Endterm Project

Aziz Sakhiyev

Astana IT University

It – 2005

Object-Oriented Programming (Java)

Elvira Aitmukhanbetova

Introduction.

Project theme: game.

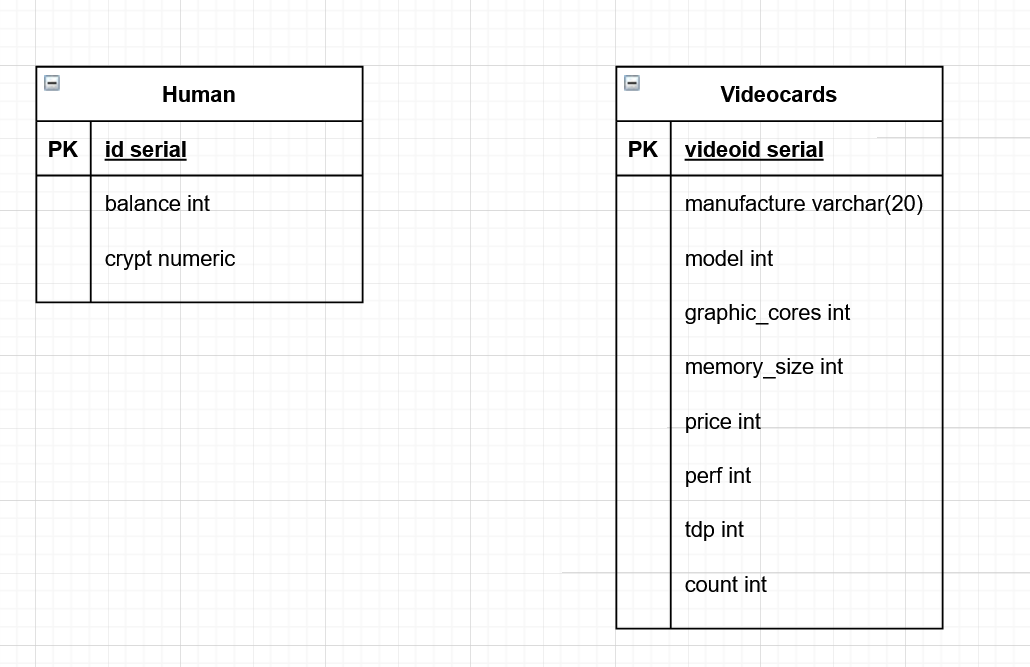
Development language: Java.

Frameworks used: Swing.

Game theme: mining simulation.

Database ER:

