AD

Álvaro Manuel Navarro Cruz

2º DAM

22/11/2024

Práctica 2

PERSISTENCIA EN BBDD RELACIONALES

Índice

[Introducción (↑) 3](#_Toc182851374)

[1.- Documentación (↑) 4](#_Toc182851375)

[1.1.- Creación Proyecto Maven 4](#_Toc182851376)

[1.2.- Creación de Paquetes y Clases 8](#_Toc182851377)

[2.- Creación de la BBDD (↑) 10](#_Toc182851378)

[3.- Capturas del código (↑) 11](#_Toc182851379)

[3.1.- Clase ClienteANC 11](#_Toc182851380)

[3.2.- Clase ClientePersistenciaANC 12](#_Toc182851381)

[3.3.- Clase TestHotelANC 14](#_Toc182851382)

[3.4.- Clase GestorConexiones 16](#_Toc182851383)

[4.- Ejecución (↑) 17](#_Toc182851384)

[4.1.- Creación de Clientes 18](#_Toc182851385)

[4.2.- Leer Cliente 19](#_Toc182851386)

[4.3.- Actualizar Cliente 20](#_Toc182851387)

[4.4.- Eliminar Cliente 22](#_Toc182851388)

[4.5.- Fin del Programa 23](#_Toc182851389)

[5.- Valoración Personal (↑) 24](#_Toc182851390)

[6.-Bibliografía 25](#_Toc182851391)

Introducción [(↑)](#indice)

En esta práctica, se pretende desarrollar una aplicación para gestionar un hotel que guarde la información en una base de datos.

Se pide realizar el código de persistencia necesario para almacenar los datos de los clientes del hotel. Para ello, necesitaremos la clase ClienteANC de tipo POJO, la clase ClientePersistenciaANC, la clase TestHotelANC y un gestor de conexiones.

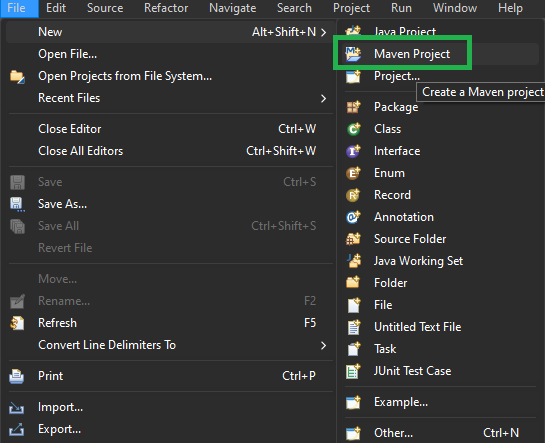
Se debe crear la Base de Datos correspondiente llamada HotelANC, con la tabla ClienteANC, que deberá llevar los campos idCliente, nombreCliente, apellidosCliente, emailCliente, dniCliente y claveCliente.

## Enlace a Repositorio en GitHub

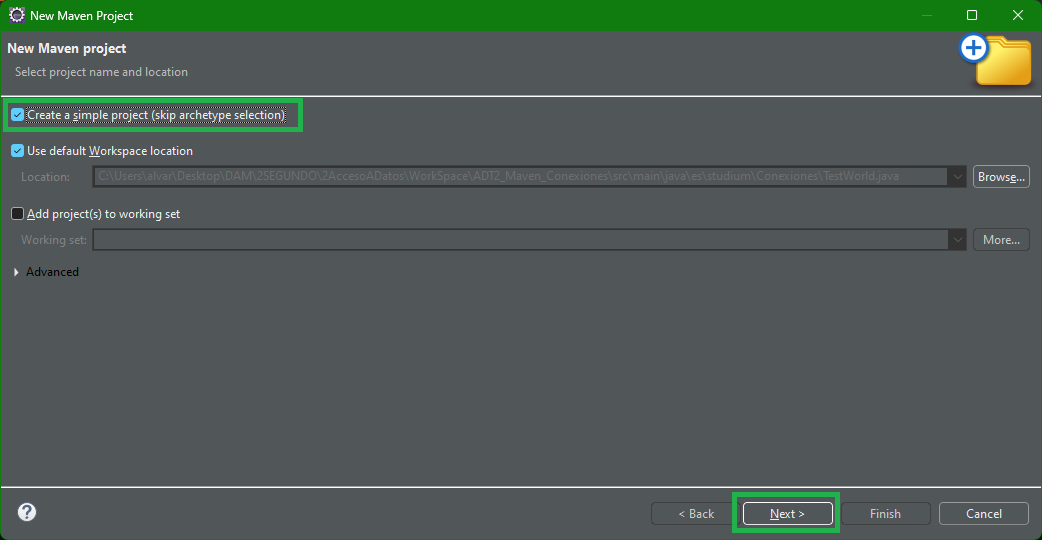
<https://github.com/AlvaroMfco/AD_Practica2.git>

1.- Documentación [(↑)](#indice)

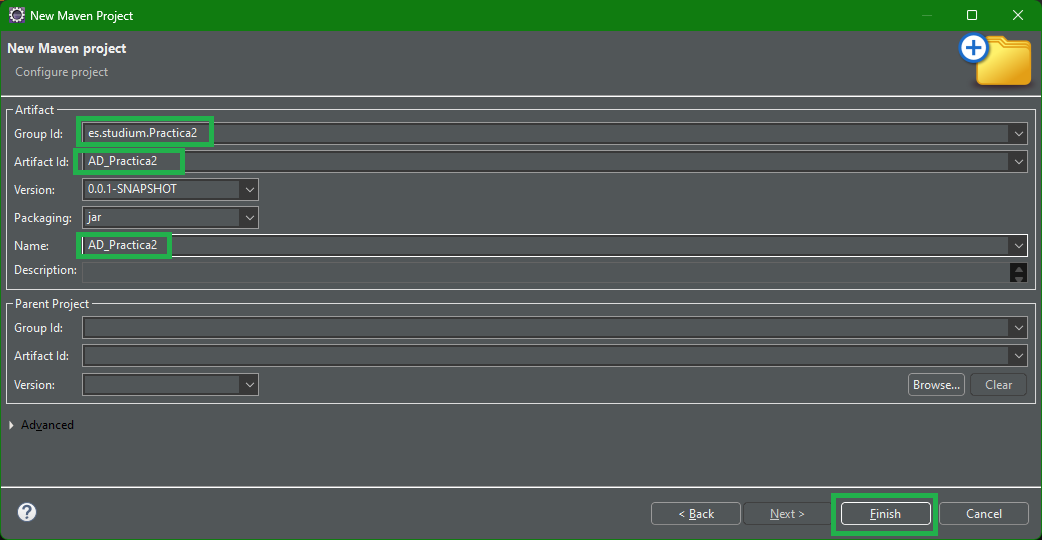
## 1.1.- Creación Proyecto Maven



Para crear un proyecto tipo Maven debemos clicar e “File -> New -> Maven Project”.

