

Tarea de la Unidad Didáctica 10 Antonio Jiménez Sevilla

1. Escribe un programa que muestre la información de todos los alumnos que hay en la base de datos que estén matriculados en el ciclo de grado superior conocido como "DAM". Haz esta consulta mediante QBE.

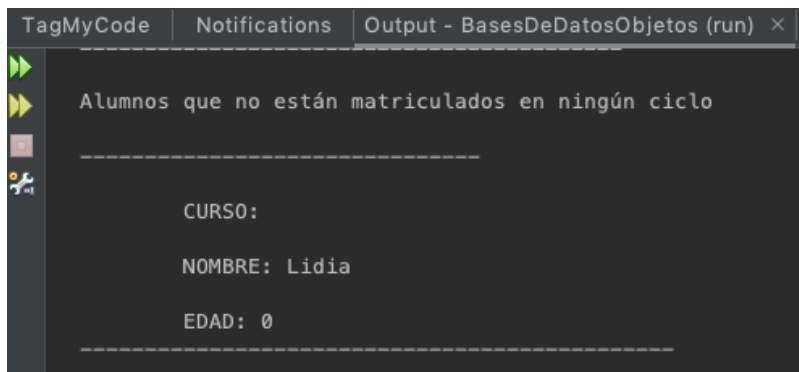
```
ObjectContainer baseDatos9 = Db4oEmbedded.openFile("BDAlumEmp");
Alumno a9 = new Alumno("DAM", null, 0);
System.out.println("La información de los alumnos de Dam es: \n");
ObjectSet result2 = baseDatos9.queryByExample(a9);
while (result2.hasNext()) {
    a9 = (Alumno) result2.next();
    a9.visualAlumno();
}
baseDatos9.close();

System.out.println("=====\n");
```

```
La información de los alumnos de Dam es:
-----
CURSO: DAM
NOMBRE: Fatima
EDAD: 23
-----
CURSO: DAM
NOMBRE: Miguel
EDAD: 0
=====
```

2. Escribe un programa que muestre la información de todos los alumnos que haya en la base de datos que no estén matriculados en ningún ciclo. Los alumnos que no estén matriculados en ningún ciclo tendrán como nombre del ciclo "". Haz esta consulta mediante QBE

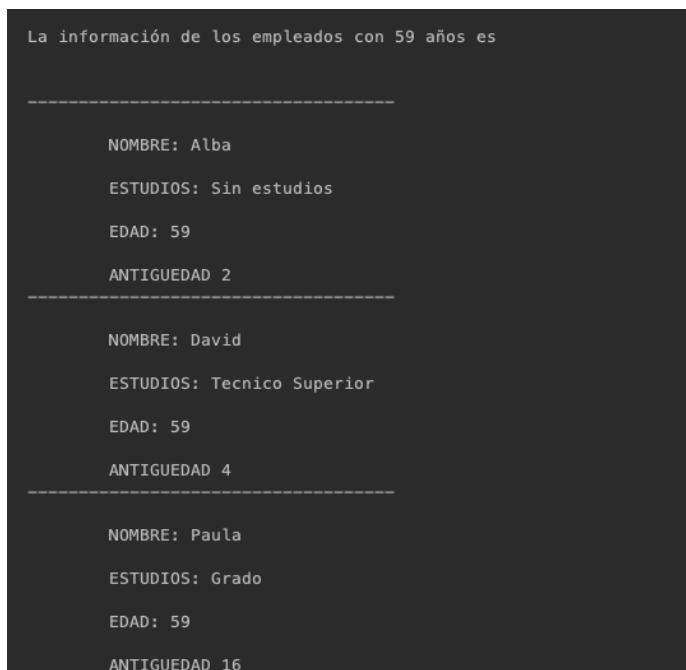
```
52 System.out.println("=====\n");
53
54 ObjectContainer baseDatos2 = Db4oEmbedded.openFile("BDAlumEmp");
55 Alumno a1 = new Alumno("", null, 0);
56 // Empleado e1 = new Empleado(null, null, 0, 0);
57 System.out.println("Alumnos que no están matriculados en ningún ciclo\n");
58 ObjectSet result = baseDatos2.queryByExample(a1);
59 while (result.hasNext()) {
60     a1 = (Alumno) result.next();
61     a1.visualAlumno();
62 }
63 baseDatos2.close();
```



```
TagMyCode | Notifications | Output - BasesDeDatosObjetos (run) X
Alumnos que no están matriculados en ningún ciclo
-----
CURSO:
NOMBRE: Lidia
EDAD: 0
-----
```

