**1.**

**public** **class** CharacterMostAppearInString {

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

String input = "Chennai";

**char** result = *findMaxChar*(input);

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

}

**public** **static** **char** findMaxChar(String input) {

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

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

charCount[input.charAt(i)]++;

}

**int** maxCount = 0;

**char** maxChar = 0;

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

**if** (charCount[i] > maxCount) {

maxCount = charCount[i];

maxChar = (**char**) i;

}

}

**return** maxChar;

}

}

2.

package com.task;

import java.util.HashMap;

import java.util.Map;

public class CountDuplicateCharacters {

public static void main(String[] args) {

String str = "Hello, world!";

Map<Character, Integer> charCountMap = new HashMap<>();

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

char c = str.charAt(i);

if (charCountMap.containsKey(c)) {

int count = charCountMap.get(c);

charCountMap.put(c, count + 1);

} else {

charCountMap.put(c, 1);

}

}

System.out.println("Duplicate characters:");

for (Map.Entry<Character, Integer> entry : charCountMap.entrySet()) {

if (entry.getValue() > 1) {

System.out.println("'" + entry.getKey() + "' appears " + entry.getValue() + " times");

}

}

}

}

3.

**package** com.task;

**public** **class** JoiningmultipleStringsWithDelimiter {

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

String[] strings = {"apple", "banana", "cherry"};

String delimiter = ", ";

String result = *joinStrings*(strings, delimiter);

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

}

**public** **static** String joinStrings(String[] strings, String delimiter) {

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

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

sb.append(strings[i]);

**if** (i != strings.length - 1) {

sb.append(delimiter);

}

}

**return** sb.toString();

}

}

4.

**package** com.task;

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

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

String s1 = "malayalam";

String s2 = "";

**for** (**int** i = s1.length()-1; i >= 0; i--) {

s2 = s2 + s1.charAt(i);

}

System.***out***.println(s2);

**if** (s1.equals(s2))

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

**else** System.***out***.println("Not palindrome");

}

}

5.

**package** com.task;

**public** **class** RemoveAllWhiteSpace {

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

**return** str.replaceAll("\\s", "");

}

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

String str = "Tentacle Tech";

System.***out***.println(*removeSpaces*(str)); // "Hello,World!"

}

}

6.

**package** com.task;

**public** **class** RemoveDuplicateCharacter {

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

String s = "javaprogram";

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

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

**char** charAt = s.charAt(i);

**int** indexOf = s.indexOf(charAt,i+1);

**if** (indexOf== -1) {

sb.append(charAt);

}

}

System.***out***.println("Removing duplicate character: "+sb);

}

}

7.

**package** com.task;

**import** java.util.Iterator;

**public** **class** ReverseString {

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

String s = "karthi";

**for** (**int** i = s.length()-1; i >= 0; i--) {

System.***out***.print(s.charAt(i));

}

}

}

8.

package com.task;

import java.util.Arrays;

import java.util.Comparator;

public class Sorting {

public static void main(String[] args) {

String[] arr = {"apple", "banana", "kiwi", "orange", "pear"};

Arrays.sort(arr, Comparator.comparingInt(String::length));

System.out.println(Arrays.toString(arr));

}

}

9.

**package** com.task;

**public** **class** StringCharacterRemover {

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

String input = "Tentacle";

**char** c = 'e';

String result = *removeCharacter*(input, c);

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

}

**public** **static** String removeCharacter(String input, **char** c) {

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

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

**char** current = input.charAt(i);

**if** (current != c) {

sb.append(current);

}

}

**return** sb.toString();

}

}

10.

**package** com.task;

**public** **class** StringContainsOnlyDigits {

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

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

**if** (!Character.*isDigit*(str.charAt(i))) {

**return** **false**;

}

}

**return** **true**;

}

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

String str1 = "12345";

String str2 = "12A45";

**if** (*containsOnlyDigits*(str1)) {

System.***out***.println(str1 + " contains only digits.");

} **else** {

System.***out***.println(str1 + " does not contain only digits.");

}

**if** (*containsOnlyDigits*(str2)) {

System.***out***.println(str2 + " contains only digits.");

} **else** {

System.***out***.println(str2 + " does not contain only digits.");

}

}

}

11.

**package** com.task;

**public** **class** StringToNumberConverter {

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

String str = "1234567";

**int** intValue = Integer.*parseInt*(str);

System.***out***.println("intValue: " + intValue);

**long** longValue = Long.*parseLong*(str);

System.***out***.println("longValue: " + longValue);

**float** floatValue = Float.*parseFloat*(str);

System.***out***.println("floatValue: " + floatValue);

**double** doubleValue = Double.*parseDouble*(str);

System.***out***.println("doubleValue: " + doubleValue);

}

}

12.

**package** com.task2;

**public** **class** Execrise1 {

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

**int** m =6;

**int** n=7;

**int** multiply = (m\*n/2)%7;

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

}

}

13.

**package** com.task2;

**public** **class** Execrise6 {

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

{

**int** num = 11111;

**int** value = 0;

**int** cal = 0;

**int** i = 1;

**int** rem;

**while** (num > 0) {

rem = num % 10;

cal = cal + rem \* i;

value = value + cal;

num = num / 10;

rem = 0;

i++;

}

System.***out***.println(value % 10);

}

}

14.

**package** com.task2;

**public** **class** Execrise8a {

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

{

**int** less = 100;

**for** (**int** a = 1; a < less; a++) {

**for** (**int** b = a + 1; b < less; b++) {

**for** (**int** c = b + 1; c < less; c++) {

**if** (a \* a + b \* b == c \* c) {

System.***out***.println(a + "^2+" + b +"^2=" + c +"^2");

}

}

}

}

}

}

}

15.

**package** com.task2;

**public** **class** Execrise8b {

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

**for** (**int** a= 1; a < 100; a++) {

**for** (**int** b = a+1; b < 100; b++) {

**for** (**int** c = b+1; c < 100; c++) {

**for** (**int** d = 0; d < 100; d++) {

//a2+b2 =c2+d2

**if** (a\* a+b\* b == c\* c+d\* d) {

System.***out***.println(a+"^2+"+ b + "^2= "+ c + "^2 +"+ d + "^2");

}

}

}

}

}

}

}