**Strings - Handling strings and common string functions**

**Functions - parameter passing, return**

**1. Count Vowels and Consonants**

**Problem:**

**Write a Java program to count the number of vowels and consonants in a given string.**

| **import java.util.Scanner; class VowelsAndConsonants{  public static void count(String str){  int vows=0, cons=0;  for(int i=0; i<str.length();i++){  char ch = str.charAt(i);  String s = Character.toString(ch);  if(s.equalsIgnoreCase("a") || s.equalsIgnoreCase("e") ||  s.equalsIgnoreCase("i") ||s.equalsIgnoreCase("o")||  s.equalsIgnoreCase("u")){  vows++;  }  else{  cons++;  }  }  System.out.println("Vowels - " + vows);  System.out.println("Consonants - " + cons);  }  public static void main(String[] args){  String str;  Scanner input = new Scanner(System.in);  str = input.nextLine();  count(str);  input.close();  } }** |
| --- |

**2. Reverse a String**

**Problem:**

**Write a Java program to reverse a given string without using any built-in reverse functions.**

**import java.util.Scanner;**

**class StringReverse{**

**public static void isReverse(String text){**

**String rev = "";**

**for(int i=0; i<text.length(); i++){**

**char ch = text.charAt(i);**

**rev = ch+rev;**

**}**

**System.out.println("Reverse of the given String : "+ rev);**

**}**

**public static void main(String[] args){**

**Scanner input = new Scanner(System.in);**

**System.out.print("Enter a String : ");**

**String text = input.next();**

**isReverse(text);**

**input.close();**

**}**

**}**

**3. Palindrome String Check**

**Problem:**

**Write a Java program to check if a given string is a palindrome (a string that reads the same forward and backward).**

**import java.util.Scanner;**

**class Palindrome{**

**public static void isPalindrome(String text){**

**String rev = "";**

**for(int i=0; i<text.length(); i++){**

**char ch = text.charAt(i);**

**rev = ch+rev;**

**}**

**if(text.equals(rev)) System.out.println("Its a Palindrome ");**

**else System.out.println("Its not a Palindrome ");**

**}**

**public static void main(String[] args){**

**Scanner input = new Scanner(System.in);**

**System.out.print("Enter a String : ");**

**String text = input.next();**

**isPalindrome(text);**

**input.close();**

**}**

**}**

**4. Remove Duplicates from a String**

**Problem:**

**Write a Java program to remove all duplicate characters from a given string and return the modified string.**

**import java.util.Scanner;**

**class DuplicateCharacters{**

**public static void removeDuplicates(String str){**

**String str1 = "";**

**for(int i=0; i<str.length(); i++){**

**char ch = str.charAt(i);**

**if(str1.indexOf(ch)==-1){**

**str1=str1+ch;**

**}**

**}**

**System.out.println(str1);**

**}**

**public static void main(String[] args){**

**Scanner input = new Scanner(System.in);**

**String str = input.nextLine();**

**removeDuplicates(str);**

**input.close();**

**}**

**}**

**5. Find the Longest Word in a Sentence**

**Problem:**

**Write a Java program that takes a sentence as input and returns the longest word in the sentence.**

**import java.util.Scanner;**

**class LongestWord{**

**public static void FindLongest(String str){**

**String[] words = str.split(" ");**

**String longestWord = "";**

**for(int i=0; i<words.length; i++){**

**if(words[i].length()>longestWord.length()){**

**longestWord = words[i];**

**}**

**}**

**System.out.println("The longest word is : " + longestWord);**

**}**

**public static void main(String[] args) {**

**Scanner input = new Scanner(System.in);**

**String sentence = input.nextLine();**

**FindLongest(sentence);**

**input.close();**

**}**

**}**

**6. Find Substring Occurrences**

**Problem:**

**Write a Java program to count how many times a given substring occurs in a string.**

| **import java.util.Scanner; class SubstringOccurance{  public static void countSubstring(String text, String substr){  int count = 0;  int len = substr.length();  for(int i=0; i<=text.length()-len; i++){  if(text.substring(i,i+len).equals(substr)){  count++;  }  }  System.out.println("Occurances : " + count);   }  public static void main(String[] args){  Scanner input = new Scanner(System.in);  System.out.print("Enter a String : ");  String text = input.nextLine();  System.out.print("Enter a Sub String : ");  String substr = input.nextLine();  countSubstring(text, substr);   input.close();  } }** |
| --- |

**7. Toggle Case of Characters**

**Problem:**

**Write a Java program to toggle the case of each character in a given string. Convert uppercase letters to lowercase and vice versa.**

**import java.util.Scanner;**

**class ToggleCase{**

**public static void CaseConvert(String str){**

**char[] s = str.toCharArray();**

**for(int i=0; i<s.length; i++){**

**if(Character.isUpperCase(s[i])){**

**s[i] = Character.toLowerCase(s[i]);**

**}**

**else if(Character.isLowerCase(s[i])){**

**s[i] = Character.toUpperCase(s[i]);**

**}**

**}**

**s.toString();**

**System.out.println(s);**

**}**

**public static void main(String[] args){**

**Scanner input = new Scanner(System.in);**

**System.out.print("Enter a String: ");**

**String str = input.nextLine();**

**CaseConvert(str);**

**input.close();**

**}**

**}**

**8. Compare Two Strings**

**Problem:**

**Write a Java program to compare two strings lexicographically (dictionary order) without using built-in compare methods.**