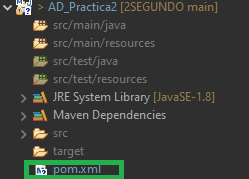


Se nos abrirá esta ventana, en la cual debemos seleccionar la opción marcada en la imagen superior.

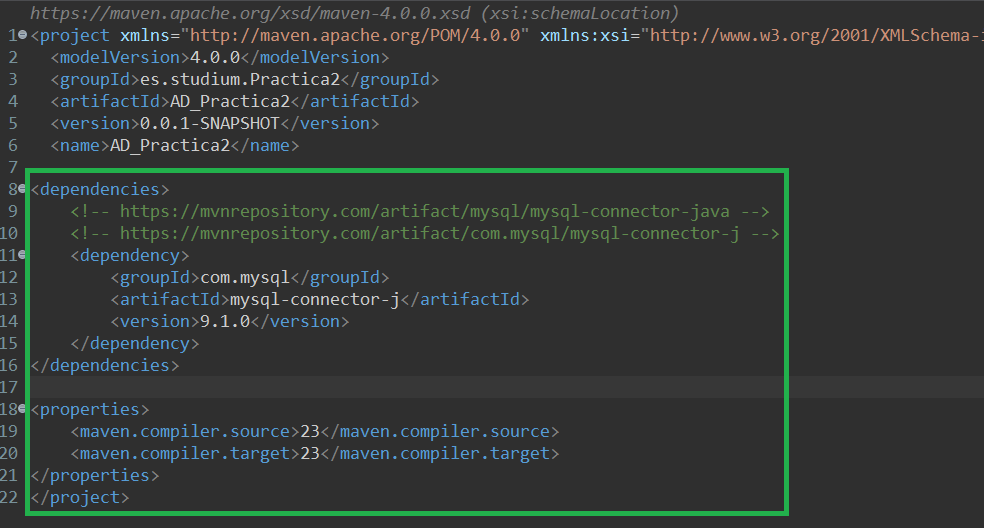


A continuación, se nos pedirán una serie de datos, a saber, “Group Id”, en el cual debemos establecer el nombre del paquete que vamos a utilizar, “Artifact Id”, donde debemos proporcionar el nombre del proyecto, que será el mismo que en “Name”.

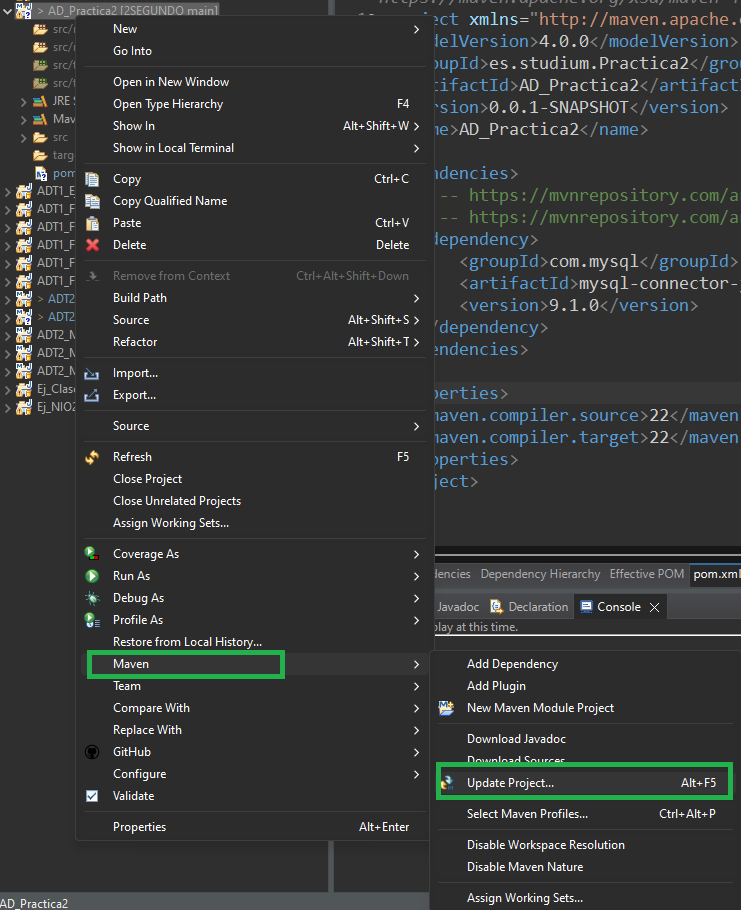
Para finalizar clicamos en “Finish”.



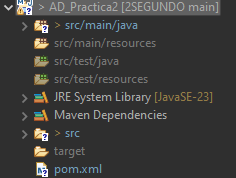
Una vez creado el proyecto, debemos acceder al fichero “pom.xml”, en el que debemos establecer las dependencias de MySQL connector para indicar su versión, además de las propiedades del compilador de Java, indicando la versión del JDK.



Para esta práctica, se utilizarán las últimas versiones disponibles tanto de MySQL como de JDK, quedando como en la imagen superior.

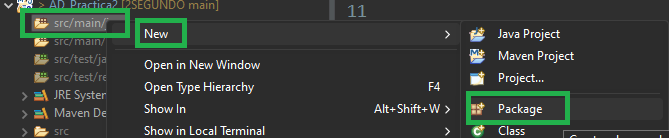


Una vez guardado el fichero pom.xml con los datos correctos, debemos actualizar el proyecto haciendo clic derecho sobre la carpeta del mismo y posteriormente clicar en “Maven -> Update Project…”.



