**DAY -3 ASSIGNMENT.**

**1.)** **Write a program that takes a String through Command Line argument and display the length of the string. Also display the string into uppercase and check whether it is a palindrome or not.**

**package** ABC;

**import** java.util.\*;

**public** **class** Palindrome {

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

    String input = "madam";

**int** length = input.length();

        String upperCaseString = input.toUpperCase();

**boolean** isPalindrome = *checkPalindrome*(input);

        System.***out***.println("Length of the string: " + length);

        System.***out***.println("Uppercase string: " + upperCaseString);

        System.***out***.println("Is palindrome: " + isPalindrome);

    }

**private** **static** **boolean** checkPalindrome(String str) {

**int** left = 0;

**int** right = str.length() - 1;

**while** (left < right) {

**if** (str.charAt(left) != str.charAt(right)) {

**return** **false**;

            }

            left++;

            right--;

        }

**return** **true**;

    }

}

**2.) Write a program that accepts two numbers in the range from 1 to 40 from the Command Line. Then compares these numbers against a single dimension array of five integer elements ranging in value from 1 to 40.**

**package** ABC;

**import** java.util.\*;

**public** **class** Bingo {

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

**try** {

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

**int** num1 =sc.nextInt();

**int** num2 =sc.nextInt();

**if** (num1 < 1 || num1 > 40 || num2 < 1 || num2 > 40) {

                System.***out***.println("Please provide numbers in the range from 1 to 40.");

**return**;

            }

**int**[] array = {7, 25, 5, 19, 30};

            System.***out***.println("Your first number was " + num1);

            System.***out***.println("Your second number was " + num2);

**boolean** foundNum1 = **false**;

**boolean** foundNum2 = **false**;

**for** (**int** value : array) {

**if** (value == num1) {

                    foundNum1 = **true**;

                }

**if** (value == num2) {

                    foundNum2 = **true**;

                }

            }

**if** (foundNum1 && foundNum2) {

                System.***out***.println("Its Bingo!");

            } **else** {

                System.***out***.println("Not Found!");

            }

            System.***out***.print("The array was ");

**for** (**int** value : array) {

                System.***out***.print(value + " ");

            }

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

        } **catch** (NumberFormatException e) {

            System.***out***.println("Please provide valid numbers.");

        }

    }

}

**3.)** **Write a program that allows you to create an integer array of 18 elements with the following values.**

**package** ABC;

**import** java.util.\*;

**public** **class** Computations{

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

**int**[] A = {3, 2, 4, 5, 6, 4, 5, 7, 3, 2, 3, 4, 7, 1, 2, 0, 0, 0};

**int** sum = 0;

**for** (**int** i = 0; i <= 14; i++) {

            sum += A[i];

        }

        A[15] = sum;

**double** totalSum = 0;

**for** (**int** i = 0; i < 18; i++) {

            totalSum += A[i];

        }

**double** average = totalSum / 18;

        A[16] = (**int**) average;

**int** smallest = A[0];

**for** (**int** i = 1; i < 18; i++) {

**if** (A[i] < smallest) {

                smallest = A[i];

            }

        }

        A[17] = smallest;

        System.***out***.println("Array elements after computations:");

**for** (**int** i = 0; i < 18; i++) {

            System.***out***.print(A[i] + " ");

        }

        System.***out***.println();

        System.***out***.println("Sum of elements from index 0 to 14: " + A[15]);

        System.***out***.println("Average of all elements: " + A[16]);

        System.***out***.println("Smallest value in the array: " + A[17]);

    }

}

**4.) Jaffer was done with the online shopping, and the next step was to complete the bill payment, for which he must provide his name, credit card number, expiry month/year and CVV code.**

**package** ABC;

**import** java.util.Scanner;

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**public** **class** Regex {

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

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

        System.***out***.print("Enter your name: ");

        String name = scanner.nextLine();

        System.***out***.print("Enter your credit card number (format: XXXX-XXXX-XXXX-XXXX): ");

        String creditCardNumber = scanner.nextLine();

        System.***out***.print("Enter expiry month (MM): ");

        String expiryMonth = scanner.nextLine();

        System.***out***.print("Enter expiry year (YY): ");

        String expiryYear = scanner.nextLine();

        System.***out***.print("Enter CVV code: ");

        String cvvCode = scanner.nextLine();

