsius: 22
Enter temperature for day 5 in Celsius: 26

--- TEMPERATURES ---
Day 1: 25.00°C = 77.00°F
Day 2: 30.00°C = 86.00°F
Day 3: 28.00°C = 82.40°F
Day 4: 22.00°C = 71.60°F
Day 5: 26.00°C = 78.80°F
```

Problem 6:

Accept number of units consumed and calculate bill based on slab rates using conditionals and methods.

```java
import java.util.Scanner;

public class ElectricityBill {


  public static double calculateBill(int units) {
    double bill = 0;

    if (units <= 100) {
      bill = units * 5; // ₹5 per unit
    } else if (units <= 200) {
      bill = 100 * 5 + (units - 100) * 7; // First 100 at 5, next at 7
    } else if (units <= 300) {
      bill = 100 * 5 + 100 * 7 + (units - 200) * 10; // Next 100 at 10
    } else {
      bill = 100 * 5 + 100 * 7 + 100 * 10 + (units - 300) * 15; // Above 300 at 15
    }
```

```
        return bill;
    }

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

        System.out.print("Enter number of units consumed: ");
        int units = sc.nextInt();

        if (units < 0) {
            System.out.println("Invalid input! Units cannot be negative.");
        } else {
            double totalBill = calculateBill(units);
            System.out.println("Total electricity bill: ₹" + totalBill);
        }

        sc.close();
    }
}
```

Output:

```
Enter number of units consumed: 250
Total electricity bill: ₹1850.0
```

Problem 7:

Input a string and check if it’s a palindrome (ignore case and spaces). Use string methods and exception handling.

```
import java.util.Scanner;
```

```
public class PalindromeChecker {

    public static boolean isPalindrome(String str) {
        str = str.replaceAll("\\s+", "").toLowerCase(); // remove spaces and convert to lowercase
        String reversed = new StringBuilder(str).reverse().toString();
        return str.equals(reversed);
    }

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

        try {
            System.out.print("Enter a string: ");
            String input = sc.nextLine();

            if (input.isEmpty()) {
                throw new Exception("Input cannot be empty!");
            }

            if (isPalindrome(input)) {
                System.out.println("✅ The string is a palindrome.");
            } else {
                System.out.println("❌ The string is not a palindrome.");
            }

        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
        sc.close();
    }
}
```

Output:

```
Enter a string: A man a plan a canal Panama

✅ The string is a palindrome.

Enter a string: Hello World

❌ The string is not a palindrome.
```

Problem 8:

Read a word (String). Use a loop and a switch on each character to replace 'a' with '4', 'e' with '3', and 'o' with '0'.

```
import java.util.Scanner;


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


        System.out.print("Enter a word: ");
        String word = sc.nextLine();
        StringBuilder result = new StringBuilder();


        for (int i = 0; i < word.length(); i++) {
            char ch = word.charAt(i);


            switch (Character.toLowerCase(ch)) {
                case 'a':
```

```
          result.append('4');
          break;
        case 'e':
          result.append('3');
          break;
        case 'o':
          result.append('0');
          break;
        default:
          result.append(ch);
      }
    }

    System.out.println("Converted word: " + result.toString());
    sc.close();
  }
}
```

Output:

```
Enter a word: apple
Converted word: 4ppl3
Enter a word: chocolate
Converted word: ch0c0l4t3
```