

Actividades de aplicación. Realizar las siguientes 14.11, 14.12, 14.13, 14.14 y 14.15.

14.11. Crea un programa que muestre todos los empleados.

```
4 public class Aa14_11 {
5
6
7 public static void main(String[] args) {
8     try {
9         Connection con = DriverManager.getConnection(url: "jdbc:mysql://127.0.0.1/empresa",
10             user: "Pepe", password: "12345");
11         String consulta = "SELECT * FROM empleados";
12         Statement sentencia = con.createStatement();
13         ResultSet rs = sentencia.executeQuery(string: consulta);
14
15         while(rs.next()){
16             System.out.println(x: "*****");
17             System.out.println("Num empleado: " + rs.getInt(string: "numemp"));
18             System.out.println("Nombre: " + rs.getString(string: "nombre"));
19             System.out.println("Edad: " + rs.getInt(string: "edad"));
20             System.out.println("Oficina: " + rs.getInt(string: "oficina"));
21             System.out.println("Puesto: " + rs.getString(string: "puesto"));
22             System.out.println("Contrato: " + rs.getDate(string: "contrato"));
23             System.out.println(x: "*****");
24         }
25         con.close();
26     } catch (SQLException ex) {
27         System.out.println(x: ex);
28     }
29 }
30 }
```

Output X

Debugger Console X Aa14_11 (run) X

Run:

```
*****
Num empleado: 101
Nombre: Antonio Alondra
Edad: 45
Oficina: 12
Puesto: Representante
Contrato: 2016-10-20
*****
*****
Num empleado: 102
Nombre: Beatriz Bóho
Edad: 33
Oficina: 12
Puesto: Director ventas
Contrato: 2017-05-19
*****
*****
Num empleado: 103
```

```

Num empleado: 103
Nombre: Claudia Cisne
Edad: 29
Oficina: 12
Puesto: Representante
Contrato: 2017-03-01
*****
Num empleado: 104
Nombre: Daniel Dingo
Edad: 21
Oficina: 21
Puesto: Administrativo
Contrato: 2015-12-10
*****
Num empleado: 105
Nombre: Ernesto Elefante
Edad: 37
Oficina: 13
Puesto: Gerente
Contrato: 2018-02-12
*****
Num empleado: 106
Nombre: Francisco Foca
Edad: 52
Oficina: 11
Puesto: Recepcionista
Contrato: 2018-06-14
*****
Num empleado: 107
Nombre: Guillermo Gato
Edad: 19
Oficina: 22
Puesto: Becario
Contrato: 2011-11-14
*****
Num empleado: 108
Nombre: Helena Halcón
Edad: 62
Oficina: 21
Puesto: Directora general
Contrato: 2019-10-12
*****

```

```

Num empleado: 109
Nombre: Isabel Ibis
Edad: 31
Oficina: 11
Puesto: Representante
Contrato: 2019-10-12
*****
Num empleado: 110
Nombre: Julia Jirafa
Edad: 41
Oficina: 0
Puesto: Auxiliar de servicios
Contrato: 2020-01-13
*****
Num empleado: 111
Nombre: Kenia Koala
Edad: 23
Oficina: 31
Puesto: Recepcionista
Contrato: 2020-01-03
*****
Num empleado: 112
Nombre: Lorena Loro
Edad: 27
Oficina: 31
Puesto: Director
Contrato: 2020-01-04
*****
Num empleado: 113
Nombre: María Marmota
Edad: 52
Oficina: 31
Puesto: Comercial
Contrato: 2021-01-05
*****
Num empleado: 114
Nombre: Nadia Narval
Edad: 29
Oficina: 32
Puesto: Administrativo
Contrato: 2021-01-07
*****

```

```

Nombre: Maki Ñu
Edad: 18
Oficina: 24
Puesto: Comercial
Contrato: 2021-02-03
*****
Num empleado: 116
Nombre: Oscar Oso
Edad: 30
Oficina: 24
Puesto: Auxiliar de servicios
Contrato: 2019-03-03
*****
Num empleado: 117
Nombre: Pedro Pelicano
Edad: 44
Oficina: 11
Puesto: Comercial
Contrato: 2019-04-03
*****
Num empleado: 118
Nombre: Qi Quebrantahuesos
Edad: 33
Oficina: 11
Puesto: Administrativo
Contrato: 2018-05-03
*****
Num empleado: 119
Nombre: Raquel Ratón
Edad: 43
Oficina: 12
Puesto: Representante
Contrato: 2017-06-03
*****
Num empleado: 120
Nombre: Sonia Salmón
Edad: 60
Oficina: 22
Puesto: Administrativo
Contrato: 2017-07-03
*****
BUILD SUCCESSFUL (total time: 0 seconds)

