

Assignment

COVER PAGE

Institute Name : Besant Technologies
Project Name : Book Manager
Submitted to : Roshiga
Submitted by : Nagasri Bondili
Student ID : 25BTF41834
Batch No : 418 Java
Co-ordinator Name : Roshiga
Date of Submission : July 30, 2025

DAY - 8

Question 1: Create 3 classes: Shopping, Bank, and Student. Each class should cover all Java data types and some constants.

```
class Shopping {
    int itemCount = 3;
    double itemPrice = 499.50;
    boolean isDiscountAvailable = true;
    char currencySymbol = '₹';
    Integer discountPercent = 15;
    Float tax = 7.5f;
    String itemName = "Sneakers";
    final double GST_RATE = 0.18;
    final String SHOP_NAME = "StyleMart";
    void displayShoppingDetails() {
        System.out.println("---- Shopping Details ----");
        System.out.println("Shop Name: " + SHOP_NAME);
        System.out.println("Item: " + itemName);
        System.out.println("Price per Item: " + currencySymbol
+ itemPrice);
        System.out.println("Discount: " + discountPercent +
"%");
        System.out.println("Tax: " + tax + "%");
        System.out.println("GST Rate: " + GST_RATE);
        System.out.println("Is Discount Available: " +
isDiscountAvailable);
        System.out.println("Total Items Bought: " +
itemCount);
    }
}
class Bank {
    long accountNumber = 9876543210L;
    float balance = 15234.75f;
    byte branchCode = 45;
    short atmPin = 4321;

    Long mobileNumber = 9988776655L;
    Double interestRate = 4.25;
    String accountHolder = "Nagasri Bondili";
    final String BANK_NAME = "Unity Bank";
    final float MIN_BALANCE = 1000.0f;
    String[] bankList = {
        "State Bank of India",
        "HDFC Bank",
        "ICICI Bank",
        "Punjab National Bank",
        "Axis Bank"
    };
    void displayBankDetails() {
        System.out.println("\n---- Bank Details ----");
```

```

        System.out.println("Bank Name: " + BANK_NAME);
        System.out.println("Account Holder: " +
accountHolder);
        System.out.println("Account Number: " +
accountNumber);
        System.out.println("Branch Code: " + branchCode);
        System.out.println("Balance: " + currencySymbol() +
balance);
        System.out.println("Interest Rate: " + interestRate +
"%");
        System.out.println("Registered Mobile: " +
mobileNumber);
        System.out.println("Minimum Balance: " + MIN_BALANCE);
        System.out.println("List of Banks in India:");
        for (String bank : bankList) {
            System.out.println("- " + bank);
        }
    }
    char currencySymbol() {
        return '₹';
    }
}
class Student {
    int rollNo = 1123;
    float percentage = 91.6f;
    boolean isPassed = true;
    char grade = 'A';
    Boolean hasScholarship = true;
    Double cgpa = 9.45;
    String name = "Sri";
    final String COLLEGE_NAME = "Vignan's Nirula";
    final int MAX_MARKS = 100;
    void displayStudentDetails() {
        System.out.println("\n---- Student Details ----");
        System.out.println("College Name: " + COLLEGE_NAME);
        System.out.println("Student Name: " + name);
        System.out.println("Roll Number: " + rollNo);
        System.out.println("Percentage: " + percentage + "%");
        System.out.println("CGPA: " + cgpa);
        System.out.println("Grade: " + grade);
        System.out.println("Passed: " + isPassed);
        System.out.println("Has Scholarship: " +
hasScholarship);
        System.out.println("Max Marks per Subject: " +
MAX_MARKS);
    }
}
public class AllClassesDemo {
    public static void main(String[] args) {
        Shopping shopping = new Shopping();
        Bank bank = new Bank();

```

```

        Student student = new Student();
        shopping.displayShoppingDetails();
        bank.displayBankDetails();
        student.displayStudentDetails();
    }
}