**boolean** isValid = **true**;

**if** (name.isEmpty()) {

            System.***out***.println("Name cannot be empty.");

            isValid = **false**;

        }

**if** (creditCardNumber.isEmpty()) {

            System.***out***.println("Credit card number cannot be empty.");

            isValid = **false**;

        }

**if** (expiryMonth.isEmpty()) {

            System.***out***.println("Expiry month cannot be empty.");

            isValid = **false**;

        }

**if** (expiryYear.isEmpty()) {

            System.***out***.println("Expiry year cannot be empty.");

            isValid = **false**;

        }

**if** (cvvCode.isEmpty()) {

            System.***out***.println("CVV code cannot be empty.");

            isValid = **false**;

        }

        Pattern ccPattern = Pattern.*compile*("^\\d{4}\\d{4}\\d{4}\\d{4}$");

        Matcher ccMatcher = ccPattern.matcher(creditCardNumber);

**if** (!ccMatcher.matches()) {

            System.***out***.println("Credit card number must be 16 characters long and include a dash (-) after each four-digit number.");

            isValid = **false**;

        }

        Pattern monthPattern = Pattern.*compile*("^(0[1-9]|1[0-2])$");

        Matcher monthMatcher = monthPattern.matcher(expiryMonth);

**if** (!monthMatcher.matches()) {

            System.***out***.println("Expiry month must be in the format MM (01-12).");

            isValid = **false**;

        }

   Pattern yearPattern = Pattern.*compile*("^\\d{2}$");

        Matcher yearMatcher = yearPattern.matcher(expiryYear);

**if** (!yearMatcher.matches()) {

            System.***out***.println("Expiry year must be in the format YY.");

            isValid = **false**;

        }

        Pattern cvvPattern = Pattern.*compile*("^\\d{3}$");

        Matcher cvvMatcher = cvvPattern.matcher(cvvCode);

**if** (!cvvMatcher.matches()) {

            System.***out***.println("CVV code must be of length 3.");

            isValid = **false**;

        }

**if** (isValid) {

            System.***out***.println("Name: " + name);

            System.***out***.println("Credit Card Number: " + creditCardNumber);

            System.***out***.println("Expiry Month/Year: " + expiryMonth + "/" + expiryYear);

            System.***out***.println("CVV Code: " + cvvCode);

        }

    }

}

**5.) Create an interface Medicine Info to represent a drug manufactured by a pharmaceutical company.**

**TestMedicine.java**

**package** interfaces;

**import** java.util.\*;

**public** **class** TestMedicine {

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

        MedicineInfo[] medicines = **new** MedicineInfo[10];

        Random random = **new** Random();

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

**int** choice = random.nextInt(3) + 1;

**switch** (choice) {

**case** 1:

                    medicines[i] = **new** Tablet();

**break**;

**case** 2:

                    medicines[i] = **new** Syrup();

**break**;

**case** 3:

                    medicines[i] = **new** Ointment();

**break**;

            }

        }

**for** (MedicineInfo medicine : medicines) {

            medicine.displayLabel();

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

        }

    }

}

**Tablet.java**

**package** interfaces;

**class** Tablet **implements** MedicineInfo {

    @Override

**public** **void** displayLabel() {

        System.***out***.println("Tablet: Store in a cool dry place.");

        System.***out***.println("Manufactured by: XYZ Pharmacy, 1234 Pharma St, Health City");

    }

}

**class** Syrup **implements** MedicineInfo {

    @Override

**public** **void** displayLabel() {

        System.***out***.println("Syrup: Shake well before use.");

        System.***out***.println("Manufactured by: ABC Pharmacy, 5678 Medicine Rd, Cure Town");

    }

}

**class** Ointment **implements** MedicineInfo {

    @Override

**public** **void** displayLabel() {

        System.***out***.println("Ointment: For external use only.");

        System.***out***.println("Manufactured by: DEF Pharmacy, 91011 Remedy Blvd, Healville");

    }

}

**MedicineInfo.java**

**package** interfaces;

**public** **interface** MedicineInfo {

**void** displayLabel();

}

**6.) Sort the given array in descending order, i.e., arrange the elements from largest to smallest.**

**package** ABC;

**import** java.util.\*;

**public** **class** Sort{

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

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

        System.***out***.println("Enter the array elements :");

        String input = scanner.nextLine();

        String[] inputStrings = input.split(" ");

        Integer[] array = **new** Integer[inputStrings.length];