```

14.12. Utiliza la técnica de mapeo objeto-relacional para cargar todas las oficinas de la base de datos en una lista de oficinas. Muestra el contenido de la lista de objetos creada.

```
1 package aa14_12;
2
3 import java.sql.*;
4 import java.util.ArrayList;
5 import java.util.List;
6
7 public class Aa14_12 {
8
9     public static void main(String[] args) {
10         List<Oficina> oficinas = new ArrayList();
11
12         try {
13             Connection con = DriverManager.getConnection(uri:"jdbc:mysql://127.0.0.1/empresa",
14                 user:"Pepe", password:"12345");
15             String consulta = "SELECT * FROM oficinas";
16             Statement sentencia = con.createStatement();
17             ResultSet rs = sentencia.executeQuery(consulta);
18
19             while (rs.next()) {
20                 oficinas.add(new Oficina(rs.getInt("oficina"),
21                     rs.getString("ciudad"),
22                     rs.getInt("superficie"),
23                     rs.getDouble("ventas")));
24             }
25             con.close();
26             for (Oficina ofi : oficinas) {
27                 System.out.println(ofi);
28             }
29         } catch (SQLException ex) {
30             System.out.println(ex);
31         }
32     }
33 }
```

```
1 package aa14_12;
2
3 public class Oficina {
4     int oficina;
5     String ciudad;
6     int superficie;
7     double ventas;
8
9     public Oficina(int oficina, String ciudad, int superficie, double ventas) {
10         this.oficina = oficina;
11         this.ciudad = ciudad;
12         this.superficie = superficie;
13         this.ventas = ventas;
14     }
15
16     @Override
17     public String toString() {
18         return "Oficina{" + "oficina=" + oficina + ", ciudad=" + ciudad + ", "
19             + "superficie=" + superficie + ", ventas=" + ventas + "}";
20     }
21
22 }
23
24
25 }
```

run:

```
Oficina{oficina=11, ciudad=Sevilla, superficie=100, ventas=10000.0}
Oficina{oficina=12, ciudad=C diz, superficie=100, ventas=8500.0}
Oficina{oficina=13, ciudad=La Coru a, superficie=350, ventas=1300.0}
Oficina{oficina=21, ciudad=Badajoz, superficie=200, ventas=5060.0}
Oficina{oficina=22, ciudad=La Coru a, superficie=180, ventas=1234.56}
Oficina{oficina=23, ciudad=Murcia, superficie=350, ventas=1000.5}
Oficina{oficina=24, ciudad=Oviedo, superficie=500, ventas=7500.0}
Oficina{oficina=31, ciudad=Zamora, superficie=200, ventas=15000.15}
Oficina{oficina=32, ciudad= lmeria, superficie=200, ventas=0.0}
BUILD SUCCESSFUL (total time: 0 seconds)
```

14.13. Modifica la Actividad de aplicación 14.12 para que la aplicación solicite al usuario el nombre de una ciudad, y que muestre las oficinas ubicadas en dicha ciudad.

```
1 package aal4_13;
2
3
4 import java.sql.*;
5 import java.util.*;
6
7 public class Aal4_13 {
8
9     public static void main(String[] args) {
10         Scanner sc = new Scanner(System.in);
11         String city;
12         try {
13             Connection con = DriverManager.getConnection("jdbc:mysql://127.0.0.1/empresa",
14                 "Pepe", "password123456");
15             System.out.println("Introduzca ciudad para mostrar sus oficinas: ");
16             city = sc.next();
17             String consulta = "SELECT * FROM oficinas WHERE ciudad LIKE ?";
18             PreparedStatement sentencia = con.prepareStatement(consulta);
19             sentencia.setString(1, "%" + city + "%");
20             ResultSet rs = sentencia.executeQuery();
21
22             while (rs.next()) {
23                 System.out.println(new Oficina(rs.getInt("oficina"),
24                     rs.getString("ciudad"),
25                     rs.getInt("superficie"),
26                     rs.getDouble("ventas")));
27             }
28             con.close();
29         } catch (SQLException ex) {
30             System.out.println(ex);
31         }
32     }
33 }
34
```

```
1 package aal4_13;
2
3 public class Oficina {
4     int oficina;
5     String ciudad;
6     int superficie;
7     double ventas;
8
9     public Oficina(int oficina, String ciudad, int superficie, double ventas) {
10         this.oficina = oficina;
11         this.ciudad = ciudad;
12         this.superficie = superficie;
13         this.ventas = ventas;
14     }
15
16     @Override
17     public String toString() {
18         return "Oficina[" + "oficina=" + oficina + ", ciudad=" + ciudad + ", "
19             + "superficie=" + superficie + ", ventas=" + ventas + "]\n";
20     }
21
22 }
23
24
25
```

