

Experiment 8

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

Source code

```
import java.util.ArrayList;
import java.util.Scanner;
public class EmployeeSearch {
    private int empNumber;
    private String name;
    private double salary;
    public EmployeeSearch(int empNumber, String name, double salary) {
        this.empNumber = empNumber;
        this.name = name;
        this.salary = salary;
    }
    @Override
    public String toString() {
        return "Employee Number: " + empNumber + "\n Name: " + name + "\n
Salary: " + salary;
    }
    public static void main(String[] args) {
        ArrayList<EmployeeSearch> employees = new ArrayList<>();
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of employees: ");
        int numExperiment 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
Employees =
scanner.nextInt();

for (int i = 0; i < numEmployees; i++) {
    System.out.print("Employee Number: ");
```

```
int empNumber = scanner.nextInt(); scanner.nextLine(); //
Consume newline
System.out.print("Name: ");
String name = scanner.nextLine();
System.out.print("Salary: ");
double salary = scanner.nextDouble();

employees.add(new EmployeeSearch(empNumber, name, salary));
}

System.out.print("Enter Employee Number to search: ");
int searchEmpNumber = scanner.nextInt();

boolean found = false;
for (EmployeeSearch emp : employees) {
    if (emp.empNumber == searchEmpNumber) {
        System.out.println("Employee Found: " + emp);
        found = true;
        break;
    }
}

if (!found) {
    System.out.println("Employee with number " + searchEmpNumber
+ "
not found.");
}

scanner.close();
}
}
```

Output

```
Enter the number of employees: 4
Employee Number: 100
Name: Jithu
Salary: 25000
Employee Number: 101
Name: Ajin
Salary: 12062001
Employee Number: 102
Name: Anshul
Salary: 4092001
Employee Number: 103
Name: Jerin
Salary: 4112003
Enter Employee Number to search: 101
Employee Found: Employee Number: 101
Name: Ajin
Salary: 1.2062001E7
```

Experiment 9

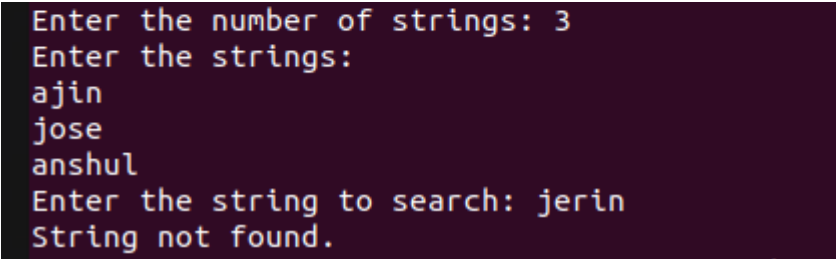
Aim :Write a Java program to store 'n' strings in an array. Search for a given string. If found, print its index; otherwise, display "String not found"

Source code

```
import java.util.Scanner;
public class StringSearch {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of strings: ");
        String[] strings = new String[scanner.nextInt()];
        scanner.nextLine(); // Consume newline
        System.out.println("Enter the strings:");
        for (int i = 0; i < strings.length; i++) {
            strings[i] = scanner.nextLine();
        }
        System.out.print("Enter the string to search: ");
        String searchString = scanner.nextLine();
        for (int i = 0; i < strings.length; i++) {
            if (strings[i].equals(searchString)) {
                System.out.println("String found at index: " + i);
            }
        }
    }
}
```

```
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of strings: ");
String[] strings = new String[scanner.nextInt()];
scanner.nextLine(); // Consume newline
System.out.println("Enter the strings:");
for (int i = 0; i < strings.length; i++) {
    strings[i] = scanner.nextLine();
}
System.out.print("Enter the string to search: ");
String searchString = scanner.nextLine();
for (int i = 0; i < strings.length; i++) {
    if (strings[i].equals(searchString)) {
        System.out.println("String found at index: " + i);
    }
}
```

output

A screenshot of a terminal window with a dark background. The text is displayed in a light color, showing the execution of the Java program. The output shows the user entering 3 for the number of strings, then entering 'ajin', 'jose', and 'anshul' as the strings. Then, the user enters 'jerin' as the string to search, and the program outputs 'String not found.'

```
Enter the number of strings: 3
Enter the strings:
ajin
jose
anshul
Enter the string to search: jerin
String not found.
```

Experiment 10

Aim: 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.

Source code

```
import java.util.Scanner; public class StringFun{ public static
void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a string: ");
String input = scanner.nextLine();
System.out.println("Length: " + input.length());
System.out.println("Uppercase: " + input.toUpperCase());
System.out.println("Lowercase: " + input.toLowerCase());
System.out.print("Enter index to extract character: ");
System.out.println("Character: " + input.charAt(scanner.nextInt()));
scanner.nextLine();
System.out.print("Enter start and end index for substring: ");
int start = scanner.nextInt(), end = scanner.nextInt();

System.out.println("Substring: " + input.substring(start, end));
System.out.println("Reversed: " + new
StringBuilder(input).reverse());
    scanner.close();
}

}
```

Output

```
Enter a string: Ajin Jose
Length: 9
Uppercase: AJIN JOSE
Lowercase: ajin jose
Enter index to extract character: 5
Character: J
Enter start and end index for substring: 5 9
Substring: Jose
Reversed: esoJ niJA
```

Experiment 11

Aim: 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.

Source code

```
import java.util.Scanner;
class Publisher {
    String name;
    Publisher(String name) { this.name = name; }
    void display() { System.out.println("Publisher: " + name); }
}
class Book extends Publisher {
    String title, author;
    Book(String name, String title, String author) {
        super(name);
        this.title = title;
        this.author = author; }
    void display() {
        super.display();
        System.out.println("Title: " + title + "\nAuthor: " + author);
    }
}
class Literature extends Book {
    Literature(String name, String title, String author) { super(name, title,
author); }
}
class Fiction extends Book {
    Fiction(String name, String title, String author) { super(name, title,
author); }
}
public class BookManagement {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter Publisher: ");
        String publisher = sc.nextLine();
```

```
System.out.print("Enter Title: ");  
String title = sc.nextLine();
```

```
System.out.print("Enter Author: ");  
String author = sc.nextLine();
```

```
System.out.print("Enter Category (Literature/Fiction): ");  
String category = sc.nextLine();
```

```
Book book = category.equalsIgnoreCase("Literature") ? new  
Literature(publisher, title, author) : new Fiction(publisher, title,  
author);
```

```
System.out.println("\nBook Details:");  
book.display();  
sc.close();  
}  
}
```

Output

```
Enter Publisher: Anshul  
Enter Title: My Sacrifice in RIT  
Enter Author: Ajin Jose  
Enter Category (Literature/Fiction): Fiction  
  
Book Details:  
Publisher: Anshul  
Title: My Sacrifice in RIT  
Author: Ajin Jose
```