**Integración de habilidades Java-jdbc-MySQL**

A continuación, realizaremos una variedad de ejercicios, con el objetivo de integrar los conocimientos en la tecnología de acceso a datos con java y jdbc. Se trata principalmente de ejercicios simples (no aplicaciones) donde haremos usos exhaustivos de la creación de métodos aplicados a la librería jdbc.

**Importante. Pautas para entregar el trabajo:**

La actividad se puede realizar de forma grupal, cada integrante del grupo debe tener una copia del trabajo para defenderlo de forma **individual**.

Para cada ejercicio se debe realizar una captura de pantalla y pegarla a continuación de cada consigna en este documento.

La defensa del trabajo comienza el 16/06 hasta 30/6 inclusive. Teniendo en cuenta que este trabajo compone la nota del cuatrimestre, recomiendo no exceder el pazo de entrega.

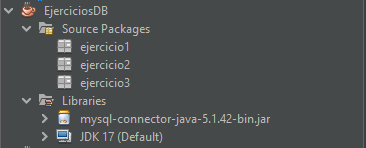
Para su mayor entendimiento, se recomienda comentar todas las líneas de códigos que se consideren pertinentes.

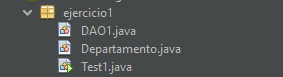
Les recuerdo que la entrega del trabajo es de carácter **OBLIGATORIO PARA TODO EL CURSO SIN EXCEPCIÓN.**

**Pautas para el desarrollo de las actividades**

* Utilizar para todos los ejercicios que vienen a continuación el script demodb.SQL
* Realizar un análisis de las tablas, atributos, datos, relaciones y restricciones (PK, FK) de la base de datos.
* Crear un proyecto (EjerciciosDB), y agrupar a los ejercicios en paquetes (ejercicio1, ejercicio2, . . .,ejercicioN).

Ejemplo de como debe quedar el proyecto



**Ejercicio N°1**Crear un método para visualizar, número y nombre de todos los departamentos, el método devuelve un ArrayList de objetos departamentos. Se recomienda utilizar tres clases, una para mapear los atributos de la tabla departamentos, una que contenga los métodos, conectar y ejecutar la consulta, y otra con el método main para realizar los test.

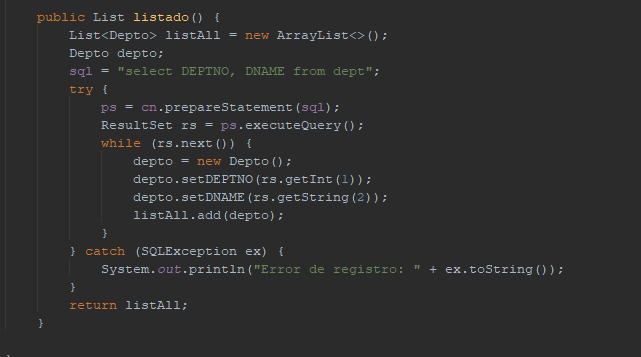
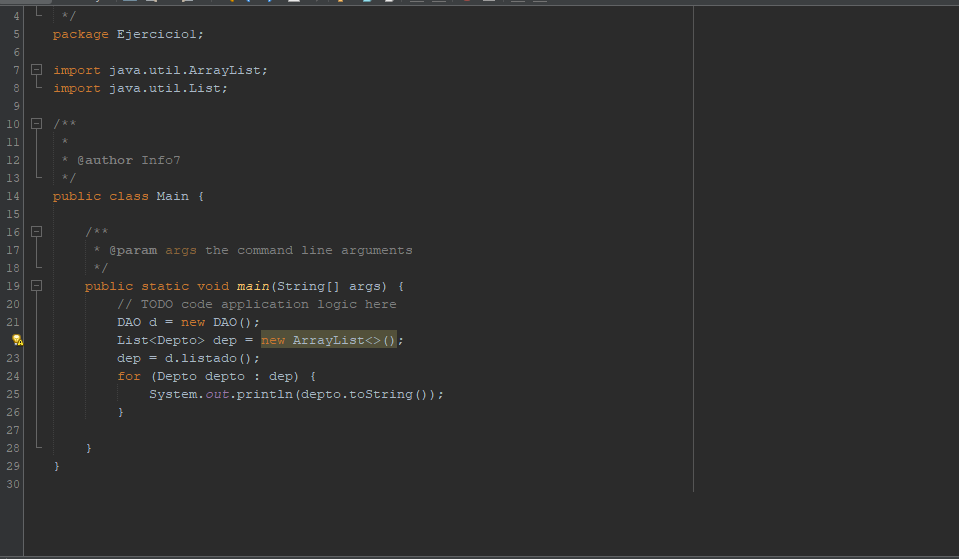
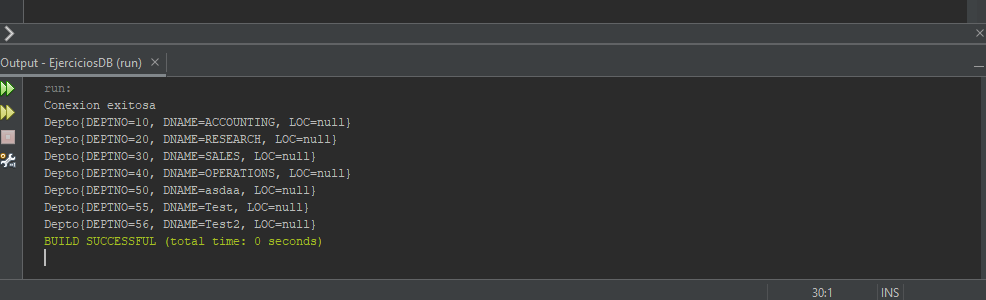
Ejemplo