Una vez actualizado el proyecto Maven, en el explorador de paquetes podemos observar que la versión de Java se ha actualizado correctamente, indicando la versión del JDK 23.

## 1.2.- Creación de Paquetes y Clases

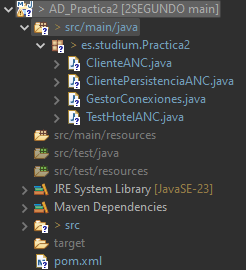


Para crear un nuevo paquete, debemos hacerlo en la carpeta “src/Main/java”. Para ello, debemos hacer clic derecho sobre la carpeta y clicar en “New -> Package”, al que daremos nombre de “es.studium.Practica2”.



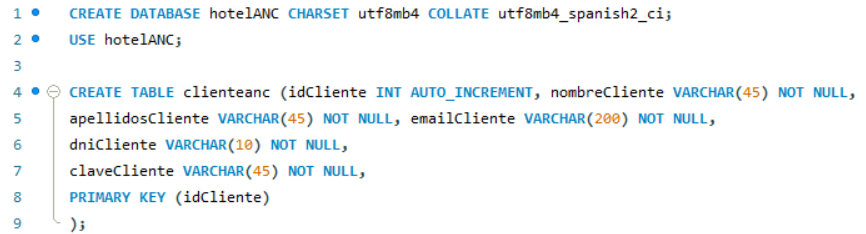
Para crear una clase, debemos hacer clic derecho sobre el paquete que acabamos de crear y navegar hasta “New -> Class”.

Una vez creadas todas las clases necesarias para la práctica, el explorador de paquetes se verá de la siguiente manera.



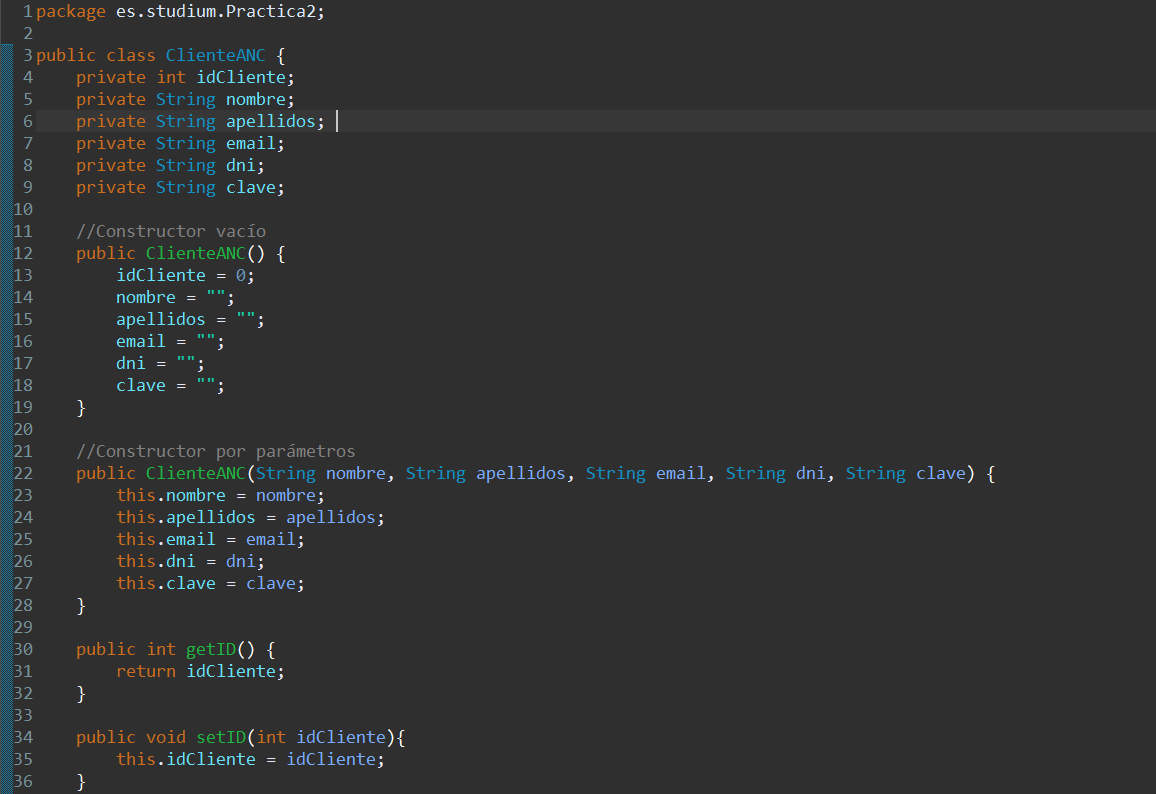
2.- Creación de la BBDD [(↑)](#indice)

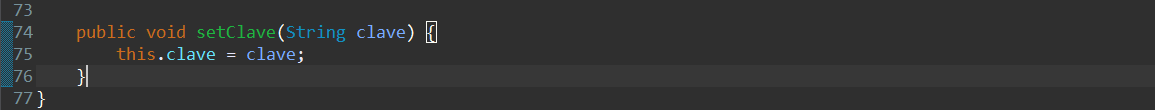
Nuestra Base de Datos, llamada **hotelanc**, tendrá una tabla llamada **clienteanc**, que a su vez estará compuesta por las columnas **idCliente, nombreCliente, apellidosCliente, emailCliente, dniCliente y claveCliente**.



