CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

DEVANG PATEL INSTITUTE OF ADVANCE TECHNOLOGY & RESEARCH

Department of Computer Engineering

Subject Name: Java programming
Semester: III
Subject Code: CSE201
Academic year: 2024-25

Part - 1

No.	Aim of the Practical
2.	Imagine you are developing a simple banking application where you need to display the current balance of a user account. For simplicity, let's say the current balance is $20. Write a java program to store this balance in a variable and then display it to the user..

PROGRAM CODE:

```
import java.lang.*;
import java.util.Scanner;

class bank
{
public static void main(String []args)
{
Scanner s=new Scanner (System.in);
System.out.println("enter balance");
int b=s.nextInt();
System.out.println("balance is" + b);
 System.out.println(" 23DCS017 Khushi Dadhaniya");
}
}
```

OUTPUT:

```
D:\java>javac bank.java

D:\java>java bank
enter balance
5
balance is5

D:\java>
```

CONCLUSION:

- The System.out.println() statement displays the current balance to the user.

3. Write a program to take the user for a distance (in meters) and the time taken (as three numbers: hours, minutes, seconds), and display the speed, in meters per second, kilometers per hour and miles per hour (hint:1 mile = 1609 meters).

PROGRAM CODE:

```
import java.lang.*;
import java.util.Scanner;

class speed
{

        public static void main(String []args)
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("enter the distence in meter");
                float d=sc.nextFloat();

                System.out.println("enter the time(in hour,min.,sec.)");
                float h=sc.nextFloat();
                float m=sc.nextFloat();
                float s=sc.nextFloat();

                float time=(h*3600)+(m*60)+s;
                System.out.println("time is "+time);
                float speed =d/time;
                System.out.println("speed is " +speed);
                float kph=(d/1000)/(time/3600);
```

```
                System.out.println("Your speed in km/h is "+kph);

                float mph=(kph*1000)/1906;
                System.out.println("Your speed in miles/h is "+mph);
                System.out.println(" 23DCS017 Khushi Dadhaniya");
        }
}
```

OUTPUT:

```
D:\java>javac speed.java

D:\java>java speed
enter the distence in meter
1000
enter the time(in hour,min.,sec.)
1
2
3
time is 3723.0
speed is 0.26860058
Your speed in km/h is 0.9669621
Your speed in miles/h is 0.50732535
 23DCS017 Khushi Dadhaniya

D:\java>
```

CONCLUSION:

- We take the user's input for distance (in meters), hours, minutes, and seconds.
- We calculate the total time in seconds.
- We compute the speed in meters per second, kilometers per hour, and miles per hour using the given formulas.

4. Imagine you are developing a budget tracking application. You need to calculate the total expenses for the month. Users will input their daily expenses, and the program should compute the sum of these expenses. Write a Java program to calculate the sum of elements in an array representing daily expenses.

PROGRAM CODE:

```
import java.lang.*;
```

```
import java.util.Scanner;

class Expensetracker
{
public static void main(String []args)
{
        int i,n;
        float sum=0;
float[] a = new float[50];
Scanner s=new Scanner(System.in);
System.out.print("Enter the number of element in the array ");
 n=s.nextInt();


for ( i = 0; i < n; i++)
      {
         System.out.print("Enter array element " + (i + 1) + ": ");
         a[i] = s.nextFloat();
               }
      for (i=0;i<n;i++)
               {
          sum=sum+a[i];
      }

      System.out.println("Total expenses for the month " + sum);
 System.out.println(" 23DCS017 Khushi Dadhaniya");



      }

}
```

OUTPUT:

```
D:\java>javac Expensetracker.java

D:\java>java Expensetracker
Enter the number of element in the array 5
Enter array element 1: 10
Enter array element 2: 20
Enter array element 3: 30
Enter array element 4: 40
Enter array element 5: 50
Total expenses for the month 150.0
 23DCS017 Khushi Dadhaniya

D:\java>
```

CONCLUSION:

This program efficiently calculates the total expenses for a month based on daily inputs using an array. It demonstrates basic array handling and iteration in Java. Depending on your application's needs, you might expand it to include functionalities like user input, error handling, or integration with a larger application framework.

Supplementary Experiment:

You are creating a library management system. The library has two separate lists of books for fiction and non-fiction. The system should merge these lists into a single list for inventory purposes. Write a Java program to merge two arrays.

PROGRAM CODE:

```
public class Mergelib {
    public static void main(String[] args) {
        int[] arr1 = { 1, 3, 4, 5 };
        int[] arr2 = { 2, 4, 6, 8 };

        int l1 = arr1.length;
        int l2 = arr2.length;
        int result = l1 + l2;

        int[] mergearray = new int[result];

        for (int i = 0; i < l1; i++) {
            mergearray[i] = arr1[i];
        }

        for (int i = 0; i < l2; i++) {
            mergearray[l1 + i] = arr2[i];
        }

        for (int num : mergearray) {
            System.out.print(num + " ");
        }
    }
}
```

OUTPUT:

```
D:\java>javac Mergelib.java

D:\java>java Mergelib
1 3 4 5 2 4 6 8
D:\java>
```

CONCLUSION:

5. An electric appliance shop assigns code 1 to motor,2 to fan,3 to tube and 4 for wires. All other items have code 5 or more. While selling the goods, a sales tax of 8% to motor,12% to fan,5% to tube light,7.5% to wires and 3% for all other items is charged. A list containing the product code and price in two different arrays. Write a java program using switch statement to prepare the bill.