**Se pide:**

Captura de pantalla del método de conexión y consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. Método de conexión a la base de datos: PIÉ DE PÁGINA  
   
2. Clase: PIÉ DE PÁGINA  
   Main:  
   
3. 

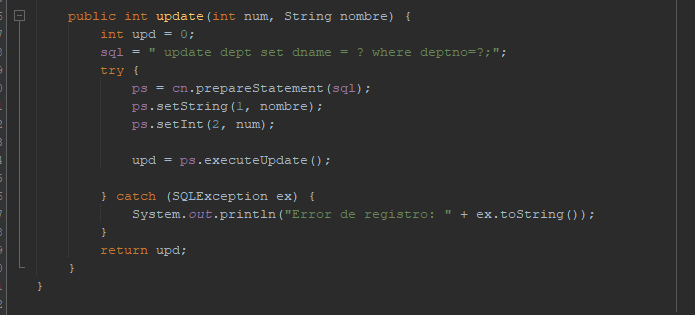
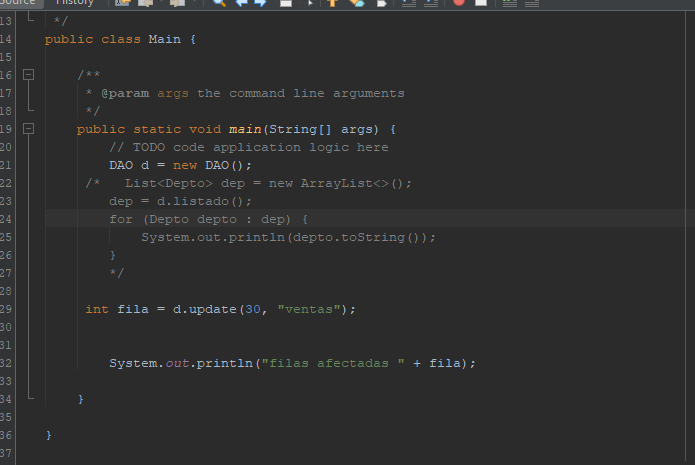
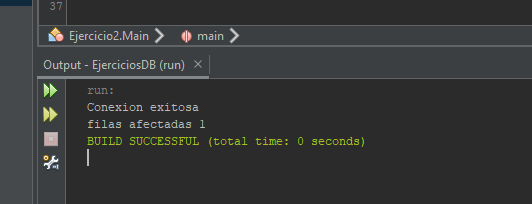
**Ejercicio N°2**Modificar el nombre de un departamento cuyo número y nombre se pasen como argumentos al método. Visualizar el número de filas afectadas.  
En caso de que consideres necesario, podes usar las clases creadas en el ejercicio anterior.