```

Output

```

---- Shopping Details ----
Shop Name: StyleMart
Item: Sneakers
Price per Item: ₹499.5
Discount: 15%
Tax: 7.5%
GST Rate: 0.18
Is Discount Available: true
Total Items Bought: 3

---- Bank Details ----
Bank Name: Unity Bank
Account Holder: Nagasri Bondili
Account Number: 9876543210
Branch Code: 45
Balance: ₹15234.75
Interest Rate: 4.25%
Registered Mobile: 9988776655
Minimum Balance: 1000.0
List of Banks in India:
- State Bank of India
- HDFC Bank
- ICICI Bank
- Punjab National Bank
- Axis Bank
---- Student Details ----
College Name: Vignan's Nirula
Student Name: Sri
Roll Number: 1123
Percentage: 91.6%
CGPA: 9.45
Grade: A
Passed: true
Has Scholarship: true
Max Marks per Subject: 100

```

Question 2 : create a String array for the Bank class and add a list of any 5 banks in India.

```

class Bank {
    String[] bankList = {

```

```

        "State Bank of India",
        "HDFC Bank",
        "ICICI Bank",
        "Punjab National Bank",
        "Axis Bank"
    };
    void displayBankList() {
        System.out.println("List of Popular Banks in India:");
        for (String bank : bankList) {
            System.out.println("- " + bank);
        }
    }
    public static void main(String[] args) {
        Bank bank = new Bank();
        bank.displayBankList();
    }
}

```

OUTPUT

List of Popular Banks in India:
- State Bank of India
- HDFC Bank
- ICICI Bank
- Punjab National Bank
- Axis Bank

DAY - 9

Write a Java program to take 3 numbers and find which number is the largest using if-else statements. Also, check if all three numbers are equal.

```

import java.util.Scanner;
public class LargestOfThree {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        int a = sc.nextInt();
        System.out.print("Enter second number: ");
        int b = sc.nextInt();
        System.out.print("Enter third number: ");
        int c = sc.nextInt();
        if (a == b && b == c) {
            System.out.println("All three numbers are equal.");
        } else if (a >= b && a >= c) {
            System.out.println("The largest number is: " + a);
        } else if (b >= a && b >= c) {
            System.out.println("The largest number is: " + b);
        } else {
            System.out.println("The largest number is: " + c);
        }
    }
    sc.close();
}

```

```
    }
}
```

Sample Output (Example 1):

```
Enter first number: 45
Enter second number: 89
Enter third number: 12
The largest number is: 89
```

Sample Output (Example 2):

```
Enter first number: 100
Enter second number: 100
Enter third number: 100
All three numbers are equal.
```

DAY – 10

Question 1: Write a Java program to print the first 5 odd numbers from the first 100 natural numbers using a while loop.

```
public class FirstFiveOdd {
    public static void main(String[] args) {

        int numbers = 100;
        int i = 1;
        int count = 1;

        while (i <= numbers && count <= 5) {
            if (i % 2 != 0) {
                System.out.println("Odd Number: " + i);
                count++;
            }
            i++;
        }
    }
}
```

Output:

```
Odd Number: 1
Odd Number: 3
Odd Number: 5
Odd Number: 7
Odd Number: 9
```

Question 2: Declare an ArrayList with student roll numbers and display each roll number using a for loop and a foreach loop.

```

import java.util.ArrayList;

public class RollNumberList {
    public static void main(String[] args) {
        ArrayList<Integer> rollNumbers = new ArrayList<>();

        // Add roll numbers
        rollNumbers.add(101);
        rollNumbers.add(102);
        rollNumbers.add(103);
        rollNumbers.add(104);
        rollNumbers.add(105);

        System.out.println("Using for loop:");
        for (int i = 0; i < rollNumbers.size(); i++) {
            System.out.println("Roll No: " +
rollNumbers.get(i));
        }

        System.out.println("\nUsing foreach loop:");
        for (int roll : rollNumbers) {
            System.out.println("Roll No: " + roll);
        }
    }
}

```

Output:

Using for loop:
 Roll No: 101
 Roll No: 102
 Roll No: 103
 Roll No: 104
 Roll No: 105

Using foreach loop:
 Roll No: 101
 Roll No: 102
 Roll No: 103
 Roll No: 104
 Roll No: 105

DAY – 11

Question 1: Create an abstract class Shopping and write

- **An abstract method called buyCloth()**
- **An implemented method called payAmount()**

```

abstract class Shopping {
    abstract void buyCloth();

    void payAmount(double amount) {
        System.out.println("Amount Paid: ₹" + amount);
    }
}

