CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

DEVANG PATEL INSTITUTE OF ADVANCE TECHNOLOGY & RESEARCH

Department of Computer Science & Engineering

Subject Name: Java Programming
Semester: 3
Subject Code: CSE201
Academic year: 2024 - 25

PART – 1 (Data Types, Variables, String, Control Statements, Operators, Arrays)

No.	Aim of the Practical
1.	Demonstration of installation steps of Java,Introduction to Object Oriented Concepts, comparison of Java with other object-oriented programming languages. Introduction to JDK, JRE, JVM, Javadoc, command line argument.Introduction to Eclipse or NetBeans IDE,or BlueJ and Console Programming. **QUESTIN ANS:** 1. Installation Steps for Java: To install Java, follow these steps: Visit the official Oracle website. Download the latest version of the JDK that matches your operating system. Run the installer and follow the on-screen instructions. Set the JAVA_HOME environment variable to the JDK installation directory. 2. Object-Oriented Concepts in Java: Object-Oriented Programming (OOP) in Java revolves around objects, which combine data (fields) and behavior (methods). Key OOP concepts include: Class: A blueprint for creating objects. Object: An instance of a class.

Inheritance: Creating a new class based on an existing one.
Polymorphism: The ability of objects to take on different forms.
Encapsulation: Bundling data (fields) and methods together.

3. Comparison of Java with Other Languages:
   Let's compare Java with a few other languages:
   Python: High-level, interpreted, and concise. Python programs are shorter, but Java has better library support.
   C++: Java was derived from C++. C++ supports operator overloading, while Java is purely object-oriented.

4. Introduction to JDK, JRE, and JVM:
   These are essential components for Java development:
   JVM (Java Virtual Machine): Executes Java bytecode, making Java platform-independent.
   JRE (Java Runtime Environment): Provides Java class libraries and the JVM for running Java applications.
   JDK (Java Development Kit): Includes JRE and development tools (compilers, JavaDoc, etc.). Use it for Java application development.

5. IDEs for Java Development:
   There are several IDEs available, including:
   Eclipse: Developed by IBM, Eclipse offers plugins, code insight, and support for multiple languages.
   IntelliJ IDEA: Developed by JetBrains, it's Java-based and feature-rich.
   NetBeans: Supports Java, has a variety of plugins, and is faster to learn.

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 :

```
class bank
{
public static void main (String[ ] args)
{
int a = 20;
System.out.println("Current Balance : " + a);
System.out.println("23DCS030_Shreya
Garasia");
}
};
```

OUTPUT:

```
Current Balance : 20
23DCS030_Shreya Garasia
```

CONCLUSION:

In This Practical We leant About how to Print Value.

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.util.Scanner;

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

      System.out.print("Enter distance in meters:
");
```

```
        float distance = scanner.nextFloat();

        System.out.print("Enter hour: ");
        float hours = scanner.nextFloat();

        System.out.print("Enter minutes: ");
        float minutes = scanner.nextFloat();

        System.out.print("Enter seconds: ");
        float seconds = scanner.nextFloat();

        float timeSeconds = (hours * 3600) +
(minutes * 60) + seconds;
        float metersPerSecond = distance /
timeSeconds;
        float kilometersPerHour = (distance /
1000.0f) / (timeSeconds / 3600.0f);
        float milesPerHour = kilometersPerHour /
1.609f;

        System.out.println("Speed in
meters/second is " + metersPerSecond);
        System.out.println("Speed in km/h is " +
kilometersPerHour);
        System.out.println("Speed in miles/h is " +
milesPerHour);
System.out.println("23DCS030_Shreya
Garasia");
        scanner.close();
    }
}
```

OUTPUT:

```
Enter distance in meters: 150
Enter hour: 2
Enter minutes: 10
Enter seconds: 60
Speed in meters/second is 0.01908397
Speed in km/h is 0.06870229
Speed in miles/h is 0.042698752
23DCS030_Shreya Garasia
```

CONCLUSION:

In This Practical We leant About Convert Values In One From To Another.

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 t calculate the sum of elements in an array representing daily expenses.

PROGRAM CODE :