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

            array[i] = Integer.*parseInt*(inputStrings[i]);

        }

        Arrays.*sort*(array, Collections.*reverseOrder*());

        System.***out***.println("Sorted array in descending order: " + Arrays.*toString*(array));

    }

}

**7.) Given a sorted array arr [] of size N, the task is to remove the duplicate elements from the array.**

**package** ABC;

**import** java.util.\*;

**public** **class** Remove{

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

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

        System.***out***.println("Enter the array elements :");

        String input = scanner.nextLine();

        String[] inputStrings = input.split(" ");

**int**[] arr = **new** **int**[inputStrings.length];

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

            arr[i] = Integer.*parseInt*(inputStrings[i]);

        }

**int** newLength = *removeDuplicates*(arr);

        System.***out***.print("Array after removing duplicates: ");

**for** (**int** i = 0; i < newLength; i++) {

            System.***out***.print(arr[i] + " ");

        }

        System.***out***.println();

    }

**public** **static** **int** removeDuplicates(**int**[] arr) {

**if** (arr.length == 0) {

**return** 0;

        }

**int** uniqueIndex = 0;

**for** (**int** i = 1; i < arr.length; i++) {

**if** (arr[i] != arr[uniqueIndex]) {

                uniqueIndex++;

                arr[uniqueIndex] = arr[i];

            }

        }

**return** uniqueIndex + 1;

    }

}

**8.)** **Given two arrays and our task is to find their common elements**.

**package** ABC;

**import** java.util.\*;

**public** **class** Array {

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

        String[] array1 = {"Article", "for", "Apple", "for", "Grapes"};

        String[] array2 = {"Article", "Apple", "Grape"};

        List<String> commonElements = *findCommonElements*(array1, array2);

        System.***out***.println("Common elements: " + commonElements);

        String[] array3 = {"a", "b", "c", "d", "e", "f"};

        String[] array4 = {"b", "d", "e", "h", "g", "c"};

        commonElements = *findCommonElements*(array3, array4);

        System.***out***.println("Common elements: " + commonElements);

    }

**public** **static** List<String> findCommonElements(String[] array1, String[] array2) {

        Set<String> set1 = **new** HashSet<>();

        Set<String> set2 = **new** HashSet<>();

**for** (String s : array1) {

            set1.add(s);

        }

**for** (String s : array2) {

            set2.add(s);

        }

set1.retainAll(set2);

**return** **new** ArrayList<>(set1);

    }

}

**9.)** **Given a 2D array of order N X M and a column number K ( 1<=K<=m). Our task is to sort the 2D array according to values in Column K.**

**package** ABC;

**import** java.util.Arrays;

**public** **class** TwoDArray{

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

**int**[][] array = {

            {39, 27, 11, 42},

            {10, 93, 91, 90},

            {54, 78, 56, 89},

            {24, 64, 20, 65}

        };

**int** columnToSoryBy =1;

        System.***out***.println("Original Array:");

*printArray*(array);

*sort2DArrayByColumn*(array, columnToSortBy);

        System.***out***.println("Sorted Array:" + (columnToSortBy + 1) + ":");

*printArray*(array);

    }

**public** **static** **void** sort2DArrayByColumn(**int**[][] array, **int** column) {

  Arrays.*sort*(array, (a, b) -> Integer.*compare*(a[column], b[column]));

    }

**public** **static** **void** printArray(**int**[][] array) {

**for** (**int**[] row : array) {

**for** (**int** element : row) {

                System.***out***.print(element + " ");

            }

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

        }

    }

}

**10.) Write the Java Program to Find Transpose of Matrix given matrix.**

**package** ABC;

**import** java.util.Scanner;

**public** **class**  Matrix{

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

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

        System.***out***.println("Enter the number of rows and columns:");

**int** rows = scanner.nextInt();

**int** columns = scanner.nextInt();

**int**[][] matrix = **new** **int**[rows][columns];

        System.***out***.println("Enter the matrix elements:");

**for** (**int** i = 0; i < rows; i++) {

**for** (**int** j = 0; j < columns; j++) {

                matrix[i][j] = scanner.nextInt();

            }

        }

**int**[][] transpose = **new** **int**[columns][rows];

**for** (**int** i = 0; i < rows; i++) {

**for** (**int** j = 0; j < columns; j++) {

                transpose[j][i] = matrix[i][j];