```

Question 2: Create a class ClothShopping that extends the abstract class Shopping, and display:

- The cloth bought
- The amount paid

```

class ClothShopping extends Shopping {
    String cloth = "T-Shirt";
    double amount = 599.0;

    void buyCloth() {
        System.out.println("Cloth Bought: " + cloth);
    }

    void displayShopping() {
        buyCloth();
        payAmount(amount);
    }
}

```

Question 3: Create an interface Institute with the following two unimplemented methods:

- getStudentName()
- getNoOfStudent()

```

interface Institute {
    String getStudentName();
    int getNoOfStudent();
}

```

Question 4: Create a class Besant that implements the Institute interface. In the main method, display:

- The student name
- The number of students

```

class Besant implements Institute {
    public String getStudentName() {
        return "Nagasri Bondili";
    }

    public int getNoOfStudent() {
        return 120;
    }
}

```

```

public static void main(String[] args) {
    Besant obj = new Besant();

    System.out.println("Student Name: " +
obj.getStudentName());
    System.out.println("No. of Students: " +
obj.getNoOfStudent());

    // From earlier class
    ClothShopping shopping = new ClothShopping();
    shopping.displayShopping();
}
}

```

Output:

```

Student Name: Nagasri Bondili
No. of Students: 120
Cloth Bought: T-Shirt
Amount Paid: ₹599.0

```

DAY – 13 (Inheritance Example)

Question 1: Create a parent class called Bank that holds common bank information. Demonstrate single inheritance by extending this class to SBI. From SBI, create another class SBP to demonstrate multilevel inheritance. Also, create another child class ICICIBank that extends Bank separately to show hierarchical inheritance.

```

// Parent class
class Bank {
    String bankName = "Generic Bank";
    String location = "India";

    void displayBankInfo() {
        System.out.println("Bank Name: " + bankName);
        System.out.println("Location: " + location);
    }
}

// Single Inheritance - SBI extends Bank
class SBI extends Bank {
    String branch = "SBI - Guntur";

    void displaySBI() {
        System.out.println("Branch: " + branch);
    }
}

// Multilevel Inheritance - SBP extends SBI
class SBP extends SBI {
    String accountType = "Savings";
}

```

```

        void displaySBP() {
            System.out.println("Account Type: " + accountType);
        }
    }

// Hierarchical Inheritance - ICICIBank also extends Bank
class ICICIBank extends Bank {
    String manager = "Mr. Ramesh";

    void displayICICI() {
        System.out.println("ICICI Manager: " + manager);
    }
}

// Main class
public class InheritanceDemo {
    public static void main(String[] args) {

        // Single + Multilevel
        SBP sbpObj = new SBP();
        System.out.println("--- SBP (Multilevel Inheritance) ---");
        sbpObj.displayBankInfo();
        sbpObj.displaySBI();
        sbpObj.displaySBP();

        // Hierarchical
        ICICIBank iciciObj = new ICICIBank();
        System.out.println("\n--- ICICI Bank (Hierarchical Inheritance) ---");
        iciciObj.displayBankInfo();
        iciciObj.displayICICI();
    }
}

```

Output:

```

--- SBP (Multilevel Inheritance) ---
Bank Name: Generic Bank
Location: India
Branch: SBI - Guntur
Account Type: Savings

--- ICICI Bank (Hierarchical Inheritance) ---
Bank Name: Generic Bank
Location: India
ICICI Manager: Mr. Ramesh

```

DAY – 15

Question: Create a list of student names using ArrayList in Java. Perform the following operations:

- Add 5 student names to the list
- Display all the student names
- Remove a student
- Update a student's name
- Check if a student exists
- Display the total number of students

```

import java.util.ArrayList;

public class StudentList {
    public static void main(String[] args) {
        // Create an ArrayList of student names
        ArrayList<String> students = new ArrayList<>();

        // Add student names
        students.add("Nagasri");
        students.add("Bhavya");
        students.add("Hema");
        students.add("Sanjana");
        students.add("Pranavi");

        // Display all names
        System.out.println("Student List: " + students);

        // Remove a student
        students.remove("Hema");
        System.out.println("After removing Hema: " +
        students);

        // Update a student's name
        students.set(2, "Anusha");
        System.out.println("After updating index 2 to Anusha:
" + students);

        // Check if a student exists
        boolean exists = students.contains("Nagasri");
        System.out.println("Is Nagasri in the list? " +
exists);

        // Total number of students
        int total = students.size();
        System.out.println("Total number of students: " +
total);
    }
}

