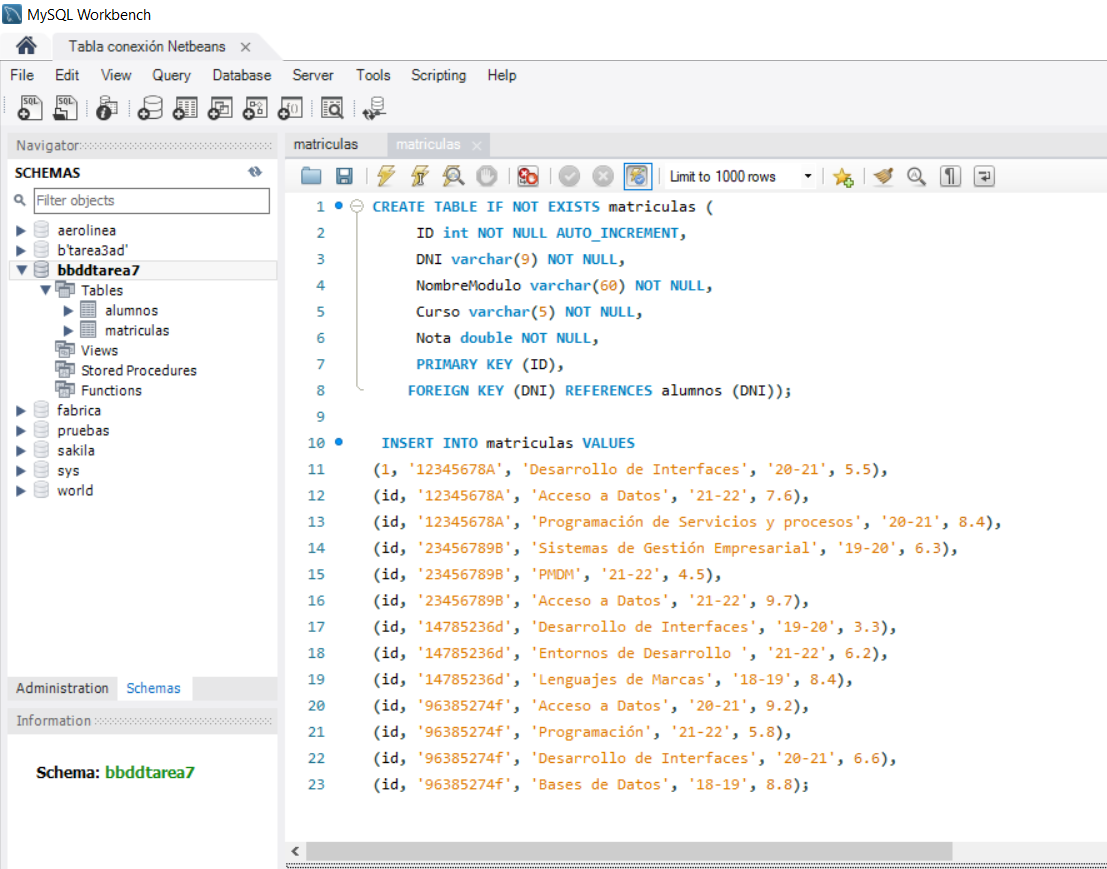
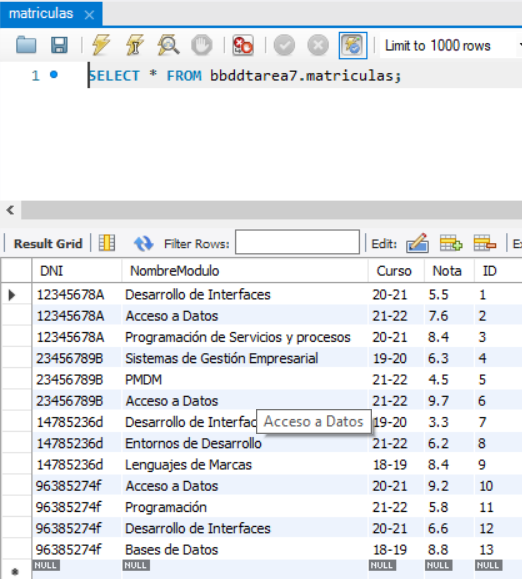
Partimos de la base de datos “alumnos” que viene en el temario y que voy a replicar en mi base de datos local. A esta le voy a añadir una tabla nueva llamada “matrículas” con los campos propuestos en la tarea y con algunos ejemplos. Para ello he creado el siguiente script (el script completo de la creación de la base de datos nueva lo he incluido en la tarea):