**Example Input:**

**String 1: "apple"**

**String 2: "banana"**

**Expected Output:**

**"apple" comes before "banana" in lexicographical order**

| **import java.util.Scanner; class StringCompare{  public static void compare(String str1, String str2){  int len = Math.min(str1.length(),str2.length());  for(int i=0;i<len;i++){  int a = (int)str1.charAt(i);  int b = (int)str2.charAt(i);  if(a > b){  System.out.println(str1 + " comes after " + str2 + " in lexicographical order ");  break;  }  else if(a<b){  System.out.println(str2 + " comes after " + str1 + " in lexicographical order ");  break;  }    }  }  public static void main(String[] args) {  Scanner input = new Scanner(System.in);  System.out.print("Enter first String : ");  String str1 = input.next();  System.out.print("Enter second String : ");  String str2 = input.next();  compare(str1, str2);  input.close();  } }** |
| --- |

**9. Find the Most Frequent Character**

**Problem:**

**Write a Java program to find the most frequent character in a string. Example Input:**

**String: "success"**

**Expected Output:**

**Most Frequent Character: 's'**

**import java.util.Scanner;**

**class MostFrequentCharacter{**

**public static void findCharacter(String str){**

**char[] arr = str.toCharArray();**

**StringBuilder s = new StringBuilder();**

**for(int i=0; i<arr.length; i++){**

**boolean unique = true;**

**for(int j=0; j<s.length(); j++){**

**if(arr[i] == s.charAt(j)){**

**unique = false;**

**}**

**}**

**if(unique) s.append(arr[i]);**

**}**

**s.toString();**

**char[] sarr = s.toString().toCharArray();**

**int[] freq = new int[sarr.length];**

**for(int i=0;i<sarr.length;i++){**

**for(int j=0; j<arr.length; j++){**

**if(sarr[i] == arr[j]){**

**freq[i]++;**

**}**

**}**

**}**

**int max = -1;**

**char ch = ' ' ;**

**for(int i=0; i<freq.length; i++){**

**if(freq[i]>max) {**

**max = freq[i];**

**ch = sarr[i];**

**}**

**}**

**System.out.println("Most repeated character : " + ch);**

**}**

**public static void main(String[] args){**

**Scanner input = new Scanner(System.in);**

**System.out.print("Enter a String : ");**

**String str = input.next();**

**findCharacter(str);**

**input.close();**

**}**

**}**

**10. Remove a Specific Character from a String**

**Problem:**

**Write a Java program to remove all occurrences of a specific character from a string. Example Input:**

**String: "Hello World"**

**Character to Remove: 'l'**

**Expected Output:**

**Modified String: "Heo Word"**

| **import java.util.Scanner; class RemoveOccurance{  public static void removechar(String str,char ch){  String s = "";  for(int i=0; i<str.length(); i++){  if(str.charAt(i) != ch){  s = s + str.charAt(i);  }  }  System.out.println("Modified String : " + s);  }  public static void main(String[] args){  Scanner input = new Scanner(System.in);  System.out.print("Enter a string : ");  String str = input.next();  System.out.print("Enter the character to remove : ");  char ch = input.next().charAt(0);  removechar(str,ch);  input.close();  } }** |
| --- |

**11. Write a Java program that accepts two strings from the user and checks if the two strings are anagrams of each other (i.e., whether they contain the same characters in any order).**

**import java.util.Scanner;**

**class Anangram{**

**public static void checkAnagram(String str1, String str2){**

**str1 = str1.toLowerCase();**

**str2 = str2.toLowerCase();**

**if(str1.length() != str2.length()){**

**System.out.println("Not anagrams");**

**}**

**else{**

**int[] arr = new int[256];**

**for(int i=0; i<str1.length(); i++){**

**arr[str1.charAt(i)]++;**

**arr[str2.charAt(i)]--;**

**}**

**boolean anagram = true;**

**for(int i=0; i<arr.length; i++){**

**if(arr[i]>0) anagram = false;**

**}**

**if (anagram) System.out.println("Both strings are anagrams");**

**else System.out.println("Both strings are not anagrams");**

**}**

**}**

**public static void main(String[] args) {**

**Scanner input = new Scanner(System.in);**

**System.out.print("Enter first String : ");**

**String str1 = input.next();**

**System.out.print("Enter second String : ");**

**String str2 = input.next();**

**checkAnagram(str1, str2);**

**input.close();**

**}**

**}**

**12. Write a replace method in Java that replaces a given word with another word in a sentence**

| **import java.util.Scanner; class StringReplace{  public static void replaceWord(String sentence, String word, String newword){  String newstr = sentence.replace(word,newword);  System.out.println("Modified Sentence : \n" + newstr);  }  public static void main(String[] args) {  Scanner input = new Scanner(System.in);  System.out.println("Enter the sentence : ");  String sentence = input.nextLine();  System.out.println("Enter the word to be replaced : ");  String word = input.next();  System.out.println("Enter the new word : ");  String newword = input.next();  replaceWord(sentence, word, newword);   input.close();  }}** |
| --- |