Realizar en WORD

1) Escribir el método

Int ocurrencias(int buscado, int[] arreglo)

Que devuelva la cantidad de ocurrencias de “buscado” en “arreglo”

**package** Principal;

**import** java.util.Scanner;

**import** javax.swing.JOptionPane;

**public** **class** main {

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

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

**int** cantidad = Integer.*parseInt*(JOptionPane.*showInputDialog*("Ingrese la cantidad de numeros: "));

**int**[] arreglo = **new** **int**[cantidad];

**int** buscado = Integer.*parseInt*(JOptionPane.*showInputDialog*("Ingrese el elemento buscado: "));

**int** existe = 0;

System.***out***.println("Ingrese los " + cantidad + " numeros: ");

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

arreglo[i] = Integer.*parseInt*(entrada.nextLine());

}

existe = *ocurrencias*(buscado, arreglo, cantidad);

}

**private** **static** **int** ocurrencias(**int** buscado, **int**[] arreglo, **int** cantidad) {

**int** cant = 0;

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

**if** (arreglo[i] == buscado) {

cant = cant + 1;

}

}

System.***out***.println("El elemento buscado aparecio: " + cant + " veces");

**return** cant;

}

}

2) Escribir un programa en Java que obtenga e imprima a partir de un arreglo de enteros la cantidad de veces que aparece cada elemento usando un asterisco para contar cada aparición del elemento.

En el resultado, los elementos deben estar ordenados de menor a mayor.

Para el ejercicio considerar elementos entre el 1 y el 9.

Por ejemplo, para el arreglo:

int[] myArray = int[] { 8,1,6,7,8,4,9,8,2 }; el resultado se vería así:  
  
1: \*

2: \*

3:

4: \*

5:

6: \*

7: \*

8: \*\*\*

9: \*

package Principal;

import java.util.Arrays;

import java.util.HashSet;

import java.util.Iterator;

import java.util.Scanner;

import java.util.Set;

import javax.swing.JOptionPane;

public class main {

public static void main(String[] args) {

Scanner entrada = new Scanner(System.in);

int cantidad = Integer.parseInt(JOptionPane.showInputDialog("Ingrese la cantidad de numeros: "));

int[] arreglo = new int[cantidad];

int[] arregloAComparar = new int[cantidad];

int buscado = 0;

System.out.println("Ingrese los " + cantidad + " numeros: ");

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

arreglo[i] = Integer.parseInt(entrada.nextLine());

}

System.out.println("Ingrese los " + cantidad + " Elementos a comparar ");

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

arregloAComparar[i] = Integer.parseInt(entrada.nextLine());

}

Arrays.sort(arregloAComparar);

Set<Integer> listaFinal = new HashSet<>();

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

listaFinal.add(arregloAComparar[i]);

}

System.out.println("La lista final es");

for (Integer i : listaFinal) {

buscado = i;

ocurrencias(buscado, arreglo, cantidad);

}

}

private static void ocurrencias(int buscado, int[] arreglo, int cantidad) {

boolean esta = false;

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

if (arreglo[i] == buscado) {

esta = true;

}

}

if (esta) {

System.out.println(buscado + " \*");

} else {

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

}

}

}

3) Escribir una script en T-SQL que obtenga un listado de empleados y su nivel educativo con la condición de que se encuentre en una gerencia donde la sumatoria de sueldos de los empleados de dicha gerencia sea mayor a $430.000

Ordenar los resultados por apellido de forma ascendente.

Tablas:  
TBL\_empleados (id, nombre, apellido, id\_gerencia, salario, educacion\_id)  
TBL\_gerencias (id, gerencia, ciudad)  
TBL\_nivel\_educacion (id, descripcion)

USE challenge;

CREATE TABLE TBL\_gerencias (

id INTEGER PRIMARY KEY,

gerencia TEXT NOT NULL,

ciudad TEXT NOT NULL

);

CREATE TABLE TBL\_nivel\_educacion (

id INTEGER PRIMARY KEY,

descripcion TEXT NOT NULL

);

CREATE TABLE TBL\_empleados (

id INTEGER PRIMARY KEY,

nombre TEXT NOT NULL,

apellido TEXT NOT NULL,

salario DEC NOT NULL,

idGerencia INTEGER NOT NULL,

idEducacion INTEGER NOT NULL,

FOREIGN KEY (idGerencia) REFERENCES TBL\_gerencias (id),

FOREIGN KEY (idEducacion) REFERENCES TBL\_nivel\_educacion (id)

);

INSERT INTO TBL\_gerencias VALUES (1, 'Gerencia 1', 'Mendoza');

INSERT INTO TBL\_gerencias VALUES (2, 'Gerencia 2', 'Salta');

INSERT INTO TBL\_gerencias VALUES (3, 'Gerencia 3', 'Entre Rios');

INSERT INTO TBL\_nivel\_educacion VALUES (1, 'Primario');

INSERT INTO TBL\_nivel\_educacion VALUES (2, 'Secundario');

INSERT INTO TBL\_nivel\_educacion VALUES (3, 'Terciario');

INSERT INTO TBL\_empleados VALUES (1, 'JUAN', 'PEREZ', 405, 1, 1);

INSERT INTO TBL\_empleados VALUES (2, 'MARCOS', 'SOSA', 605, 1, 2);

INSERT INTO TBL\_empleados VALUES (3, 'JUAN', 'GOMEZ', 432, 2, 3);

SELECT tbl\_empleados.nombre, tbl\_empleados.apellido,

tbl\_nivel\_educacion.descripcion

FROM tbl\_empleados

JOIN tbl\_nivel\_educacion on idEducacion = tbl\_nivel\_educacion.id WHERE tbl\_empleados.idGerencia

IN (SELECT tbl\_empleados.idGerencia FROM tbl\_empleados GROUP BY tbl\_empleados.idGerencia HAVING SUM(tbl\_empleados.salario) > 430)

ORDER BY tbl\_empleados.apellido;

4) Escribir un diagrama de clases que represente el mundo animal teniendo en cuenta a los siguientes ejemplares:

**gato**, **águila**, **perro**, **león**, ***oso***, **gaviota, jirafa, búho, tigre**