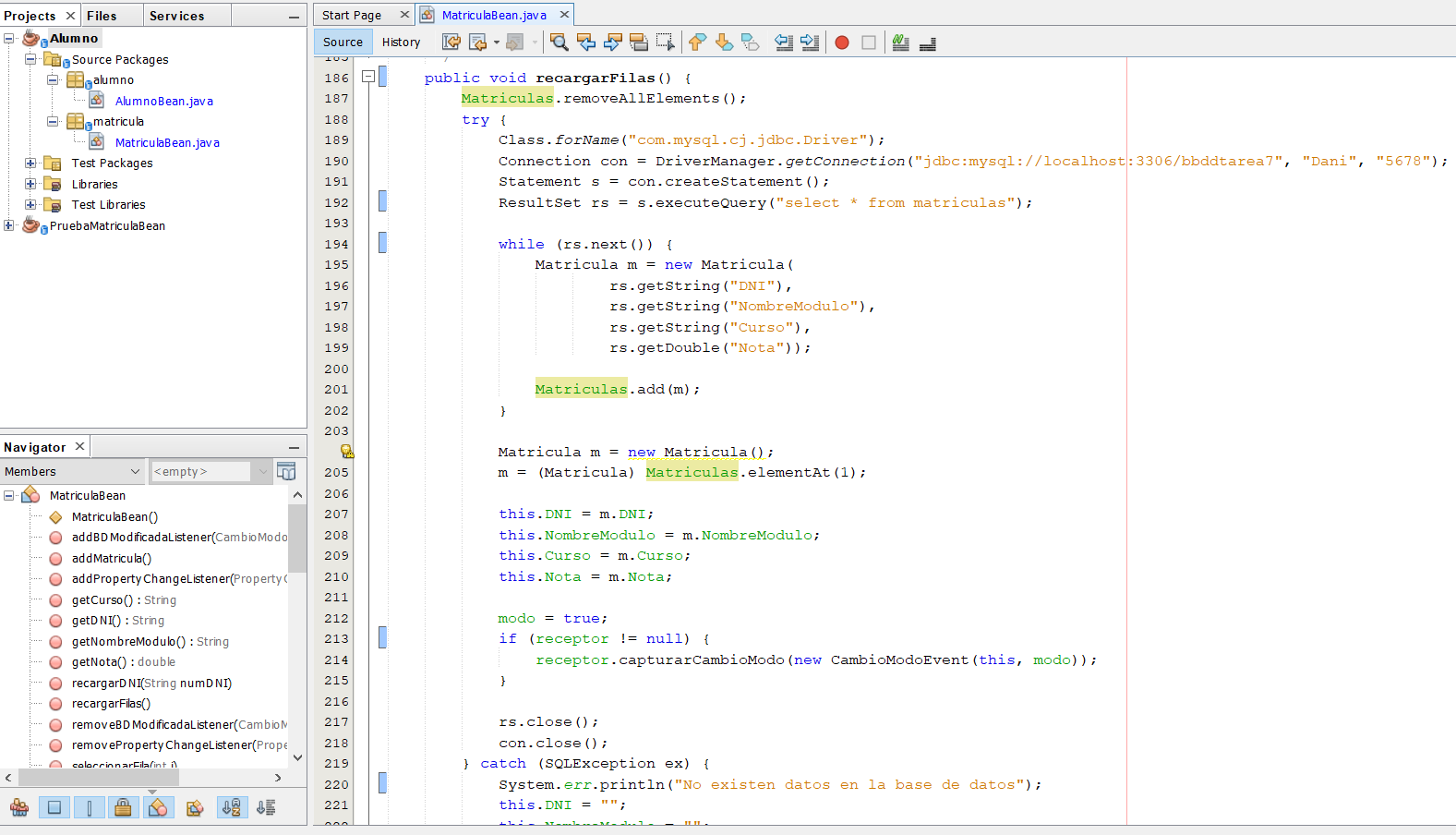
Al esquema le he llamado “bbddtarea7”, ya dentro cargo el script y lo ejecuto. En el navegador de la izquierda podemos ver que se han creado las 2 tablas