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:  
   
3. 

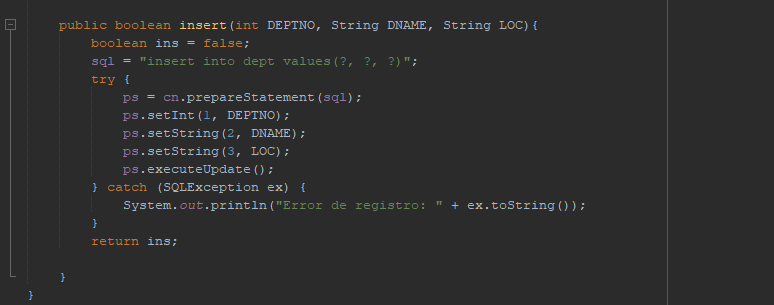
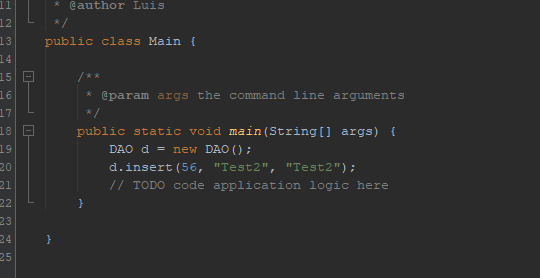
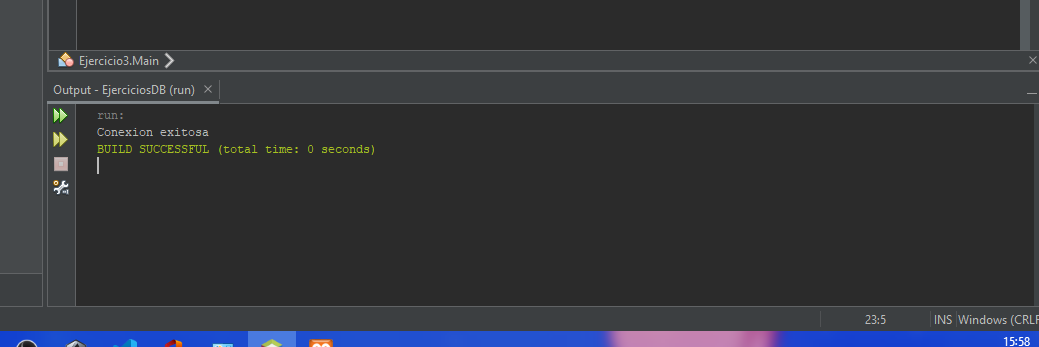
**Ejercicio N°3**Insertar un departamento. El método recibirá tres argumentos (número, nombre y localidad).

