CSA09 Programming in Java

# Day 2

# ASSIGNMENT – 2

Questions for Developing a code in Java

1. Write a java program

i. to compare two strings lexicographically, ignoring case differences.

PROGRAM:

import java.util.Scanner;

public class StringComparator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the first string: ");

String str1 = scanner.nextLine();

System.out.print("Enter the second string: ");

String str2 = scanner.nextLine();

int result = str1.compareToIgnoreCase(str2);

if (result == 0) {

System.out.println("The two strings are equal.");

} else if (result < 0) {

System.out.println("The first string is lexicographically smaller.");

} else {

System.out.println("The first string is lexicographically larger.");

}

}

}

ii. to check whether a given string ends with the contents of another string.

PROGRAM:

import java.util.Scanner;

public class stringendswith {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the first string: ");

String str1 = scanner.nextLine();

System.out.print("Enter the second string: ");

String str2 = scanner.nextLine();

if (str1.endsWith(str2)) {

System.out.println("The first string ends with the contents of the second string.");

} else {

System.out.println("The first string does not end with the contents of the second string.");

}

}

}

iii. to print current date and time in the specified format.

PROGRAM:

import java.time.LocalDateTime;

import java.time.format.DateTimeFormatter;

public class currentdatetime {

public static void main(String[] args) {

LocalDateTime currentDateTime = LocalDateTime.now();

DateTimeFormatter formatter = DateTimeFormatter.ofPattern("dd-MM-yyyy HH:mm:ss");

String formattedDateTime = currentDateTime.format(formatter);

System.out.println("Current Date and Time: " + formattedDateTime);

}

}

iv. to get the index of all the characters of the alphabet.

PROGRAM:

import java.util.Scanner;

public class alphabetindex {

public static void main(String[] args) {

String alphabet = "abcdefghijklmnopqrstuvwxyz";

int length = alphabet.length();

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

char c = alphabet.charAt(i);

int index = c - 'a' + 1;

System.out.println(c + " has index " + index);

}

}

}

v. To replace each substring of a given string that matches the given regular

expression with the given replacement. In the below string replace all the fox with

cat.

PROGRAM:

import java.util.Scanner;

public class stringreplace{

public static void main(String[] args) {

String str = "The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.";

String regex = "fox";

String replacement = "cat";

String newStr = str.replaceAll(regex, replacement);

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

System.out.println("New String: " + newStr);

}

}

vi. to get a substring of a given string between two specified positions.

PROGRAM:

import java.util.Scanner;

public class substring {

public static void main(String[] args)

{

String str = "The quick brown fox jumps over the lazy dog.";

String new\_str = str.substring(10, 26);

System.out.println("old = " + str);

System.out.println("new = " + new\_str);

}

}

vii. to trim any leading or trailing whitespace from a given string.

PROGRAM:

import java.util.Scanner;

public class stringtrim{

public static void main(String[] args) {

String str = " Hello, World! ";

String trimmedStr = str.trim();

System.out.println("Original String: \"" + str + "\"");

System.out.println("Trimmed String: \"" + trimmedStr + "\"");

}

}

viii. to convert all the characters in a string to lowercase.

PROGRAM:

import java.util.Scanner;

public class lowercasestring {

public static void main(String[] args) {

String originalString = "This Is a STRING With Mixed CASES";

String lowercaseString = originalString.toLowerCase();

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

System.out.println("Lowercase String: " + lowercaseString);

}

}

ix. to get the length of a given string.

PROGRAM:

import java.util.Scanner;

public class codescracker

{

public static void main(String[] args)

{

String str;

int len=0;

Scanner scan = new Scanner(System.in);

System.out.print("Enter the String: ");

str = scan.nextLine();

char[] strChars = str.toCharArray();

for(char ch: strChars)

len++;

System.out.println("\nLength of String = " +len);

}

}

x. to check whether two String objects contain the same data

Sample string: &quot;The quick brown fox jumps over the lazy dog.&quot;

PROGRAM:

import java.util.Scanner;

public class stringequalitychecker {

public static void main(String[] args) {

String str1 = "The quick brown fox jumps over the lazy dog.";

String str2 = "The quick brown fox jumps over the lazy dog.";

if (str1.equals(str2)) {

System.out.println("The two strings contain the same data.");

} else {

System.out.println("The two strings do not contain the same data.");

}

}

}

2. Implement a class Account. An account has

a balance

functions to add

and withdraw money,

And a function to inquire the current balance.

Condition:

1. Pass a value into a constructor to set an initial balance.

2. If no value is passed the initial balance should be set to $0.

3. Charge a $5 penalty if an attempt is made to withdraw more money than available

in the account.