            }

        }

        System.***out***.println("Transpose of the matrix:");

**for** (**int** i = 0; i < columns; i++) {

**for** (**int** j = 0; j < rows; j++) {

                System.***out***.print(transpose[i][j] + " ");

            }

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

        }

    }

}

**11.)** **Given a matrix of size Row x Col Print the boundary elements of the matrix. Boundary elements are those elements which are not surrounded by elements on all four directions, i.e. elements in the first row, first column, last row and last column.**

**package** ABC;

**import** java.util.\*;

**public** **class** Matrix2{

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

**int**[][] matrix = {

    {1,2,3},

    {4,5,6},

    {7,8,9}

    };

*printBoundaryElements*(matrix);

    }

**public** **static** **void** printBoundaryElements(**int**[][] matrix) {

**int** rows = matrix.length;

**int** cols = matrix[0].length;

**for** (**int** i = 0; i < rows; i++) {

**for** (**int** j = 0; j < cols; j++) {

**if**(i == 0 || i == rows -1 || j == 0 || j == cols - 1) {

           System.***out***.print(matrix[i][j] + " ");

           } **else** {

           System.***out***.print(" ");

           }

           }

      System.***out***.println( );

    }

    }

}

**12.)** **For a given 2D square matrix of size N\*N, the task is to find the sum of elements in the Principal and Secondary diagonals.**

**package** ABC;

**import** java.util.Scanner;

**public** **class** Diagonals{

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

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

        System.***out***.print("Enter size of square matrix (N): ");

**int** N = scanner.nextInt();

**int**[][] matrix = **new** **int**[N][N];

        System.***out***.println("Enter the matrix elements:");

**for** (**int** i = 0; i < N; i++) {

**for** (**int** j = 0; j < N; j++) {

                matrix[i][j] = scanner.nextInt();

            }

        }

**int** principalDiagonalSum = 0;

**int** secondaryDiagonalSum = 0;

**for** (**int** i = 0; i < N; i++) {

            principalDiagonalSum += matrix[i][i];

            secondaryDiagonalSum += matrix[i][N - i - 1];

        }

        System.***out***.println("Principal Diagonal: " + principalDiagonalSum);

        System.***out***.println("Secondary Diagonal: " + secondaryDiagonalSum);

    }

}

**13.)** **Write a program that takes String as input and count the last ‘n’ vowels of a given String. If the number is greater than the vowels found, then print ‘Mismatch in Vowel Count .**

**package** ABC;

**import** java.util.Scanner;

**public** **class** Vowels {

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

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

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

        String inputString = scanner.nextLine();

        System.***out***.print("Enter the number of vowels : ");

**int** n = scanner.nextInt();

**int** vowelCount = *countVowels*(inputString);

        String lastNVowels = *getLastNVowels*(inputString, n);

**if** (n > vowelCount) {

            System.***out***.println("Mismatch in Vowel Count");

        } **else** {

            System.***out***.println("Last " + n + " vowels : " + lastNVowels);

        }

    }

**public** **static** **int** countVowels(String str) {

**int** count = 0;

        str = str.toLowerCase();

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

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

**if** (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {

                count++;

            }

        }

**return** count;

    }

**public** **static** String getLastNVowels(String str, **int** n) {

        StringBuilder sb = **new** StringBuilder();

        str = str.toLowerCase();

**for** (**int** i = str.length() - 1; i >= 0 && n > 0; i--) {

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

**if** (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {

                sb.insert(0, ch);

                n--;

            }

        }

**return** sb.toString();

    }

}

**14.)** **Write a Java Program to reverse a string.**

**package** ABC;

**import** java.util.\*;

**public** **class** Reverse {

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

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

        System.***out***.print("Enter a string to reverse: ");

        String input = scanner.nextLine();

        String reversed = *reverseString*(input);

        System.***out***.println("Reversed string: " + reversed);

    }

**public** **static** String reverseString(String str) {

        StringBuilder sb = **new** StringBuilder(str);

        sb.reverse();

**return** sb.toString();

    }

}

**15.)** **Write the Java Program to Sort a String.**

**package** ABC;

**import** java.util.\*;

**public** **class** SortString {

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

        String input = "JavaProgram";

        System.***out***.println("String is : " + input );

        String sortedString = *sortString*(input);

        System.***out***.println("Sorted string: " + sortedString);

