

## Experiment 8

### Employee Search Using an Array of Objects

#### Problem Statement

Write a Java program to store employee details including employee number, name, and salary, and search for an employee by employee number

```
class Employee {
    int empNo;
    String name;
    double salary;

    Employee(int empNo, String name, double salary) {
        this.empNo = empNo;
        this.name = name;
        this.salary = salary;
    }

    void display() {
        System.out.println("Employee Number: " + empNo);
        System.out.println("Employee Name: " + name);
        System.out.println("Employee Salary: " + salary);
    }
}

public class EmployeeSearch {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of employees: ");
        int n = sc.nextInt();
        Employee[] employees = new Employee[n];

        for (int i = 0; i < n; i++) {
            System.out.println("\nEnter details for Employee " + (i + 1) + ":");
            System.out.print("Employee Number: ");
            int empNo = sc.nextInt();
            sc.nextLine();
            System.out.print("Name: ");
            String name = sc.nextLine();
            System.out.print("Salary: ");
            double salary = sc.nextDouble();
            employees[i] = new Employee(empNo, name, salary);
        }

        System.out.print("\nEnter Employee Number to search: ");
        int searchEmpNo = sc.nextInt();
        boolean found = false;

        for (Employee emp : employees) {
            if (emp.empNo == searchEmpNo) {
                System.out.println("\nEmployee Found:");
                emp.display();
                found = true;
                break;
            }
        }

        if (!found) {
            System.out.println("Employee not found.");
        }

        sc.close();
    }
}
```

```
24mca13@mcaserver:~/java$ java EmployeeSearch
Enter the number of employees: 2

Enter details for Employee 1:
Employee Number: 1
Name: Athul
Salary: 100000000000

Enter details for Employee 2:
Employee Number: 2
Name: Anandhu Vasudev
Salary: 400000000000

Enter Employee Number to search: 2

Employee Found:
Employee Number: 2
Employee Name: Anandhu Vasudev
Employee Salary: 4.0E11
```

## Experiment 9

### String Search in an Array

#### Problem Statement

Write a Java program to store 'n' strings in an array. Search for a given string. If found, print its index; otherwise, display "String not found."

```
import java.util.Scanner;

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

        System.out.print("Enter the number of strings: ");
        int n = sc.nextInt();
        sc.nextLine();

        String[] strings = new String[n];

        System.out.println("Enter " + n + " strings:");
        for (int i = 0; i < n; i++) {
            strings[i] = sc.nextLine();
        }

        System.out.print("Enter the string to search: ");
        String searchString = sc.nextLine();

        boolean found = false;
        for (int i = 0; i < n; i++) {
            if (strings[i].equals(searchString)) {
                System.out.println("String found at index " + i);
                found = true;
                break;
            }
        }

        if (!found) {
            System.out.println("String not found.");
        }

        sc.close();
    }
}
```

```
24mca13@mcaserver:~/java$ java StringSearch
Enter the number of strings: 2
Enter 2 strings:
apple
orange
Enter the string to search: apple
String found at index 0
```

## Experiment 10

### String Manipulations

#### Problem Statement

Write a Java program to perform various string manipulations, including finding the length, converting to uppercase and lowercase, extracting characters and substrings, and reversing the string.

```
import java.util.Scanner;

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

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

        int length = str.length();
        System.out.println("Length of the string: " + length);

        System.out.println("Uppercase: " + str.toUpperCase());

        System.out.println("Lowercase: " + str.toLowerCase());

        if (length >= 3) {
            System.out.println("Substring (first 3 chars): " + str.substring(0, 3));
        } else {
            System.out.println("Substring: " + str);
        }

        String reversed = new StringBuilder(str).reverse().toString();
        System.out.println("Reversed string: " + reversed);

        sc.close();
    }
}
```

```
24mca13@mcaserver:~/java$ java StringManipulations
Enter a string: Athul
Length of the string: 5
Uppercase: ATHUL
Lowercase: athul
Substring (first 3 chars): Ath
Reversed string: luhtA
```

## Experiment 11

### Inheritance in Java

#### Problem Statement

Write a Java program to implement hierarchical inheritance for a book management system. Define a base class 'Publisher', a derived class 'Book', and two subclasses 'Literature' and 'Fiction'. Include methods to read and display book details and demonstrate the functionality using user input.

```
Fiction(String publisherName, String bookTitle, String authorName, String category) {
    super(publisherName, bookTitle, authorName);
    this.category = category;
}

void display() {
    System.out.println("\n[Fiction Book Details]");
    displayBook();
    System.out.println("Category: " + category);
}
}

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

        System.out.println("Enter details for Literature book:");
        System.out.print("Publisher Name: ");
        String pub1 = sc.nextLine();
        System.out.print("Book Title: ");
        String title1 = sc.nextLine();
        System.out.print("Author Name: ");
        String author1 = sc.nextLine();
        System.out.print("Genre: ");
        String genre = sc.nextLine();

        System.out.println("\nEnter details for Fiction book:");
        System.out.print("Publisher Name: ");
        String pub2 = sc.nextLine();
        System.out.print("Book Title: ");
        String title2 = sc.nextLine();
        System.out.print("Author Name: ");
        String author2 = sc.nextLine();
        System.out.print("Category: ");
        String category = sc.nextLine();

        Literature litBook = new Literature(pub1, title1, author1, genre);
        Fiction ficBook = new Fiction(pub2, title2, author2, category);

        litBook.display();
        ficBook.display();

        sc.close();
    }
}
```



```

import java.util.Scanner;

class Publisher {
    String publisherName;

    Publisher(String publisherName) {
        this.publisherName = publisherName;
    }

    void displayPublisher() {
        System.out.println("Publisher: " + publisherName);
    }
}

class Book extends Publisher {
    String bookTitle;
    String authorName;

    Book(String publisherName, String bookTitle, String authorName) {
        super(publisherName);
        this.bookTitle = bookTitle;
        this.authorName = authorName;
    }

    void displayBook() {
        displayPublisher();
        System.out.println("Book Title: " + bookTitle);
        System.out.println("Author: " + authorName);
    }
}

class Literature extends Book {
    String genre;

    Literature(String publisherName, String bookTitle, String authorName, String genre) {
        super(publisherName, bookTitle, authorName);
        this.genre = genre;
    }

    void display() {
        System.out.println("\n[Literature Book Details]");
        displayBook();
        System.out.println("Genre: " + genre);
    }
}

class Fiction extends Book {
    String category;

    Fiction(String publisherName, String bookTitle, String authorName, String category) {

```

## OUTPUT

```

24mca13@mcaserver:~/java$ java BookManagement
Enter details for Literature book:
Publisher Name: Apple
Book Title: Iphone
Author Name: Steav Jobs
Genre: Tech

Enter details for Fiction book:
Publisher Name: Warner Brothers
Book Title: Harry Potter
Author Name: J.K Jowling
Category: Magical

[Literature Book Details]
Publisher: Apple
Book Title: Iphone
Author: Steav Jobs
Genre: Tech

[Fiction Book Details]
Publisher: Warner Brothers
Book Title: Harry Potter
Author: J.K Jowling
Category: Magical

```

