|  | **Pimpri Chinchwad Education Trust’s**  **Pimpri Chinchwad College of Engineering** |
| --- | --- |

**Practical Assignment**

| **Department: MCA Semester:** I | **Academic Year:** 2024-2025 **Year and Div.: FYMCA** | | **Maximum Marks:** |
| --- | --- | --- | --- |
| **Subject: Java Programming Lab** | | **Subject Teacher: Prof. Rajkamal Sangole**  **Name & sign** | |
| **Assignment No: 3** | **Date:** | | **Date of Submission:** |

1. Write a Java program to find the length of a given string using the length() method.
2. Write a Java program to count the frequency of each character in a string.
3. Write a Java program to check if a given string is a palindrome using both String and StringBuffer.
4. Write a Java program to concatenate two strings using both the + operator and the concat() method.
5. Write a Java program to reverse a string using a StringBuffer and the reverse() method.
6. Write a Java program to extract a substring from a given string using the substring() method.
7. Write a Java program to replace all occurrences of a specific character in a string with another character using the replace() method.
8. Write a Java program to count the number of vowels and consonants in a string.
9. Write a Java program to split a string into an array of substrings using a delimiter with the split() method.
10. Write a Java program to check if two given strings are anagrams of each other.

**Practice Programs:**

1. Write a Java program to compare the performance of StringBuffer and StringBuilder for appending strings in a loop.
2. Write a Java program to remove all leading and trailing whitespace from a string using the trim() method.
3. Write a Java program to count the number of words in a given string.
4. Write a Java program to convert all characters in a string to uppercase and lowercase using the toUpperCase() and toLowerCase() methods.
5. Write a Java program to find the character at a specific index in a string using the charAt() method.
6. Write a Java program to demonstrate the use of the append() and delete() methods of the StringBuffer class.
7. Write a Java program to check if a given substring exists within a string using the contains() method.
8. Write a Java program to compare two strings lexicographically (dictionary order) using the compareTo() method.
9. Write a Java program to generate all permutations of a given string using recursion.
10. Write a Java program to reverse each word in a given string while keeping the word order intact using StringBuffer.