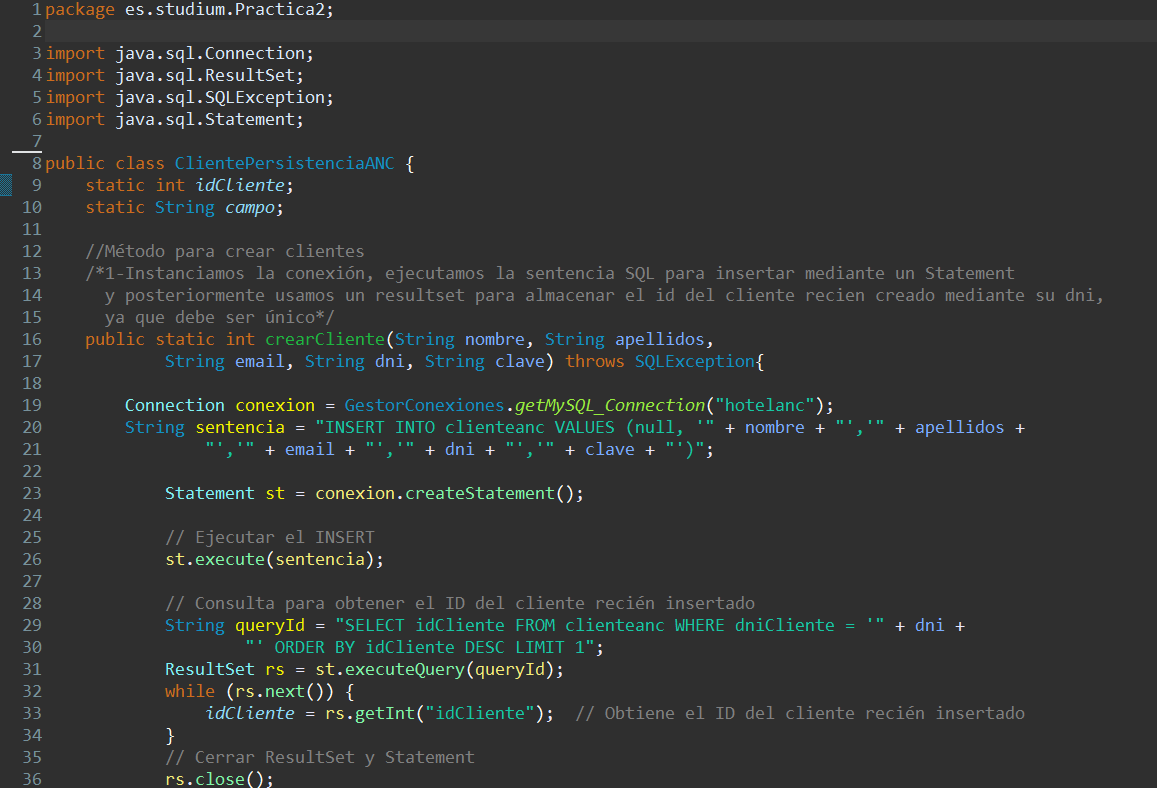
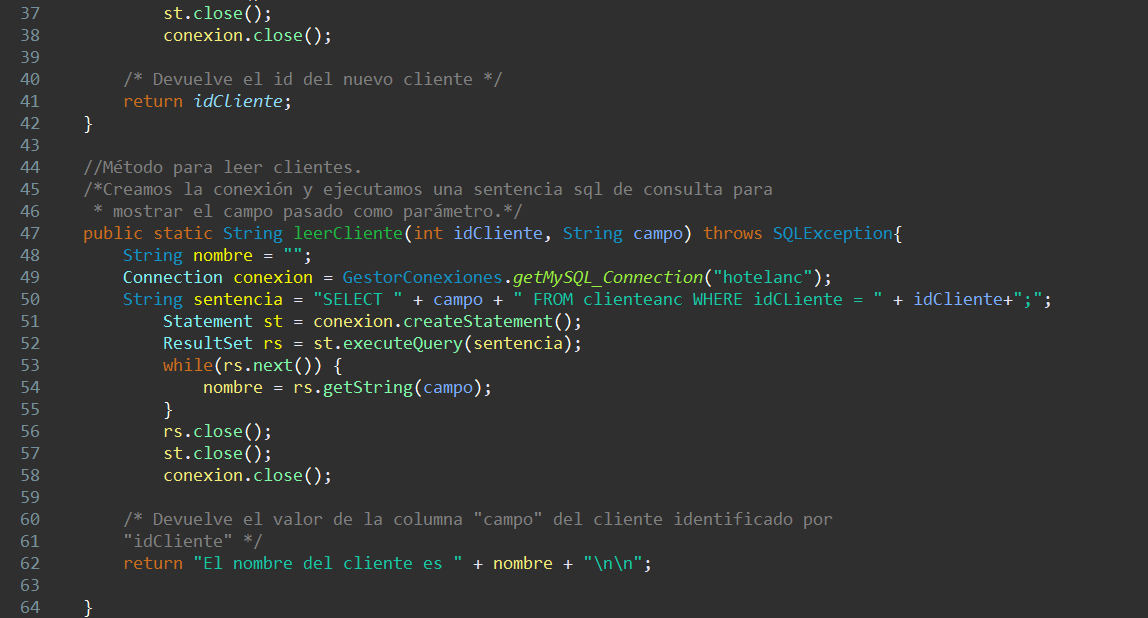
3.- Capturas del código [(↑)](#indice)