```
import java.util.Scanner;

public class budget
{
  public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter the number of days in the month: ");
      int numDays = scanner.nextInt();
      double[] expenses = new double[numDays];
      for (int i = 0; i < numDays; i++)
{
  System.out.print("Enter daily expense for day : "+ (i + 1) );
       expenses[i] = scanner.nextDouble();
      }
      double sumBudget = 0;
      for (int i = 0; i < numDays; i++)
{
         sumBudget = sumBudget + expenses[i];
      }
```

```
        System.out.println("Your Monthly Budget: " + sumBudget + " Rupees");
        System.out.println("23DCS030_Shreya Garasia");

         scanner.close();
    }
}
```

OUTPUT:

```
Enter the number of days in the month: 7
Enter daily expense for day : 1 75
Enter daily expense for day : 2 120
Enter daily expense for day : 3 250
Enter daily expense for day : 4 100
Enter daily expense for day : 5 80
Enter daily expense for day : 6 275
Enter daily expense for day : 7 50
Your Monthly Budget: 950.0 Rupees
23DCS030_Shreya Garasia
```

CONCLUSION:

In This Practical We did Sum Of Monthly Budget.

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.util.Scanner;

class pra5
{
    public static void main(String[] args)
 {
        Scanner sc = new Scanner(System.in);

        // Product codes and prices
        double[] prices = {0,250,550,150,100};
```

```
        System.out.print("Enter product code (1 for motor, 2 for fan, 3 for tube, 4 for
wires): ");
        int tax = sc.nextInt();

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



        double totalPrice = prices[tax] * quantity;

        switch (tax)
 {
            case 1: // Motor
                totalPrice =  totalPrice + totalPrice *8 /100;; // 8% tax
                break;
            case 2: // Fan
                totalPrice =  totalPrice + totalPrice *12 /100;; // 12% tax
                break;
            case 3: // Tube
                totalPrice =  totalPrice + totalPrice *5 /100; // 5% tax
                break;
            case 4: // Wires
                totalPrice =   totalPrice + totalPrice *75 /100 ; // 7.5% tax
                break;
            default:
                // For other items, no additional tax
                break;
        }

        System.out.println("Total bill amount: " + totalPrice);
        System.out.println("23DCS030_Shreya Garasia");


    }
}
```

OUTPUT:

```
Enter product code (1 for motor, 2 for fan, 3 for tube, 4 for wires): 2
Enter quantity: 4
Total bill amount: 2464.0
23DCS030_Shreya Garasia
```

CONCLUSION:

In This Practical We leant About Switch Case To Get Product Code

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.util.Scanner;
//PRACTICAL 6
public class routine {
      static int fibonacci(int n)
{
       if (n <= 1)
          return n;
       return fibonacci(n - 1) + fibonacci(n - 2);
  }
    public static void main(String[] args)
 {
       Scanner scanner = new Scanner(System.in);

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

       System.out.println("Exercise routine for " + n + " days:");
       for (int i = 0; i < n; i++)
 {
          int duration = fibonacci(i);
          System.out.println("Day " + (i + 1) + ": " + duration + " minutes");
```

```
  }
      System.out.println("23DCS030_Shreya Garasia");
      scanner.close();
   }
}
```

OUTPUT:

```
Enter the number of days : 14
Exercise routine for 14 days:
Day 1: 0 minutes
Day 2: 1 minutes
Day 3: 1 minutes
Day 4: 2 minutes
Day 5: 3 minutes
Day 6: 5 minutes
Day 7: 8 minutes
Day 8: 13 minutes
Day 9: 21 minutes
Day 10: 34 minutes
Day 11: 55 minutes
Day 12: 89 minutes
Day 13: 144 minutes
Day 14: 233 minutes
23DCS030_Shreya Garasia
```

CONCLUSION:

In This Practical We Did display the first n terms of the Fibonacci series, representing the exercise duration for each day.s