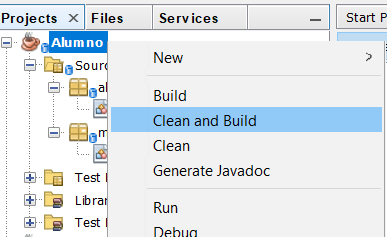
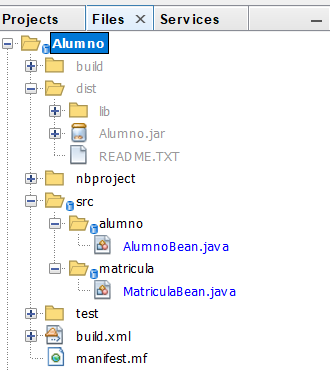
Comprobamos que en la tabla matrículas se han insertado los datos correctamente

A continuación, abrimos NetBeans y cargamos el proyecto “Alumno” descargado del temario. Este contiene el componente “AlumnoBean” que hace referencia a la tabla alumnos. Ahora vamos a crear un nuevo componente llamado “MatriculaBean” que nos haga las mismas funciones pero para la tabla matrículas.

Se nos pide que tenga los métodos **seleccionarFila(i), recargarDNI(numDNI), addMatricula()** y que la misma genere y gestione un evento dependiendo del modo en el que se encuentra (listado completo de matrículas o listado por dni). Para crear cada uno de estos métodos me he basado en los que vienen en el componente “AlumnoBean” reutilizando partes del código, adaptándolas y añadiendo sus particularidades. La nueva clase generada quedará de la siguiente manera:

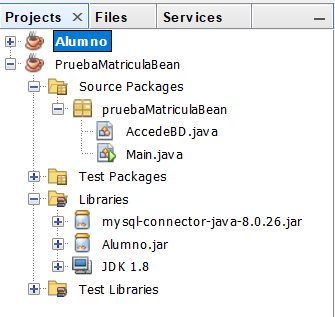


Una vez tenemos ya creado el componente, comprobado errores, contempladas excepciones y documentado, compilamos y construimos el proyecto para que se nos genere el .jar en la carpeta “dist”:

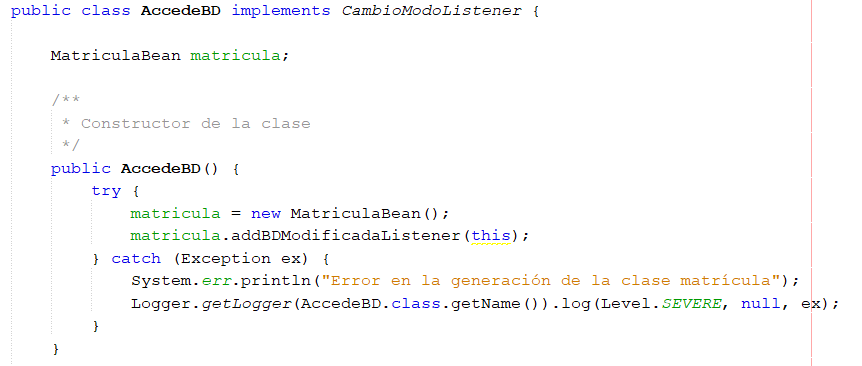


Vemos que se nos ha generado el archivo Alumno.jar dentro de la carpeta dist

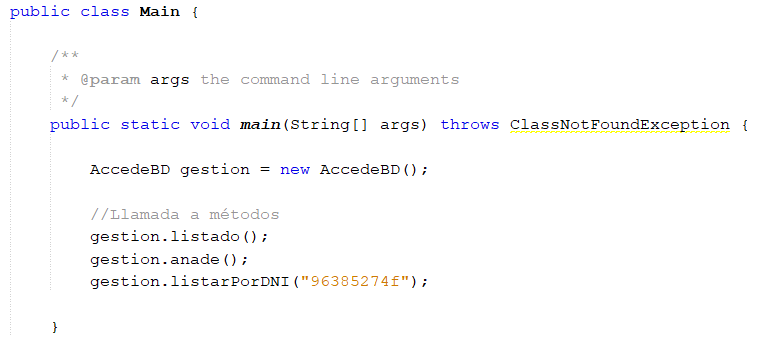
Ya tenemos el componente creado y listo para poder ser utilizado, así que nos disponemos a probarlo. Para ello voy a crear un nuevo proyecto en NetBeans al que voy a añadir los archivos .jar tanto del componente como del conector a la base de datos de MySQL:



Creamos una clase dentro del proyecto a la que llamaré “AccedeBD” en la que voy a incluir todos los métodos requeridos en la tarea, pero sin hacerlos de nuevo. Lo que haremos será hacer uso de los que hemos creado en el componente y como lo hemos añadido al proyecto instanciaremos un objeto de la clase matrícula para poder hacer uso de los atributos y métodos creados:

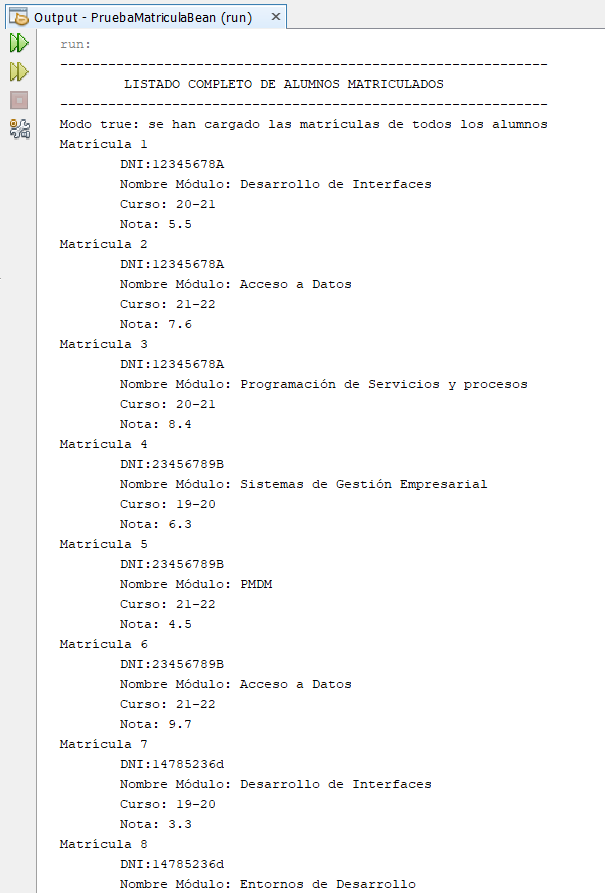


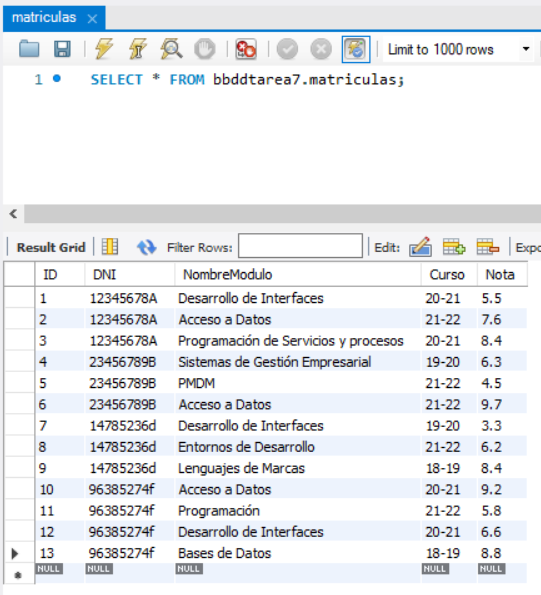
Crearé una clase “main” que será el comienzo de la aplicación y desde la cual vamos a llamar a los 3 métodos creados en la clase “AccedeBD”:



En este punto el proyecto está finalizado solo falta ejecutarlo y comprobar que realiza lo que se nos pide en la tarea:

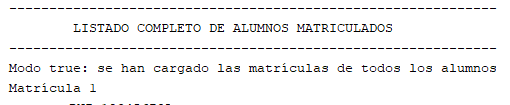
***Listado de las matrículas de todos los alumnos***