PROGRAM CODE:

```
import java.lang.*;
import java.util.Scanner;

class Bill
{
public static void main(String []args)
{
Scanner s=new Scanner(System.in);
        float motorTax = 0.08f;
        float fanTax = 0.12f;
        float tubeTax = 0.05f;
        float wiresTax = 0.075f;
        float otherItemTax = 0.03f;
                float sum=0;

                System.out.print("Enter 1 for motor, 2 for fan, 3 for tube, 4 for wires:
");
        int code=s.nextInt();
                System.out.print("Enter price of the item: ");
        float price = s.nextFloat();
                switch(code)
                {
                   case 1:
                        System.out.print("Enter quantity ");
```

```
                    int q1=s.nextInt();
                   sum = q1*price * motorTax;
            break;
        case 2:
                    System.out.print("Enter quantity ");
                    int q2=s.nextInt();
            sum = q2*price * fanTax;
            break;
        case 3:
                    System.out.print("Enter quantity ");
                    int q3=s.nextInt();
            sum = q3*price * tubeTax;
            break;
        case 4:
                    System.out.print("Enter quantity ");
                    int q4=s.nextInt();
            sum = q4*price * wiresTax;
            break;
        default:
                    System.out.print("Enter quantity ");
                    int q5=s.nextInt();
            sum = q5*price * otherItemTax;
            break;
                }
                float totalAmount = price + sum;

      System.out.println("Item price: $" + price);
      System.out.println("Tax amount: $" + sum);
      System.out.println("Total amount (including tax): " + totalAmount);
System.out.println(" 23DCS017 Khushi Dadhaniya");

}
}
```

OUTPUT:

```
D:\java>javac Bill.java

D:\java>java Bill
Enter 1 for motor, 2 for fan, 3 for tube, 4 for wires: 2
Enter price of the item: 500
Enter quantity 5
Item price: $500.0
Tax amount: $300.0
Total amount (including tax): 800.0
 23DCS017 Khushi Dadhaniya

D:\java>
```

CONCLUSION:

This program demonstrates how to use a switch statement in Java to calculate a bill based on product codes and prices, applying specific tax rates depending on the product type. It's a practical example of conditional logic and arithmetic operations in a real-world scenario (like a sales transaction in a shop). Depending on your requirements, you can expand this program to handle more products, include user input for prices, or integrate with a larger billing system.

6. Create a Java program that prompts the user to enter the number of days (n) for which they want to generate their exercise routine. The program should then calculate and display the first n terms of the Fibonacci series, representing the exercise duration for each day.

PROGRAM CODE:

```
import java.lang.*;
import java.util.Scanner;

class fibo
{

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

```
                int n1=0,n2=1,n3,i;
                System.out.println("Enter the number of days");
                int days=s.nextInt();



                System.out.println ("fibonacci series is");
                System.out.println(" "+ n1);
                System.out.println(" "+ n2);

                for(i=2;i<days;i++)
                {
                        n3=n1+n2;
                        System.out.println(" "+ n3);
                        n1=n2;
                        n2=n3;

                }
 System.out.println(" 23DCS017 Khushi Dadhaniya");

        }

}
```

OUTPUT:

```
D:\java>javac fibo.java

D:\java>java fibo
Enter the number of days
6
fibonacci series is
 0
 1
 1
 2
 3
 5
 23DCS017 Khushi Dadhaniya
```

CONCLUSION:

This Java program effectively calculates and displays the exercise duration for each day based on the Fibonacci series. It demonstrates basic user input handling (Scanner), iterative Fibonacci sequence generation, and output formatting. You can

modify and expand upon this program for further functionality, such as adding validation for user input or calculating additional metrics based on exercise durations.

Supplementary Experiment:

Imagine you are developing a classroom management system. You need to keep track of the grades of students in a class. After collecting the grades, you want to display each student's grade along with a message indicating if they have passed or failed. Let's assume the passing grade is 50.

PROGRAM CODE:

```
import java.util.Scanner;

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

        System.out.print("Enter the number of students: ");
        int numStudents = scanner.nextInt();

        // Assuming passing grade is 50
        int passingGrade = 50;

        for (int i = 1; i <= numStudents; i++) {
            System.out.print("Enter grade for student " + i + ": ");
            int studentGrade = scanner.nextInt();

            if (studentGrade >= passingGrade) {
                System.out.println("Student " + i + ": Passed");
            } else {
                System.out.println("Student " + i + ": Failed");
            }
        }

 System.out.println(" 23DCS017 Khushi Dadhaniya");

    }
}
```

OUTPUT:

```
D:\java>javac Grade.java

D:\java>java Grade
Enter the number of students: 5
Enter grade for student 1: 78
Student 1: Passed
Enter grade for student 2: 45
Student 2: Failed
Enter grade for student 3: 90
Student 3: Passed
Enter grade for student 4: 23
Student 4: Failed
Enter grade for student 5: 47
Student 5: Failed

D:\java>
```

CONCLUSION:

This Java program effectively manages and displays students' grades along with their pass/fail status based on a predefined passing grade threshold. It demonstrates basic array handling, iteration, conditional logic, and output formatting in a classroom management context. Depending on your specific requirements, you can extend this program to include additional functionalities such as calculating averages, sorting grades, or storing data in a database for persistence.