3. Escribe un programa que muestre la información de todos los empleados que haya en la base de datos que tengan 59 años. Haz esta consulta mediante QBE.

```
// En el siguiente ejercicio mostramos todos los empleados que haya en la
// base de datos que tengan 59 años
ObjectContainer baseDatos10 = Db4oEmbedded.openFile("BDA\lumEmp");
Empleado e3 = new Empleado(null, null, 59, 0);
System.out.println("La información de los empleados con 59 años es\n\n");
ObjectSet result5 = baseDatos10.queryByExample(e3);
while (result5.hasNext()) {
    e3 = (Empleado) result5.next();
    e3.visualEmpleado();
}
baseDatos10.close();
```



```
La información de los empleados con 59 años es
-----
NOMBRE: Alba
ESTUDIOS: Sin estudios
EDAD: 59
ANTIGUEDAD 2
-----
NOMBRE: David
ESTUDIOS: Tecnico Superior
EDAD: 59
ANTIGUEDAD 4
-----
NOMBRE: Paula
ESTUDIOS: Grado
EDAD: 59
ANTIGUEDAD 16
```

4. Muestra los datos de los empleados que están entre 30 y 45 años

```
System.out.println("=====\\n");
System.out.println("Empleados entre 30 y 45 años \\n\\n");
Empleado emplConsultado;
ObjectSet inforSolic;
Query todo2;
ObjectContainer baseDatos5 = Db4oEmbedded.openFile("BDA\\umEmp");
todo2 = baseDatos5.query();
todo2.constrain(Empleado.class);
todo2.descend("edad").constrain(30).greater();
todo2.descend("edad").constrain(45).smaller(); // doble restricción
inforSolic = todo2.execute();
while (inforSolic.hasNext()) {
    emplConsultado = (Empleado) inforSolic.next();
    emplConsultado.visualEmpleado();
}
baseDatos5.close();
```

Empleados entre 30 y 45 años

NOMBRE: Pedro

ESTUDIOS: Sin estudios

EDAD: 42

ANTIGUEDAD 2

NOMBRE: Alberto

ESTUDIOS: Tecnico Superior

EDAD: 39

ANTIGUEDAD 4

=====

5. Escribe un programa que muestre los datos de los empleados cuya edad no esté entre 30 y 45 años. Muéstralos de forma ascendente por la edad.

```
System.out.println("=====\\n");
System.out.println("Datos de los empleados que no están entre 30 y 45 años \\n");
Empleado emplConsultado2;
ObjectSet inforSolic3;
Query todo3;
ObjectContainer baseDatos6 = Db4oEmbedded.openFile("BDA\\umEmp");
todo3 = baseDatos6.query();
todo3.constrain(Empleado.class);
todo3.descend("edad").constrain(30).greater();
todo3.descend("edad").constrain(45).smaller().not();
inforSolic3 = todo3.execute();
while (inforSolic3.hasNext()) {
    emplConsultado2 = (Empleado) inforSolic3.next();
    emplConsultado2.visualEmpleado();
}
baseDatos6.close();
```