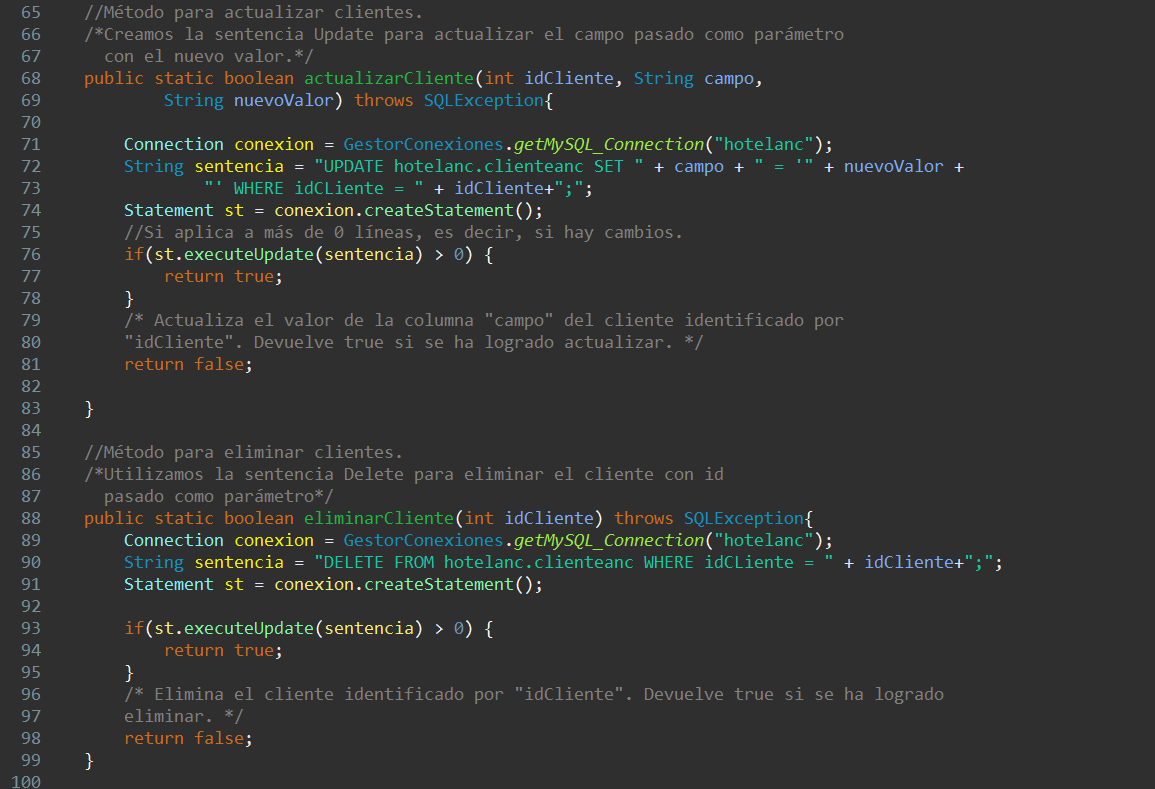
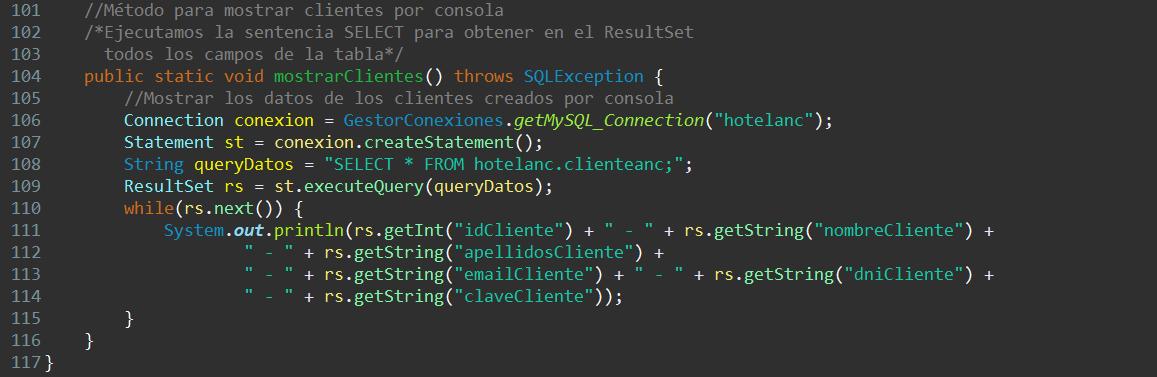
## 3.1.- Clase ClienteANC