4. Enhance the Account class to compute interest on the current balance. (10)

PROGRAM:

import java.util.Scanner;

public class account {

private double balance;

public Account() {

this.balance = 0;

}

public Account(double initialBalance) {

this.balance = initialBalance;

}

public void deposit(double amount) {

this.balance += amount;

}

public void withdraw(double amount) {

if (amount > this.balance) {

System.out.println("Insufficient funds. A $5 penalty will be charged.");

this.balance -= 5;

} else {

this.balance -= amount;

}

}

public void inquiry() {

System.out.println("Current balance: $" + this.balance);

}

public void computeInterest(double rate) {

double interest = this.balance \* rate / 100;

this.balance += interest;

}

}

Questions for Debugging a code in Java

3. Given two strings needle and haystack, return the index of the first occurrence of

needle in haystack, or -1 if needle is not part of haystack.

Example 1:

Input: haystack = &quot;sadbutsad&quot;, needle = &quot;sad&quot;

Output: 0

Explanation: &quot;sad&quot; occurs at index 0 and 6.

The first occurrence is at index 0, so we return 0.

PROGRAM:

import java.util.Scanner;

public int strStr(String haystack, String needle) {

if (needle == null || needle.length() == 0) {

return 0;

}

int n = haystack.length();

int m = needle.length();

for (int i = 0; i <= n - m; i++) {

int j;

for (j = 0; j < m; j++) {

if (haystack.charAt(i + j) != needle.charAt(j)) {

break;

}

}

if (j == m) {

return i;

}

}

return -1;

}

Example 2:

Input: haystack = &quot;leetcode&quot;, needle = &quot;leeto&quot;

Output: -1

Explanation: &quot;leeto&quot; did not occur in &quot;leetcode&quot;, so we return -1.

Constraints:

1 &lt;= haystack.length, needle.length &lt;= 104

haystack and needle consist of only lowercase English characters.

PROGRAM:

import java.util.Scanner;

public class solution {

public int strStr(String haystack, String needle) {

int index = haystack.indexOf(needle);

return index;

}

}

Given a string s consisting of words and spaces, return the length of the last word in the

string.

A word is a maximal

substring

consisting of non-space characters only.

Example 1:

Input: s = &quot;Hello World&quot;

Output: 5

Explanation: The last word is &quot;World&quot; with length 5.

Example 2:

Input: s = &quot; fly me to the moon &quot;

Output: 4

Explanation: The last word is &quot;moon&quot; with length 4.

Example 3:

Input: s = &quot;luffy is still joyboy&quot;

Output: 6

Explanation: The last word is &quot;joyboy&quot; with length 6.

Constraints:

1 &lt;= s.length &lt;= 104

s consists of only English letters and spaces &#39; &#39;.

There will be at least one word in s.

PROGRAM:

import java.util.Scanner;

public class lastword{

public int lengthOfLastWord(String s) {

String trimmed = s.trim();

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

return words[words.length - 1].length();

}

}

Questions for Finding error in Java to determine the factor

import java.io.\*;

import java.util.\*;

class factor {

public static void main(String args[]) {

try {

Scanner sc=new Scanner(System.in);

in count=0,n=100,i,j=0,m=4;

int []a=new int [10];

System.out.println(&quot;Enter the number:&quot;);

n=sc.nextInt();

if(n&lt;=0)

{

System.out.println(&quot;Enter valid number&quot;);

}

else {

for(i=1;i&lt;=n;i--);

{

if(n%i!=0)

{

a[j] = i;

System.out.println(&quot;...&quot; + i);

count++;

j++;

}

}

System.out.println(&quot;The number of factors:&quot;+count);

}

System.out.println(m + &quot;th item &quot; + a[m-1]);

}

catch(Exception e) {

System.out.println(&quot;Enter only numbers&quot;);

}

}

}

PROGRAM:

import java.io.\*;

import java.util.\*;

class factor {

public static void main(String args[]) {

try {

Scanner sc=new Scanner(System.in);

int count=0, n=100, i, j=0, m=4;

int[] a=new int[10];

System.out.println("Enter the number:");

n=sc.nextInt();

if(n<=0) {

System.out.println("Enter a valid number");

} else {

for(i=1;i<=n/2;i++) {

if(n%i==0) {

a[j] = i;

System.out.println("..." + i);

count++;

j++;

}

}

System.out.println("The number of factors: " + count);

}

System.out.println(m + "th item: " + a[m-1]);

} catch(Exception e) {

System.out.println("Enter only numbers");

}

}

}