Datos de los empleados que no están entre 30 y 45 años

NOMBRE: Juan Luis

ESTUDIOS: Grado

EDAD: 53

ANTIGUEDAD 25

NOMBRE: Juanjo

ESTUDIOS: Sin estudios

EDAD: 62

ANTIGUEDAD 5

NOMBRE: Juanjo

ESTUDIOS: Sin estudios

EDAD: 62

ANTIGUEDAD 5

NOMBRE: Alba

ESTUDIOS: Sin estudios

EDAD: 59

ANTIGUEDAD 2

NOMBRE: David

ESTUDIOS: Tecnico Superior

EDAD: 59

ANTIGUEDAD 4

NOMBRE: Paula

ESTUDIOS: Grado

EDAD: 59

ANTIGUEDAD 16

6. Muestra los datos de los empleados que llevan un mínimo de cuatro años en la empresa y son técnicos superiores o llevan más de 15 años en la empresa

```
System.out.println("=====");
System.out.println("Datos de los empleados que llevan un mínimo de 4 años en la empresa y son técnicos superiores");
System.out.println("o llevan más de 15 años en la empresa \n");
Empleado emplConsultado3;
ObjectSet inforSolic4;
Query todo4;
ObjectContainer baseDatos7 = Db4oEmbedded.openFile("BDA\lumEmp");
todo4 = baseDatos7.query();
todo4.constrain(Empleado.class);
Constraint segundCond = todo4.descend("antiguedad").constrain(3).greater().and(todo4.descend("estudios").constrain("Tecnico Superior"));
todo4.descend("antiguedad").constrain(15).greater().or(segundCond);
inforSolic4 = todo4.execute();
while (inforSolic4.hasNext()) {
    emplConsultado3 = (Empleado) inforSolic4.next();
    emplConsultado3.visualEmpleado();
}
baseDatos7.close();
```

```
=====
Datos de los empleados que llevan un mínimo de 4 años en la empresa y son técnicos superiores
o llevan más de 15 años en la empresa

=====
NOMBRE: Juan Luis
ESTUDIOS: Grado
EDAD: 53
ANTIGUEDAD 25
=====
NOMBRE: David
ESTUDIOS: Tecnico Superior
EDAD: 59
ANTIGUEDAD 4
=====
NOMBRE: Paula
ESTUDIOS: Grado
EDAD: 59
ANTIGUEDAD 16
=====
NOMBRE: Alberto
ESTUDIOS: Tecnico Superior
EDAD: 39
ANTIGUEDAD 4
```

7. Escribe un programa que modifique la antigüedad de los empleados que tienen más de cincuenta años, aumentando su antigüedad en cinco años.

```
System.out.println("Empleados que tienen mas de 50 años");
Empleado emplModificado;
ObjectSet inforSolic2;
Query todo12;
ObjectContainer baseDatos12 = Db4oEmbedded.openFile("BDA\\umEmp");
todo12 = baseDatos12.query();
todo12.constrain(Empleado.class);
todo12.descend("edad").constrain(50).greater();
inforSolic = todo12.execute();
while (inforSolic.hasNext()) {
    emplModificado = (Empleado)inforSolic.next();
    emplModificado.visualEmpleado();
}

inforSolic = todo12.execute();
System.out.println("Ahora le vamos a aumentar la antigüedad en 5 años\n");
while (inforSolic.hasNext()) {
    emplModificado = (Empleado)inforSolic.next();
    emplModificado.incAntigüedad(5); // se le incrementa la antigüedad en 5 años
    baseDatos12.store(emplModificado);
    emplModificado.visualEmpleado();
}

System.out.println("Ahora mostramos empleados modificados\n");
inforSolic = todo12.execute();
while (inforSolic.hasNext()) {
    emplModificado = (Empleado)inforSolic.next();
    emplModificado.visualEmpleado();
}

baseDatos12.close();
System.out.println("=====\\n");
```

Creo un método para aumentar la antigüedad.

```
package basesdedatosobjetos;

/**
 *
 * @author Antonio
 */
public class Empleado {
    String nombre;
    String estudios;
    int edad;
    int antigüedad;

    public Empleado(String nom, String estudios, int edad, int antigüedad) {
        this.nombre = nom;
        this.estudios = estudios;
        this.edad = edad;
        this.antigüedad = antigüedad;
    }

    void incAntigüedad (int incremento) {
        this.antigüedad += incremento;
    }

    void visualEmpleado() {
        System.out.println("-----");
        System.out.println("\n\tNOMBRE: " + nombre);
        System.out.println("\n\tESTUDIOS: " + estudios);
        System.out.println("\n\tEDAD: " + edad);
        System.out.println("\n\tANTIGÜEDAD " + antigüedad);
    }
}
```

```

ANTIGUEDAD 4
Empleados que tienen mas de 50 años
-----

NOMBRE: Juan Luis
ESTUDIOS: Grado
EDAD: 53
ANTIGUEDAD 25
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 5
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 5
-----

NOMBRE: Alba
ESTUDIOS: Sin estudios
EDAD: 59
ANTIGUEDAD 2
-----

NOMBRE: David
ESTUDIOS: Tecnico Superior
EDAD: 59
ANTIGUEDAD 4
-----

NOMBRE: Paula
ESTUDIOS: Grado
EDAD: 59
ANTIGUEDAD 16