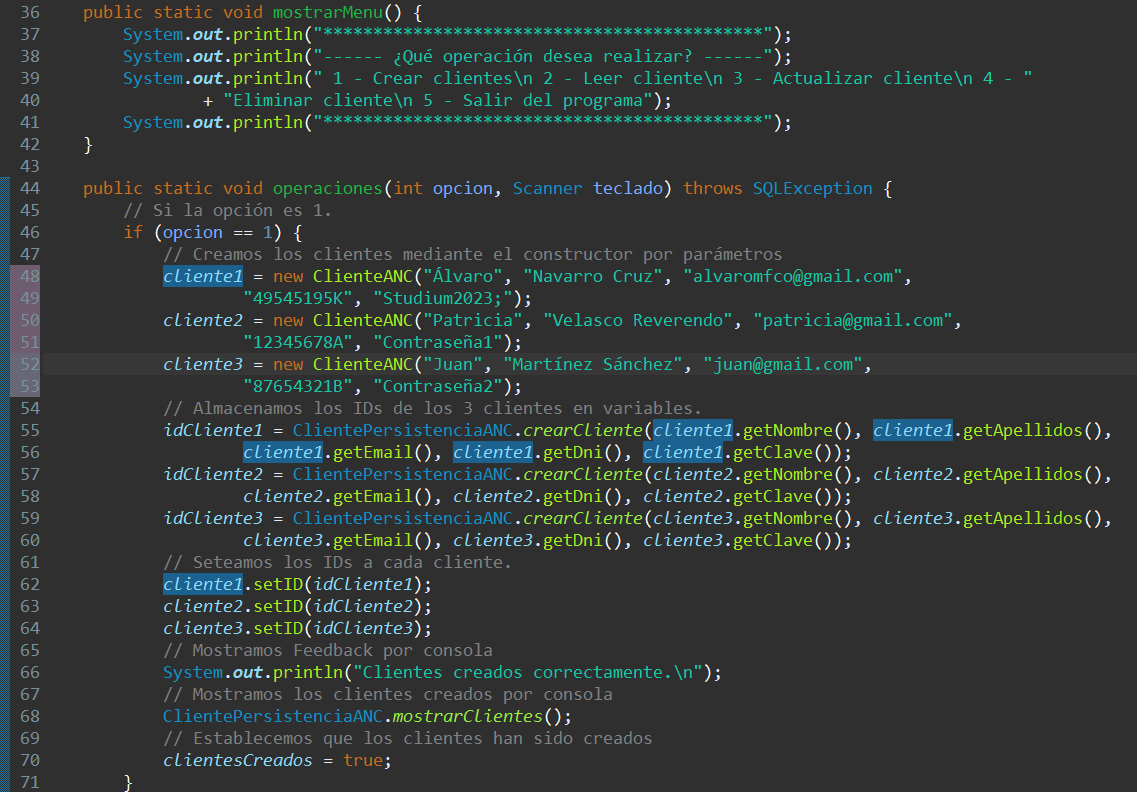


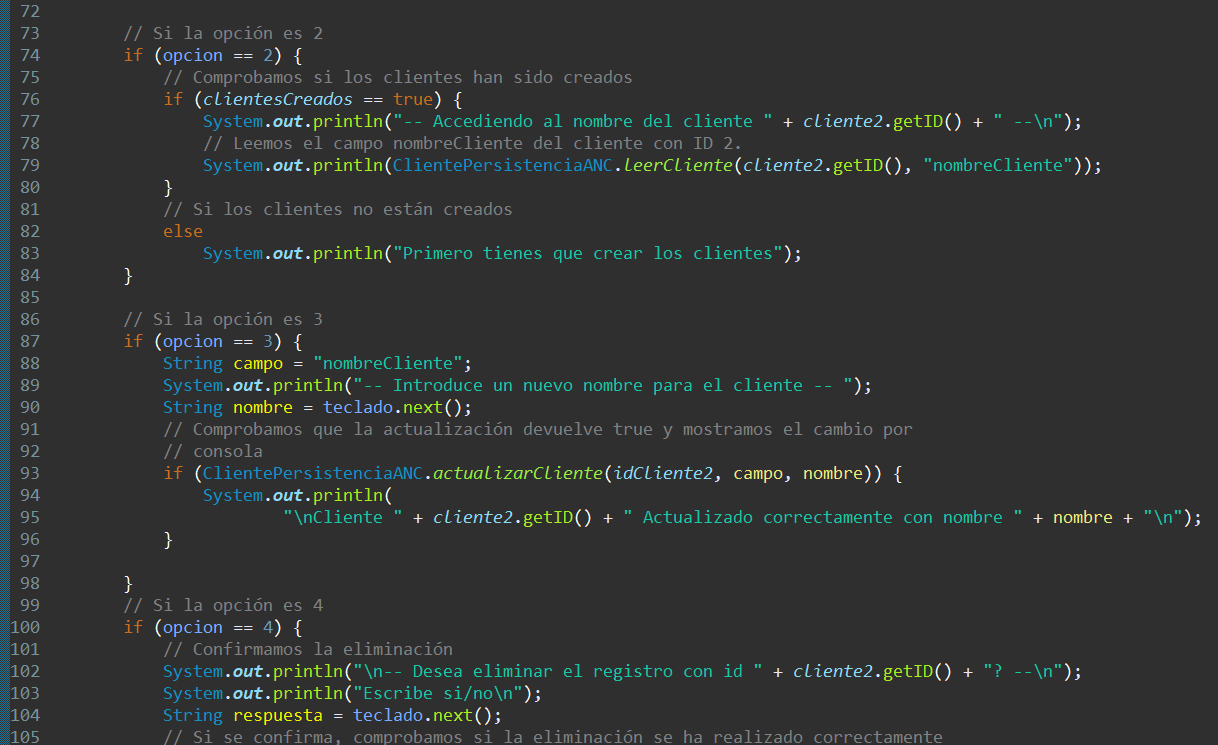
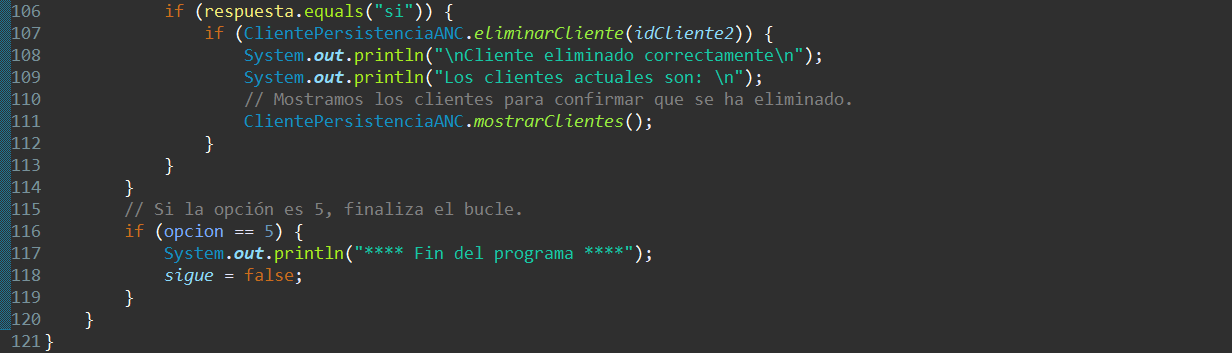
## 3.2.- Clase ClientePersistenciaANC