```

Output:

```

Student List: [Nagasri, Bhavya, Hema, Sanjana, Pranavi]
After removing Hema: [Nagasri, Bhavya, Sanjana, Pranavi]
After updating index 2 to Anusha: [Nagasri, Bhavya, Anusha,
Pranavi]

```

```
Is Nagasri in the list? true
Total number of students: 4
```

DAY – 19

Question 1: Write a Java program to read the contents of a file using File and FileReader classes. The program should display each character on the console until the end of the file is reached.

```
package org.besant.fileoperations;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

public class FileOperations {
    public static void main(String[] args) {
        File file = new File("C:\\\\Users\\\\ADMIN\\\\eclipse-
workspace\\\\Batch418_JavaTraining\\\\BesantFileOperationsProject\\\\
src\\\\org\\\\besant\\\\fileoperations\\\\Test.txt");
        FileReader fr;

        try {
            fr = new FileReader(file);
            int i = -1;
            while (true) {
                try {
                    i = fr.read();
                    System.out.print((char) i);
                } catch (IOException e) {
                    e.printStackTrace();
                }
                if (i == -1) {
                    break;
                }
            }
            try {
                fr.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } finally {
            file = null;
        }
    }
}
```

Output (Assuming file contains):
Hello, this is a sample file.

Question 2: Write a Java program using FileWriter to write a custom message into a file. The program should create or overwrite a file and write the content into it.

```
package org.besant.fileoperations;

import java.io.FileWriter;
import java.io.IOException;

public class FileWriterConcept {
    public static void main(String[] args) {
        try {
            FileWriter fw = new
FileWriter("C:\\\\Users\\\\ADMIN\\\\eclipse-
workspace\\\\Batch418_JavaTraining\\\\BesantFileOperationsProject\\\\src\\\\org\\\\besant\\\\fileoperations\\\\TestOutput.txt");
            fw.write("Hello This is my first file write
program");
            fw.flush();
            fw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Output (Written inside file TestOutput.txt):
Hello This is my first file write program

DAY – 20

Question 1: Write a Java program using BufferedWriter to write data into a text file.

```
package org.besant.fileoperations;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class FileWriterConcept {

    public static void main(String[] args) {
        try {
            FileWriter fw = new FileWriter("TestOutput.txt");
            BufferedWriter bw = new BufferedWriter(fw);
            bw.write("Hello This is Buffered Writer program");
            bw.flush();
            bw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

```
}
```

Output:

The file "TestOutput.txt" is created and the content written is:
Hello This is Buffered Writer program

Question 2: Write a Java program using BufferedReader to read data from a file and print it to the console line by line.

```
package org.besant.fileoperations;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileNotFoundException;
import java.io.IOException;

public class FileOperations {

    public static void main(String[] args) throws IOException
    {
        File file = new File("Test.txt");
        FileReader fr;
        try {
            fr = new FileReader(file);
            BufferedReader br = new BufferedReader(fr);
            String data;
            while ((data = br.readLine()) != null) {
                System.out.println(data);
            }
            fr.close();
            br.close();
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } finally {
            file = null;
        }
    }
}
```

Output:

(Depends on the content inside Test.txt. Example below)
Welcome to Java File Handling
This file is read using BufferedReader

Question 3: Write a Java program to use DataOutputStream to write data types into a binary file, and read them back using DataInputStream.

```
package org.besant.fileoperations;

import java.io.DataInputStream;
import java.io.DataOutputStream;
```

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class DataInputOutputStream {

    public static void main(String[] args) throws IOException
    {
        FileOutputStream fos = new
FileOutputStream("data.bin");
        DataOutputStream dos = new DataOutputStream(fos);

        dos.writeInt(25);
        dos.writeBoolean(true);

        dos.flush();
        dos.close();
        fos.close();

        FileInputStream fis = new FileInputStream("data.bin");
        DataInputStream dis = new DataInputStream(fis);

        int age = dis.readInt();
        boolean status = dis.readBoolean();