Output x

Debugger Console x Aal4_13 (run) x

run:

Introduzca ciudad para mostrar sus oficinas:

Murcia

Oficina(oficina=13, ciudad=Murcia, superficie=350, ventas=1000.5)

BUILD SUCCESSFUL (total time: 3 seconds)

14.14. Muestra un listado con el nombre y la edad de los empleados cuya edad se encuentra comprendido entre unos valores máximos y mínimos que introducirá el usuario.

```
1 package aal4_14;
2
3 import java.sql.*;
4 import java.util.*;
5
6 public class Aal4_14 {
7
8     public static void main(String[] args) {
9         Scanner sc = new Scanner(System.in);
10         int max, min;
11         try {
12             Connection con = DriverManager.getConnection("jdbc:mysql://127.0.0.1/empresa",
13                 user: "Pepe", password: "12345");
14             System.out.println("Introduzca edad minima: ");
15             min = sc.nextInt();
16             System.out.println("Introduzca edad maxima: ");
17             max = sc.nextInt();
18             String consulta = "SELECT nombre, edad FROM empleados "
19                 + "WHERE ? < edad AND ? > edad";
20             PreparedStatement sentencia = con.prepareStatement(consulta);
21             sentencia.setInt(1, min);
22             sentencia.setInt(2, max);
23             ResultSet rs = sentencia.executeQuery();
24
25             while(rs.next()){
26                 System.out.println("*****");
27                 System.out.println("Nombre: " + rs.getString("nombre"));
28                 System.out.println("Edad: " + rs.getInt("edad"));
29                 System.out.println("*****");
30             }
31
32             con.close();
33         } catch (SQLException ex) {
34             System.out.println(ex);
35         }
36     }
37 }
```

```
run:
Introduzca edad minima:
30
Introduzca edad maxima:
40
*****
Nombre: Beatriz Boho
Edad: 33
*****
Nombre: Ernesto Elefante
Edad: 37
*****
Nombre: Isabel Ibis
Edad: 31
*****
Nombre: Qi Quebrantahuesos
Edad: 33
*****
BUILD SUCCESSFUL (total time: 21 seconds)
```

14.15. Crea un programa que lea todos los datos de un empleado, excepto la fecha de contratación que será la de hoy, y los inserte en la base de datos.

Antes de la ejecución:

numemp	nombre	edad	oficina	puesto	contrato
101	Antonio Alondra	45	12	Representante	2016-10-20
102	Beatriz Búho	33	12	Director ventas	2017-05-19
103	Claudia Cisne	29	12	Representante	2017-03-01

```
1 package aal4_15;
2
3 import java.sql.*;
4 import java.util.*;
5
6 public class Aal4_15 {
7
8     public static void main(String[] args) {
9         Scanner sc = new Scanner(System.in);
10         int max, min;
11         try {
12             Connection con = DriverManager.getConnection(url:"jdbc:mysql://127.0.0.1/empresa",
13                 user:"Pepe", password:"12345");
14
15             String consulta = "UPDATE empleados SET contrato = CURRENT_DATE() "
16                 + "WHERE numemp = 101";
17             Statement sentencia = con.createStatement();
18
19             int modificados = sentencia.executeUpdate(string:consulta);
20             System.out.println("Se han modificado " + modificados + " elementos.");
21             con.close();
22         } catch (SQLException ex) {
23             System.out.println(x: ex);
24         }
25     }
26 }
27
28 }
29
```

Output - Aa14_15 (run) x

```
run:
Se han modificado 1 elementos.
BUILD SUCCESSFUL (total time: 0 seconds)
```

Después de la actualización:

numemp	nombre	edad	oficina	puesto	contrato
101	Antonio Alondra	45	12	Representante	2024-03-21
102	Beatriz Búho	33	12	Director ventas	2017-05-19
103	Claudia Cisne	29	12	Representante	2017-03-01