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 

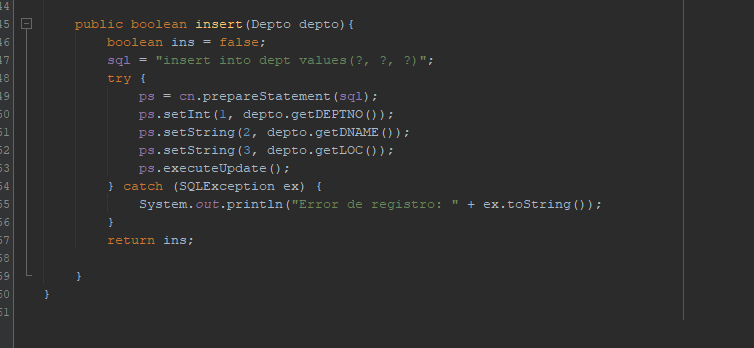
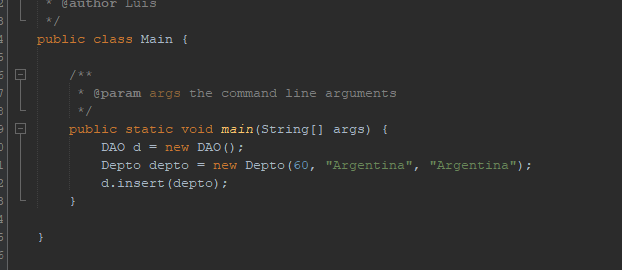
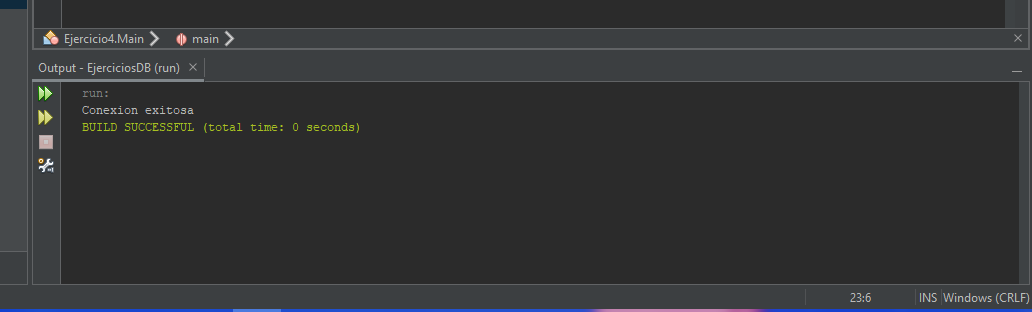
**Ejercicio N°4**El mismo que el ejercicio anterior, pero recibiendo un solo argumento, un objeto de la clase departamento. Será necesario por tanto crear una clase departamento, con sus atributos y métodos getter y setter (esta clase se creó en el ejercicio 1).

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:  
   
3. 

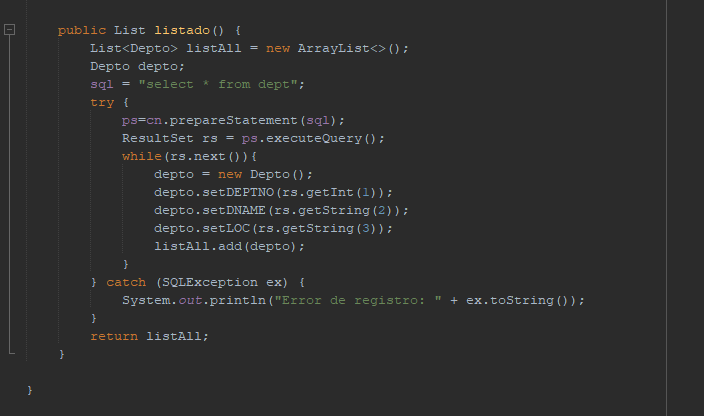
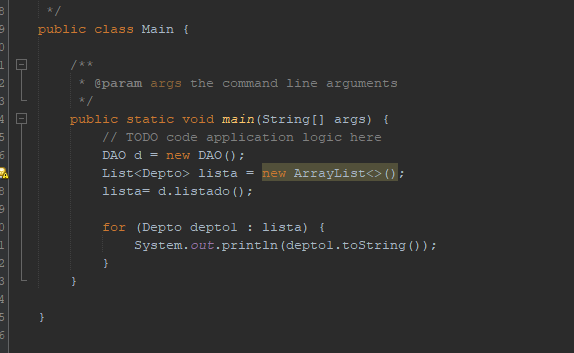
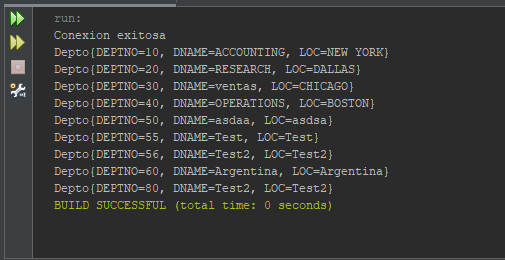
**Ejercicio N°5**  
Método que devuelva un ArrayList de objetos departamento ante la consulta de todas las columnas de todos los departamentos de la tabla DEPT;