```

```

ANTIGUEDAD 10
Ahora le vamos a aumentar la antigüedad en 5 años
-----

NOMBRE: Juan Luis
ESTUDIOS: Grado
EDAD: 53
ANTIGUEDAD 30
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 10
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 10
-----

NOMBRE: Alba
ESTUDIOS: Sin estudios
EDAD: 59
ANTIGUEDAD 7
-----

NOMBRE: David
ESTUDIOS: Tecnico Superior
EDAD: 59
ANTIGUEDAD 9
-----

NOMBRE: Paula
ESTUDIOS: Grado
EDAD: 59
ANTIGUEDAD 21

```

```

ANTIGUEDAD 21
Ahora mostramos empleados modificados
-----

NOMBRE: Juan Luis
ESTUDIOS: Grado
EDAD: 53
ANTIGUEDAD 30
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 10
-----

NOMBRE: Juanjo
ESTUDIOS: Sin estudios
EDAD: 62
ANTIGUEDAD 10
-----

NOMBRE: Alba
ESTUDIOS: Sin estudios
EDAD: 59
ANTIGUEDAD 7
-----

NOMBRE: David
ESTUDIOS: Tecnico Superior
EDAD: 59
ANTIGUEDAD 9
-----

NOMBRE: Paula
ESTUDIOS: Grado
EDAD: 59
ANTIGUEDAD 21
=====
=====

```

8. Escribe un programa que borre de la base de datos a todos los alumnos que estén matriculados en el "DAM".

```
System.out.println("=====\\n\\n\\n\\n\\n");
System.out.println("Ahora mostramos todos los alumnos que hay en la base de datos\\n");

ObjectContainer baseDatos11 = Db4oEmbedded.openFile("BDAalumEmp");
Alumno a11 = new Alumno(null, null, 0);
ObjectSet result3 = baseDatos11.queryByExample(a11);

while(result3.hasNext()) {
    a11 = (Alumno)result3.next();
    a11.visualAlumno();
}

System.out.println("Alumnos que están matriculados en DAM");

a11 = new Alumno("DAM", null, 0);

System.out.println("Ahora vamos a borrar los alumnos que estan matriculados en DAM\\n");
result3 = baseDatos11.queryByExample(a11);
while(result3.hasNext()) {
    a11 = (Alumno)result3.next();
    a11.visualAlumno();
    baseDatos11.delete(a11);
}

// Ahora vamos a borrar los alumnos que están en el DAM
System.out.println("Ahora mostramos los alumnos que quedan en la base de datos\\n");
a11 = new Alumno(null, null, 0);
result3 = baseDatos11.queryByExample(a11);

while(result3.hasNext()) {
    a11 = (Alumno)result3.next();
    a11.visualAlumno();
}
baseDatos11.close();
}
```

Ahora mostramos todos los alumnos que hay en la base de datos

CURSO: DAM
NOMBRE: Fatima
EDAD: 23

CURSO: DAM
NOMBRE: Miguel
EDAD: 0

CURSO:
NOMBRE: Lidia
EDAD: 0

CURSO: ASIR
NOMBRE: Sergio
EDAD: 25

Alumnos que están matriculados en DAM
Ahora vamos a borrar los alumnos que estan matriculados en DAM

CURSO: DAM
NOMBRE: Fatima
EDAD: 23

CURSO: DAM
NOMBRE: Miguel
EDAD: 0

Ahora mostramos los alumnos que quedan en la base de datos

CURSO:
NOMBRE: Lidia
EDAD: 0

CURSO: ASIR
NOMBRE: Sergio
EDAD: 25

BUILD SUCCESSFUL (total time: 0 seconds)