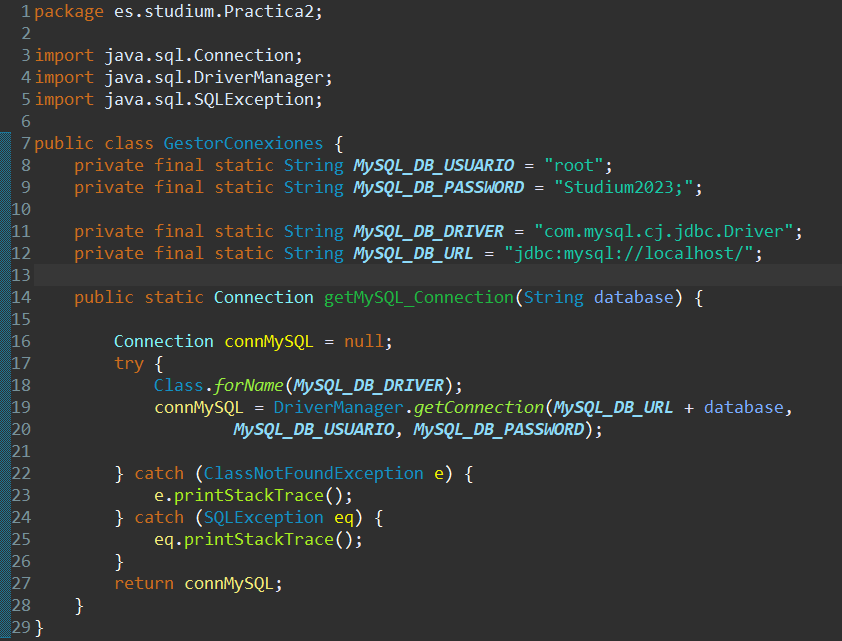


## 3.3.- Clase TestHotelANC



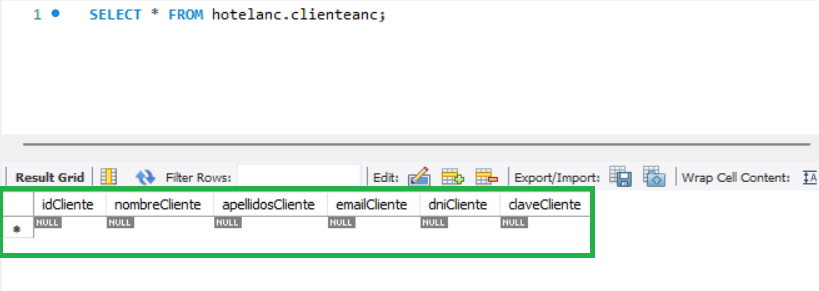


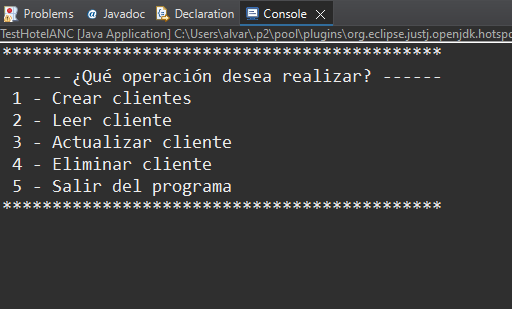
## 3.4.- Clase GestorConexiones



4.- Ejecución [(↑)](#indice)

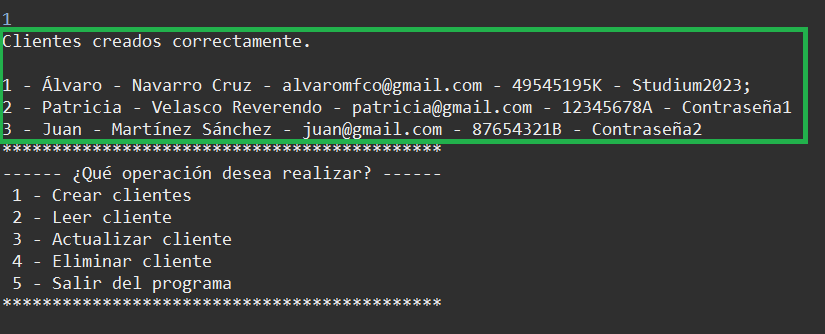
Para la ejecución de nuestro programa, en primer lugar, comprobamos que la tabla de la BBDD está vacía.





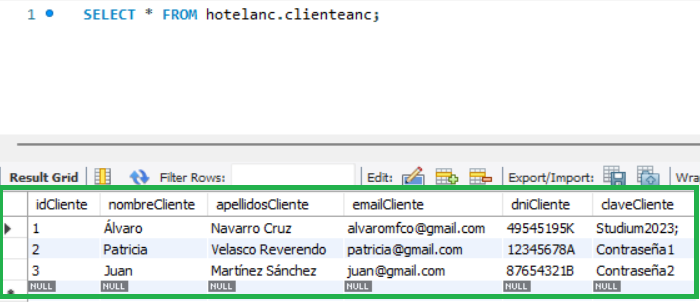
En la ejecución de nuestro programa, aparecerá un menú por consola para elegir la acción que queremos realizar. Debemos elegir la opción introduciendo el número correspondiente.

## 4.1.- Creación de Clientes



Si elegimos la opción 1 – Crear Clientes, aparecerá por consola un mensaje indicando si se han creado correctamente o si, por el contrario, ha ocurrido algún error.

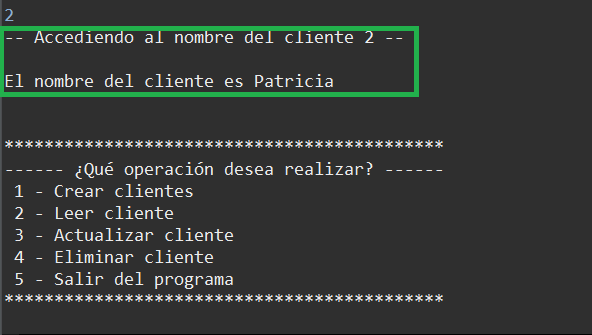
Si los clientes se crean correctamente, se mostrarán por consola.



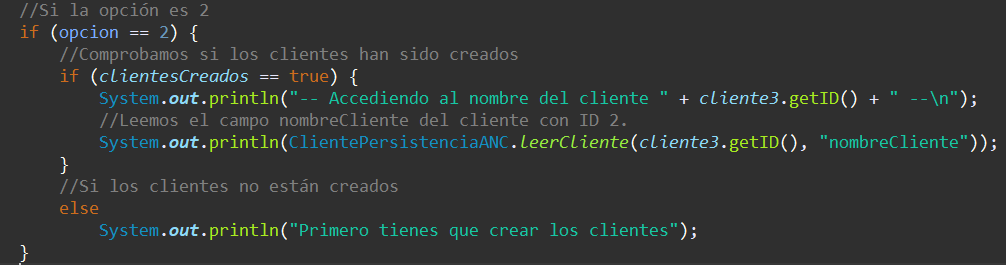
Como podemos ver, la tabla de la BBDD se ha actualizado con los datos correctos.

## 4.2.- Leer Cliente

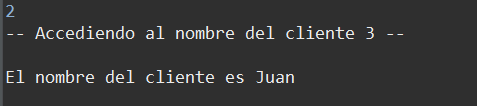
Para esta ejecución, vamos a leer el campo “nombreCliente”, del cliente con ID = 2.



Vamos a cambiar el código para acceder ahora al nombre del cliente 3:



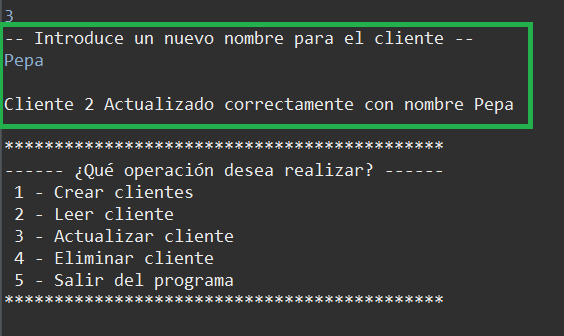
Al ejecutar el programa y elegir la opción 2, se mostrará el nombre del cliente con Id 3.