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 

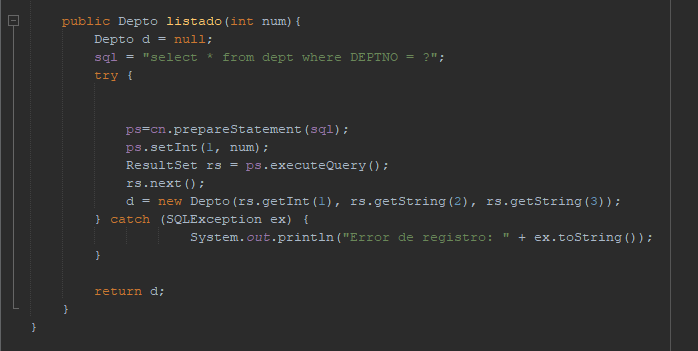
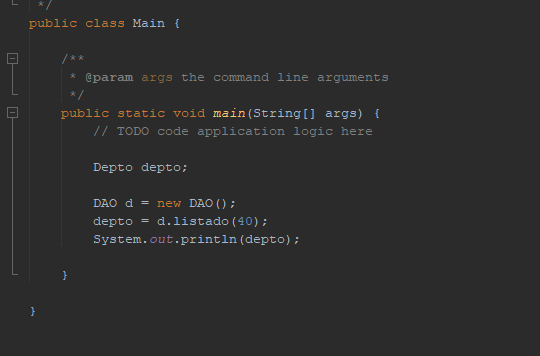
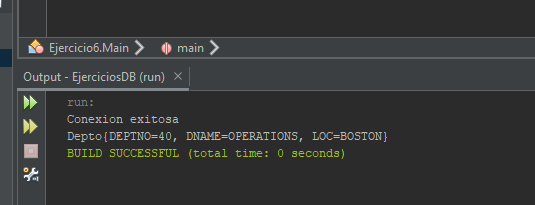
**Ejercicio N°6**  
Método que reciba un número de departamento y devuelva sus datos mediante un objeto departamento.

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 

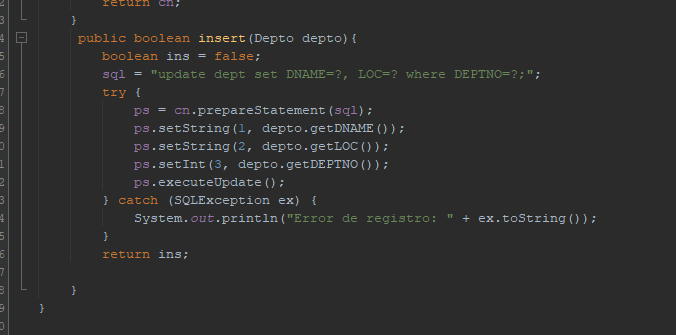
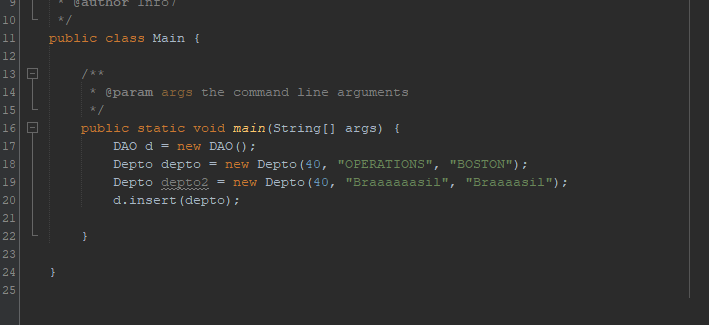
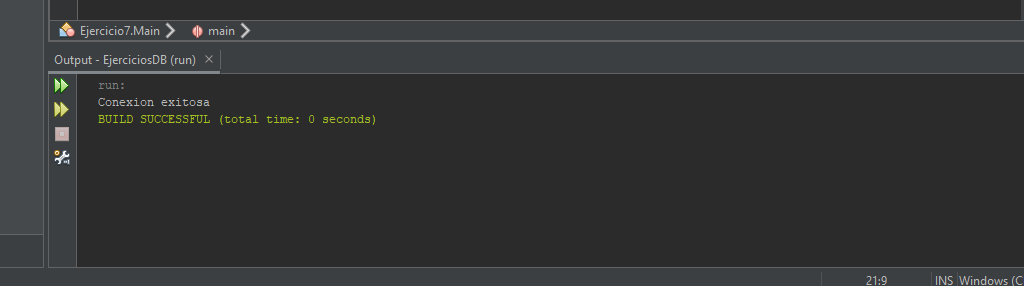
**Ejercicio N°7**  
Método que reciba un objeto departamento y actualice la tabla DEPT.

