# TASKS

**Write efficient algorithms to solve each of the three tasks below:**

1. Given an array K with N integers from 1 to N+1 such that the array has exactly one integer missing, write a Java function that returns the missing integer.

*e.g. given K = [3,5,4,1], the function should return 2*

Solution:

private static int findMissingElement(int[] array){

int missing = 0;

// first we sort the array in ascending order to get the missing element

Arrays.sort(array); // sorts the array in place, hence no space needed

// next we iterate the sorted array checking if the difference between neighbouring elements is one

for (int i = 0; i < array.length-1; i++) {

if (array[i+1] - array[i] > 1){

missing = array[i] + 1;

break;

}

}

return missing;

}

1. Given a string S of length N, write a Java function that transforms the string by reversing characters in groups of four, and returns the transformed string.

*e.g. when S = 'Lorem at' the output should be 'eroLta m' when S = ' Tempor ip' the output should be 'meT roppi'*

Solution:

import java.util.ArrayList;

import java.util.List;

public class StringPartsReversal {

public static void main(String[] args) {

String text = "Hello world";

System.out.println("Reversed: "+reverseText(text));

System.out.println("Reversed parts: "+reverseTextByParts(text));

}

private static String reverseTextByParts(String text){

// create a list that will hold the string parts of length 4 or less

List<String> parts = new ArrayList<>();

int index = 0;

// iterate the string while splitting it on the 4th index and store in the list

while (index < text.length()){

parts.add(text.substring(index, Math.min(index+4, text.length())));

index += 4; // increment index by 4

}

/\*

now that we have the string parts to be reversed,

let's call the reverse function on each part while appending the result to the string builder

\*/

StringBuilder builder = new StringBuilder();

for (String item: parts) {

builder.append(reverseText(item));

}

return builder.toString();

}

private static String reverseText(String text){

StringBuilder builder = new StringBuilder();

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

builder.append(text.charAt(i));

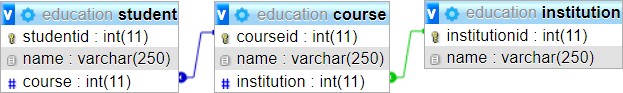
}

return builder.toString();

}

}

1. You're given a database (education) with three tables (student, course, institution) as illustrated below:

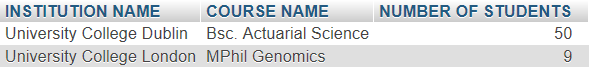


* 1. Use the diagram above to create the tables (the database, table and column names should remain as indicated)

CREATE TABLE student ( studentid INT, name VARCHAR(250), course INT PRIMARY KEY(studentid));

CREATE TABLE course(courseid INT, name VARCHAR(250), institution INT, PRIMARY KEY (courseid), FOREIGN KEY (courseid)REFERENCES student(course);

CREATE TABLE institution(institutionid INT, name VARCHAR(250), FOREIGN KEY (institutionid) REFERENCES course(institution);

* 1. Write a MySQL query that will display the number of students per course per institution in the format below

SELECT institution.name AS `INSTITUITION NAME`, course.name AS `COURSE NAME`, COUNT(studentid) AS `NUMBER OF STUDENTS`

FROM course

INNER JOIN student ON course.courseid = student.course

INNER JOIN institution ON institution.institutionid=course.institution

ORDER BY `NUMBER OF STUDENTS` DESC ;