        System.out.println("Age is:::" + age + "::Status:::" +
status);
    }
}
```

Output:

Age is:::25::Status:::true

Day - 26

Question : Create a index.html and capture empname and empage, create a servlet called EmployeeServlet and received those 2 parameters in the servlet and use prepared statement to insert the data in the employee table

1. index.html

```
html
<!DOCTYPE html>
<html>
<head>
    <title>Employee Form</title>
</head>
<body>
    <h2>Enter Employee Details</h2>
    <form action="EmployeeServlet" method="post">
        <label for="empname">Employee Name:</label>
        <input type="text" id="empname" name="empname" required><br><br>

        <label for="empage">Employee Age:</label>
        <input type="number" id="empage" name="empage" required><br><br>

        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

2. EmployeeServlet.java

```
java
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/EmployeeServlet")
public class EmployeeServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // Database credentials and URL
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/your_database_name";
    private static final String DB_USER = "your_username";
```

```
private static final String DB_PASS = "your_password";

protected void doPost(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {

String empname = request.getParameter("empname");
String empageStr = request.getParameter("empage");

int empage = Integer.parseInt(empageStr);

response.setContentType("text/html");
PrintWriter out = response.getWriter();

try {
    // Load the JDBC driver
    Class.forName("com.mysql.cj.jdbc.Driver");

    // Connect to the database
    Connection conn =
DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);

    // SQL query using PreparedStatement
    String sql = "INSERT INTO employee (empname,
empage) VALUES (?, ?)";
    PreparedStatement pstmt =
conn.prepareStatement(sql);
    pstmt.setString(1, empname);
    pstmt.setInt(2, empage);

    int rows = pstmt.executeUpdate();

    if (rows > 0) {
        out.println("<h3>Employee inserted
successfully!</h3>");
    } else {
        out.println("<h3>Error inserting
employee.</h3>");
    }

    pstmt.close();
    conn.close();

} catch (Exception e) {
    out.println("<h3>Error: " + e.getMessage() +
"</h3>");

}
}
```

3. Sample MySQL Table

Make sure the employee table is created:

```
sql
CREATE TABLE employee (
    id INT AUTO_INCREMENT PRIMARY KEY,
    empname VARCHAR(100),
    empage INT
);
```

Day - 27

Question: Create a index.html and capture empname and empage, create a servlet called EmployeeServlet and received those 2 parameters in the servlet and use prepared statement to update the data in the employee table

1. index.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Update Employee</title>
</head>
<body>
    <h2>Update Employee Age</h2>
    <form action="EmployeeServlet" method="post">
        <label for="empname">Employee Name:</label>
        <input type="text" id="empname" name="empname" required><br><br>

        <label for="empage">New Age:</label>
        <input type="number" id="empage" name="empage" required><br><br>

        <input type="submit" value="Update">
    </form>
</body>
</html>
```

2. EmployeeServlet.java

```
java
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
```

```
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/EmployeeServlet")
public class EmployeeServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // Change these as per your DB setup
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/your_database_name";
    private static final String DB_USER = "your_username";
    private static final String DB_PASS = "your_password";

    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
        throws ServletException, IOException {

        String empname = request.getParameter("empname");
        String empageStr = request.getParameter("empage");

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        try {
            int empage = Integer.parseInt(empageStr);

            // Load JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Connect to the database
            Connection conn =
DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);

            // SQL query to update employee age
            String sql = "UPDATE employee SET empage = ? WHERE
empname = ?";
            PreparedStatement pstmt =
conn.prepareStatement(sql);
            pstmt.setInt(1, empage);
            pstmt.setString(2, empname);

            int rowsUpdated = pstmt.executeUpdate();

            if (rowsUpdated > 0) {
                out.println("<h3>Employee age updated
successfully.</h3>");
            } else {

```

```
        out.println("<h3>No employee found with the  
given name.</h3>");  
    }  
  
    pstmt.close();  
    conn.close();  
} catch (Exception e) {  
    out.println("<h3>Error: " + e.getMessage() +  
"</h3>");  
}  
}  
}
```

3. Make Sure the Table Exists

If not already created:

```
sql  
CREATE TABLE employee (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    empname VARCHAR(100),  
    empage INT  
) ;
```