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 

**Ejercicio N°8**

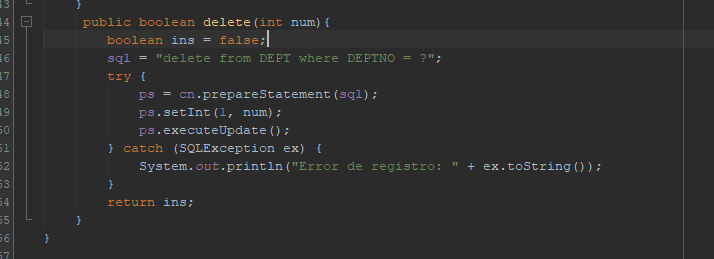
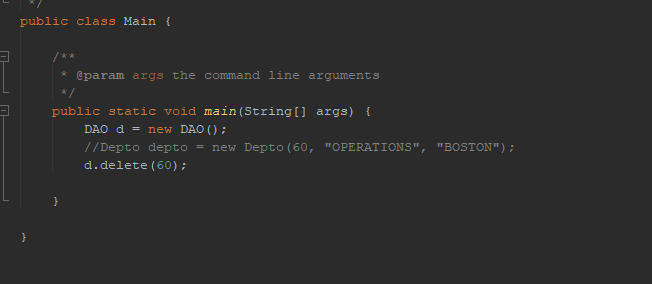
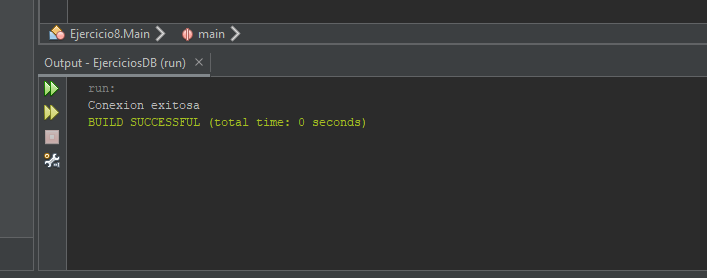
Método que reciba un número de departamento y lo dé de baja.

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:  
   
3. 

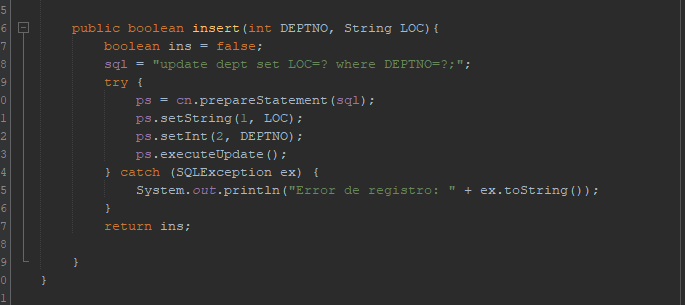
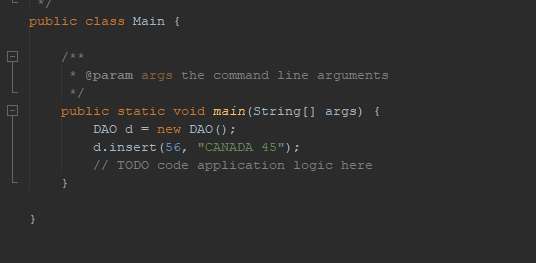
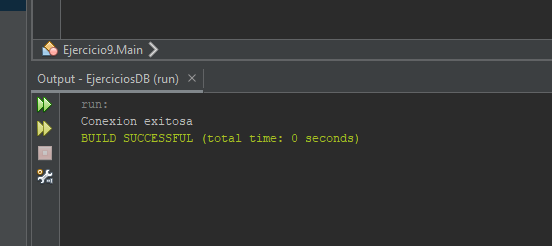
**Ejercicio N°9**Método que reciba un número de departamento y actualice su localidad, la misma se debe pasar como segundo argumento del método