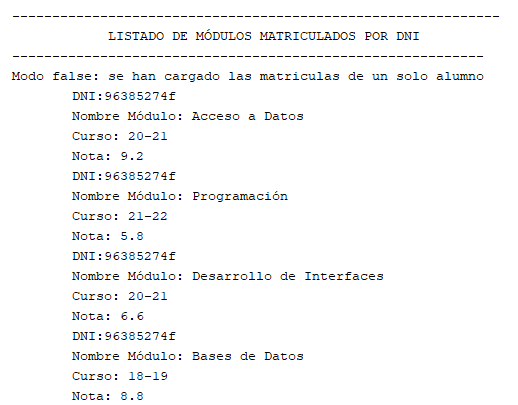
*Vista de la tabla matrículas en MySQL*

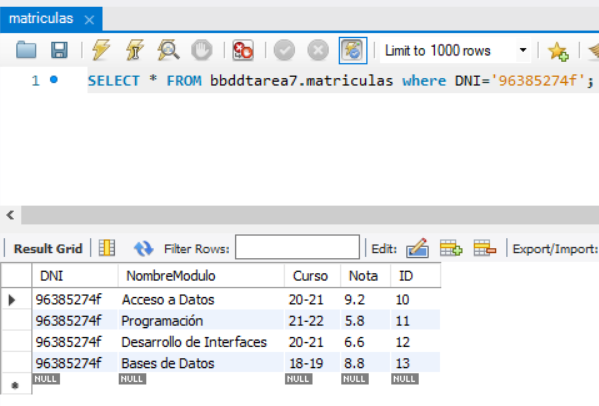
*Parte de la salida por pantalla de la aplicación, podemos ver que se muestran correctamente todos los datos de matrículas*

A la vez que se ejecuta el método se ha generado un nuevo evento de “modo” como así indica la aplicación por pantalla:



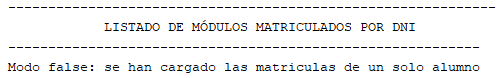
***Listado de matrículas de un alumno por DNI***

 La aplicación nos muestra todas las matrículas de un alumno concreto dado su dni:



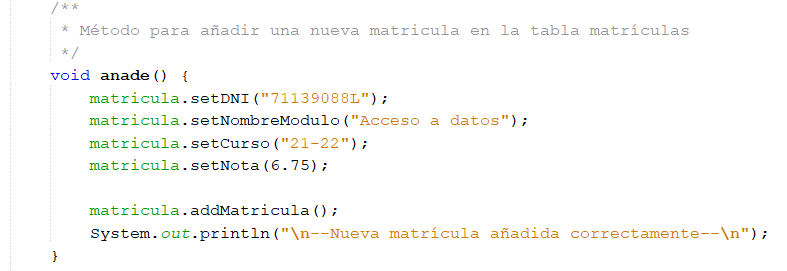
*Comparamos el resultado de la aplicación con el que nos muestra MySQL y vemos que el resultado es el esperado para el alumno con dni "96385274f"*

Además se ha generado un nuevo evento de “modo” ya que solo se muestra un alumno y no todos como en el caso anterior. Como antes se nos muestra por pantalla un mensaje que advierte el cambio de modo:



***Añadir una nueva matrícula a la base de datos***

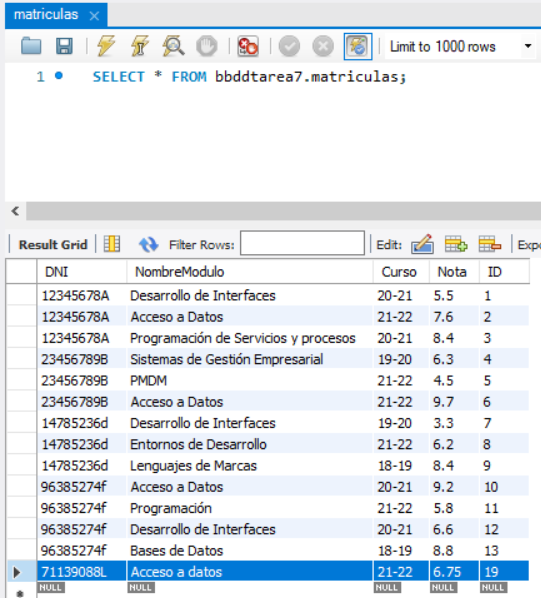
En el código, dentro del método, he rellenado datos a modo de ejemplo para ser añadidos a la tabla:



Incluyendo un mensaje que se mostrará por pantalla para informar al usuario que se ha añadido la nueva matrícula:



Nos vamos a MySQL y seleccionamos la vista de todos los datos de la tabla matrículas:



*Comprobamos en la última línea que se han añadido los datos de la nueva matrícula*