

**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.*;
public class CoffeeOrder {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String[] drinks = {"Espresso", "Latte", "Cappuccino", "Mocha", "Americano"};
        System.out.println("Available drinks: " + Arrays.toString(drinks));
        System.out.print("Enter drink name: ");
        String drink = sc.nextLine();
        System.out.print("Enter size (S/M/L): ");
        char size = sc.next().charAt(0);
        double price = switch (size) {
            case 'S' -> 100;
            case 'M' -> 150;
            case 'L' -> 200;
            default -> 0;
        };
        System.out.print("Add extra shot? (yes/no): ");
        String add = sc.next();
        if (add.equalsIgnoreCase("yes")) price += 30;
        System.out.println("Total for " + drink + ": ₹" + price);
    }
}
```

The program takes the customer's coffee choice and cup size (S/M/L) as input.

A switch statement is used to assign the base price according to the chosen size.

The user is also asked if they want an extra shot, and if yes, ₹30 is added to the total.

The program then calculates the total cost and displays the final bill with the drink name.

This program demonstrates use of arrays, conditional statements, and user input handling in Java.

**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.*;
public class Calculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        double a = sc.nextDouble();
        System.out.print("Enter second number: ");
        double b = sc.nextDouble();
        System.out.print("Enter operation (+,-,*,/): ");
        char op = sc.next().charAt(0);
        double result = switch (op) {
            case '+' -> a + b;
            case '-' -> a - b;
            case '*' -> a * b;
            case '/' -> b != 0 ? a / b : 0;
            default -> 0;
        };
        System.out.println("Result: " + result);
    }
}
```

```
}
```

1. The program accepts two numbers and an operation symbol (+, -, \*, /) from the user.
2. A **switch** statement is used to perform the selected arithmetic operation.
3. Each case performs one operation and stores the result in a variable.
4. Division is handled safely by checking if the divisor is zero to avoid runtime errors.
5. The final result is displayed on the screen.
6. This demonstrates decision-making and the use of a **switch** block for multiple choices.

**Problem 3: Input a string and count vowels, consonants, digits, and special characters using loops and conditionals.**

```
import java.util.*;
public class StringCount {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String s = sc.nextLine().toLowerCase();
        int v=0,c=0,d=0,sp=0;
        for (char ch : s.toCharArray()) {
            if ("aeiou".indexOf(ch) != -1) v++;
            else if (Character.isLetter(ch)) c++;
            else if (Character.isDigit(ch)) d++;
            else sp++;
        }
        System.out.println("Vowels="+v+" Consonants="+c+" Digits="+d+" Specials="+sp);
    }
}
```

1. The user enters a string which is converted to lowercase for easier checking
2. The program then loops through each character in the string.
3. Using conditions, it checks whether each character is a vowel, consonant, digit, or special character.
4. The counts for each category are stored in separate counters.
5. At the end, the program displays the number of vowels, consonants, digits, and special characters.
6. This shows the use of loops, conditionals, and string manipulation in Java.

**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.*;
public class BankInterest {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of customers: ");
        int n = sc.nextInt();
        for (int i=1; i<=n; i++) {
            System.out.print("Name: ");
            String name = sc.next();
            System.out.print("Account type (savings/fixed): ");
            String type = sc.next();
            System.out.print("Balance: ");
```

```

        double bal = sc.nextDouble();
        double rate = type.equalsIgnoreCase("savings") ? 0.04 : 0.06;
        bal += bal * rate;
        System.out.println(name + "'s updated balance: " + bal);
    }
}

```

The program first asks how many customers' details will be entered.

For each customer, it takes their name, account type, and current balance.

Depending on the account type, it applies an interest rate — 4% for savings and 6% for fixed accounts.

The updated balance is calculated by adding the interest amount to the existing balance.

Finally, it displays the customer's name and updated balance.

This problem demonstrates use of loops, conditional logic, and percentage calculations.

**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.*;
public class TempConvert {
    static double toF(double c){ return (c * 9/5) + 32; }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        double[] temp = new double[5];
        for(int i=0;i<5;i++){
            System.out.print("Enter temp " + (i+1) + " in Celsius: ");
            temp[i] = sc.nextDouble();
        }
        for(double c: temp)
            System.out.println(c + "°C = " + toF(c) + "°F");
    }
}

```

The program reads 5 temperature values (in Celsius) from the user and stores them in an array.

It uses a separate method `toF()` that converts Celsius to Fahrenheit using the formula  $(C \times 9/5) + 32$ .

A loop is used to call this method for each element of the array.

The program then prints both the original Celsius value and its Fahrenheit equivalent.

This program demonstrates the use of arrays, methods, loops, and basic mathematical formulas in Java.

**Problem 6: Accept number of units consumed and calculate bill based on slab rates using conditionals and methods.**

```

import java.util.*;
public class ElectricityBill {
    static double calcBill(int units){
        if(units<=100) return units*1.5;
        else if(units<=200) return 100*1.5 + (units-100)*2.5;
        else return 100*1.5 + 100*2.5 + (units-200)*3.5;
    }
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter units consumed: ");
        int units = sc.nextInt();
    }
}

```

```

        System.out.println("Total Bill: ₹" + calcBill(units));
    }
}

```

The program reads the number of electricity units consumed by the user.

It defines a method `calcBill()` which calculates the bill amount based on slab rates.

- For 0–100 units: ₹1.5/unit
- For 101–200 units: ₹2.5/unit
- Above 200 units: ₹3.5/unit

The `if-else` structure is used to check the correct slab and compute the total amount.

The method is called in the main program and the total bill is displayed.

This shows the use of methods, conditional statements, and modular programming.

### **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.*;
public class PalindromeCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        try {
            System.out.print("Enter a string: ");
            String s = sc.nextLine().replaceAll(" ", "").toLowerCase();
            String rev = new StringBuilder(s).reverse().toString();
            System.out.println(s.equals(rev) ? "Palindrome" : "Not Palindrome");
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

The program takes a string as input from the user.

It removes spaces and converts all letters to lowercase to ignore case and space differences.

A `StringBuilder` is used to reverse the string easily.

The reversed string is compared with the original string to check if it's a palindrome.

The `try-catch` block handles any possible input exceptions to avoid errors.

Finally, it prints whether the string is a palindrome or not.

This problem demonstrates string handling, exception handling, and logical comparison in Java.

### **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.*;
public class ReplaceChar {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a word: ");
        String s = sc.nextLine().toLowerCase();
        StringBuilder res = new StringBuilder();
        for(char ch : s.toCharArray()){
            switch(ch){
                case 'a' -> res.append('4');
                case 'e' -> res.append('3');
                case 'o' -> res.append('0');
            }
        }
    }
}

```

```
        default -> res.append(ch);
    }
}
System.out.println("Converted word: " + res);
}
}
```

1. The user enters a word which is converted to lowercase.
2. The program loops through each character of the word.
3. A **switch** statement is used to replace specific characters:
  - 'a' → '4'
  - 'e' → '3'
  - 'o' → '0'
4. Other characters remain unchanged.
5. The modified characters are stored using **StringBuilder** and printed as the final converted word.
6. This program shows the use of loops, switch-case statements, and string building in Java.