**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 

**Ejercicio N°10**

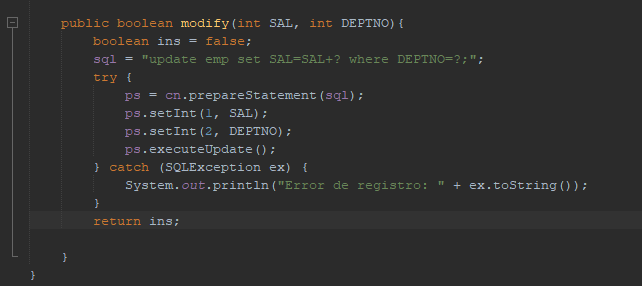
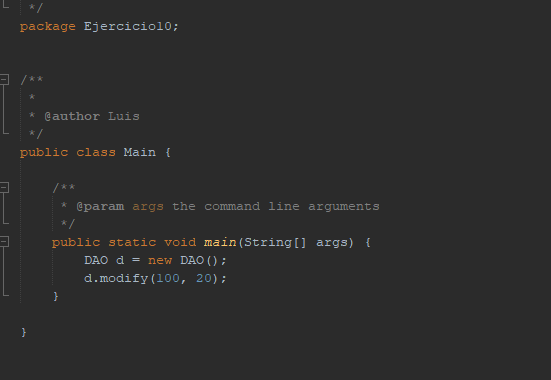
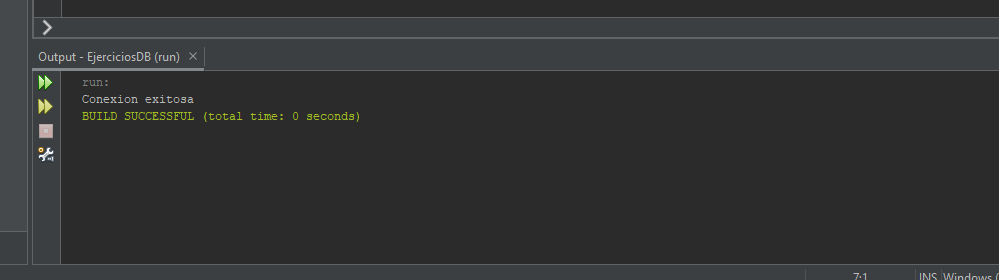
Método que reciba una cantidad y un número de departamento e ncremente el sueldo de todos los empleados de ese departamento en la cantidad pasada como parámetro.

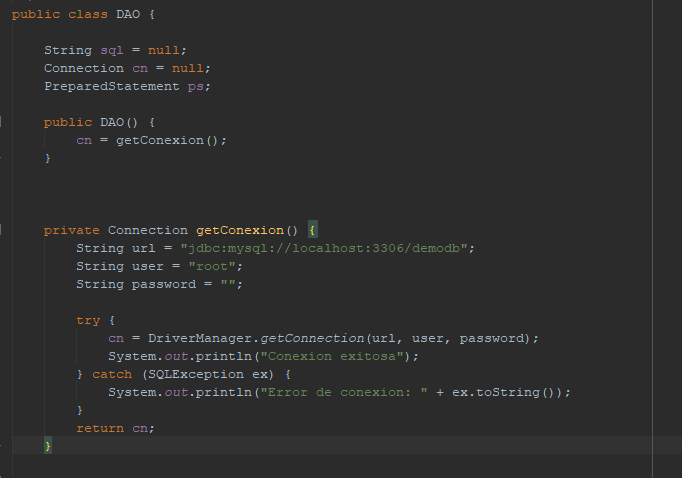
**Se pide:**

Captura de pantalla del método que realiza la consulta (pegar acá)

Captura de pantalla de la clase Test y el método main (pegar acá)

Captura de pantalla de la salida por consola (pegar acá)

1. 
2. Clase: PIÉ DE PÁGINA  
   Main:
3. 
4. Método de conexión a la base de datos:



1. Clase:  
   