

---

## Experiment 5: Write a java program to demonstrate the string manipulation such as:

- a) `toUpperCase()/toLowerCase()`
- b) `length()`
- c) `reverse()`
- d) `replace()`
- e) `charAt()`

### Aim

To write a Java program to demonstrate various **string manipulation operations** such as converting case, finding length, reversing a string, replacing characters, and accessing characters.

### Theory

Java provides a built-in String class that supports various methods for manipulating strings. Using these methods, we can modify, analyze, and access string data easily.

### Algorithm

1. Accept a string from the user.
2. Convert the string to uppercase and lowercase.
3. Find the length of the string.
4. Reverse the string using a loop.
5. Replace characters in the string.
6. Display a character at a given index.
7. Display all results.

### Program Code

```
import java.util.Scanner;  
class StringManipulationDemo {  
    public static void main(String[] args) {  
  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter a string: ");
```

```

String str = sc.nextLine();
// a. toUpperCase() and toLowerCase()
System.out.println("Uppercase: " + str.toUpperCase());
System.out.println("Lowercase: " + str.toLowerCase());

// b. length()
System.out.println("Length of string: " + str.length());

// c. reverse()
String rev = "";
for (int i = str.length() - 1; i >= 0; i--) {
    rev = rev + str.charAt(i);
}
System.out.println("Reversed string: " + rev);

// d. replace()
System.out.println("After replacing 'a' with '@': " + str.replace('a', '@'));

// e. charAt()
System.out.println("Character at index 2: " + str.charAt(2));

sc.close();
}
}
  
```

## Result

The program successfully demonstrates string manipulation operations such as case conversion, length calculation, reversing a string, replacing characters, and accessing individual characters.

## Notes

- `toUpperCase()` → Converts string to uppercase
- `toLowerCase()` → Converts string to lowercase
- `length()` → Returns total number of characters
- `replace()` → Replaces characters in a string
- `charAt()` → Returns character at specified index