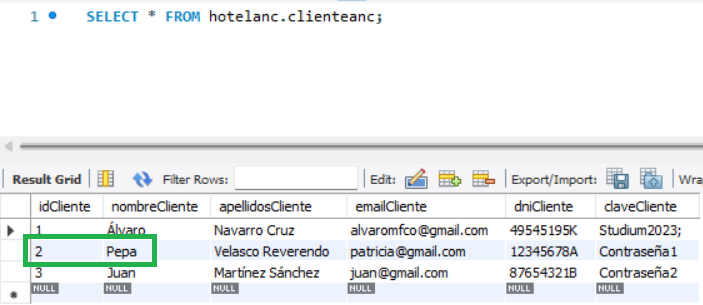
## 4.3.- Actualizar Cliente

Si elegimos la opción 3, se nos pedirá un nuevo nombre para editar el cliente con Id 2.

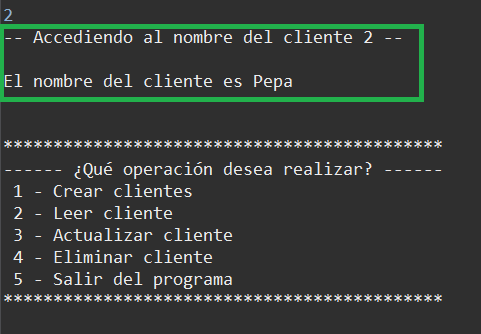
Al introducirlo, se mostrará un mensaje que confirma si se ha actualizado correctamente el nombre.



Si comprobamos la tabla de la BBDD, podemos observar que se ha actualizado el nombre correctamente.



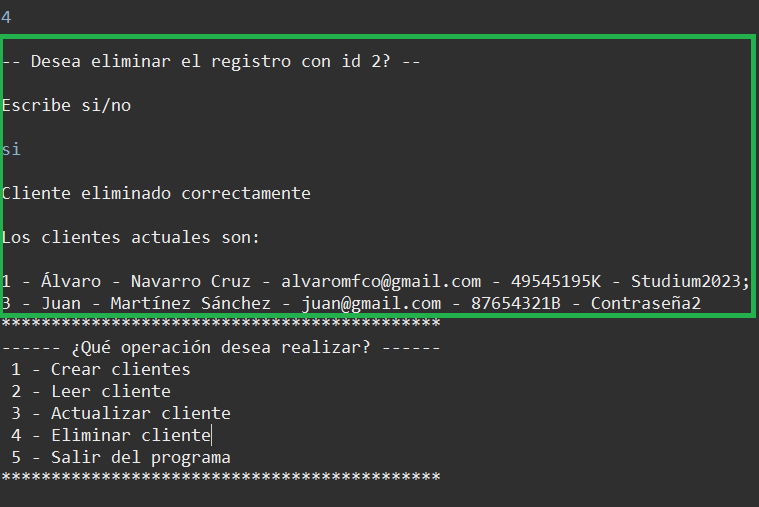
Si, por otra parte, volvemos a elegir la opción 2 – Leer Cliente, con el Id del cliente 2, aparecerá el nombre cambiado correctamente:



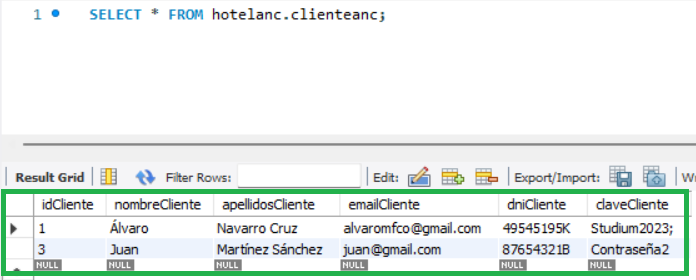
## 4.4.- Eliminar Cliente

Si elegimos la opción 4 – Eliminar Cliente, aparecerá un mensaje para confirmar si deseamos continuar con la operación.

Si escribimos “si”, el cliente se eliminará de nuestra Base de Datos y se mostrará por consola los clientes que quedan.

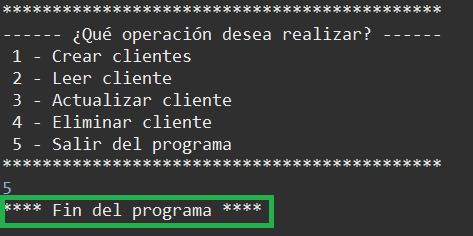


Si hacemos un SELECT en la Base de Datos, podremos ver que el cliente con Id 2 ha sido eliminado.



## 4.5.- Fin del Programa

Para finalizar el bucle de ejecución, debemos introducir el número 5 por consola, lo que hará que se muestre por pantalla que el programa se ha detenido.



5.- Valoración Personal [(↑)](#indice)

La realización de esta práctica me ha parecido muy interesante, a la vez que entretenida, teniendo que utilizar conexiones a una Base de Datos y la creación de métodos para las operaciones CRUD de una manera mucho más simple que en el primer curso.

Además, ha servido para afianzar los conocimientos vistos en las asignaturas tanto de Bases de Datos, como en Acceso a Datos.

Tengo mucho interés en esta asignatura y me parece que el método de conexión con Hibernate, que veremos en el próximo tema, será mucho más sencillo de implementar, facilitando y agilizando el proceso de creación de este tipo de programas.

6.-Bibliografía

Grupo Studium [(↑)](#indice)

1. *María José Martínez Navas, Acceso a Datos (Práctica Tema 2)*. Publicado en **Grupo Studium.**

Recuperado: <https://campustudium.com/pluginfile.php/15090/mod_resource/content/6/Tema%202%20-%20Persistencia%20en%20BD%20Relacionales%20-%20Pr%C3%A1ctica_v2.pdf>

Último acceso (18/11/2024).