    }

**public** **static** String sortString(String str) {

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

        Arrays.*sort*(charArray);

        String sortedString = **new** String(charArray);

**return** sortedString;

    }

}

**16.)** **Given string str, the task is to write a Java program to swap the pairs of characters of a string. If the string contains an odd number of characters, then the last character remains as it is.**

**package** ABC;

**import** java.util.\*;

**public** **class** Swap{

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

        String str1 = "Java";

        String str2 = "Testing";

        System.***out***.println("Input: " + str1);

        System.***out***.println("Output: " + *swapPairs*(str1));

        System.***out***.println("\nInput: " + str2);

        System.***out***.println("Output: " + *swapPairs*(str2));

    }

**public** **static** String swapPairs(String str) {

        StringBuilder result = **new** StringBuilder();

**for** (**int** i = 0; i < str.length() - 1; i += 2) {

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

**char** second = str.charAt(i + 1);

            result.append(second).append(first);

        }

**if** (str.length() % 2 != 0) {

            result.append(str.charAt(str.length() - 1));

        }

**return** result.toString();

    }

}

**17.)** **Given string str, the task is to write Java Program check whether the given string is a pangram or not. A string is a pangram string if it contains all the character of the alphabets ignoring the case of the alphabets.**

**package** ABC;

**import** java.util.\*;

**public** **class** Pangram{

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

        String str1 = "Abcdefghijklmnopqrstuvwxyz";

        String str2 = "welcome";

        System.***out***.println("Input: " + str1);

        System.***out***.println("Output: " + (*isPangram*(str1) ? "yes" : "no"));

        System.***out***.println("\nInput: " + str2);

        System.***out***.println("Output: " + (*isPangram*(str2) ? "yes" : "no"));

    }

**public** **static** **boolean** isPangram(String str) {

**boolean**[] alphabetPresent = **new** **boolean**[26];

        String lowerCaseStr = str.toLowerCase();

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

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

**if** ('a' <= ch && ch <= 'z') {

                alphabetPresent[ch - 'a'] = **true**;

            }

        }

**for** (**boolean** present : alphabetPresent) {

**if** (!present) {

**return** **false**;

            }

        }

**return** **true**;

    }

}

**18.) Write the java program to extract the first letter of each word in it.**

**package** ABC;

**import** java.util.\*;

**public** **class** Extract{

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

        String input1 = "Practice Java Program";

        String input2 = "United Kingdom";

        System.***out***.println("Input: " + input1);

        System.***out***.println("Output: " + *extractFirstLetters*(input1));

        System.***out***.println("Input: " + input2);

        System.***out***.println("Output: " + *extractFirstLetters*(input2));

    }

**public** **static** String extractFirstLetters(String str) {

        StringBuilder result = **new** StringBuilder();

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

**for** (String word : words) {

**if** (!word.isEmpty()) {

                result.append(word.charAt(0));

            }

        }

**return** result.toString();

    }

}

**19.)** **Given a String, the task is to insert another string in between the given String at a particular specified index in Java.**

**package** ABC;

**import** java.util.\*;

**public** **class** Index {

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

        String originalString = "Computer Portal";

        String stringToBeInserted = "Science";

**int** index = 8;

        String modifiedString = *insertString*(originalString, stringToBeInserted, index);

        System.***out***.println("Original String: " + originalString);

        System.***out***.println("String to be inserted: " + stringToBeInserted);

        System.***out***.println("Index: " + index);

        System.***out***.println("Modified String: " + modifiedString);

    }

**public** **static** String insertString(String originalString, String stringToBeInserted, **int** index) {

        StringBuilder sb = **new** StringBuilder(originalString);

        sb.insert(index, stringToBeInserted + " ");

**return** sb.toString();

    }

}

**20.)** **Write a Java program to print Even length words in a String.**

**package** ABC;

**import** java.util.\*;

**public** **class** EvenLength {

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

        String s1 = "This is java language";

        String s2 = "i am Tester";

        System.***out***.println("Input: s = \" " + s1 + "\"");

*printEvenLengthWords*(s1);

        System.***out***.println();

        System.***out***.println("Input: s = \" " + s1 + "\"");

*printEvenLengthWords*(s2);

    }

**public** **static** **void** printEvenLengthWords (String s) {

        String[] words = s.split(" ");

**for** (String word : words) {

**if** (word.length() % 2 == 0) {

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

            }

        }

    }

}