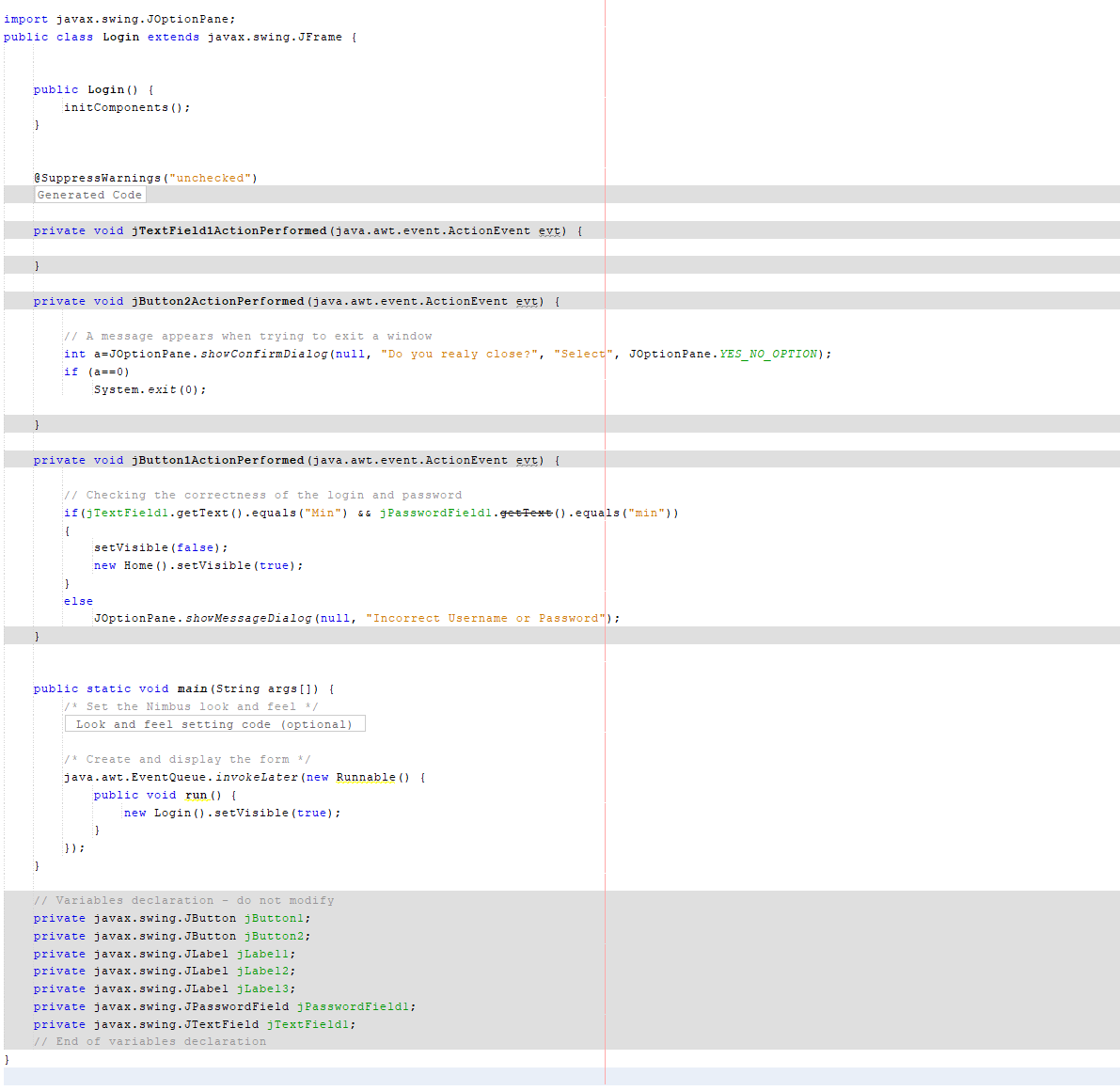
GitHub link: https://github.com/RDRrod/Test-3

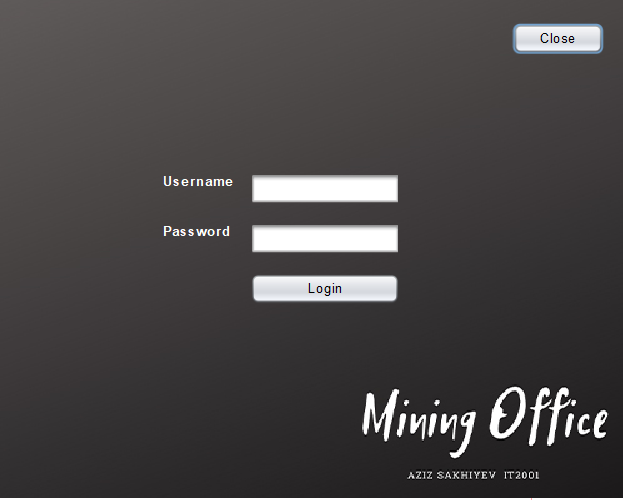
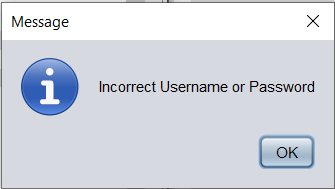


Main part.

1.Login

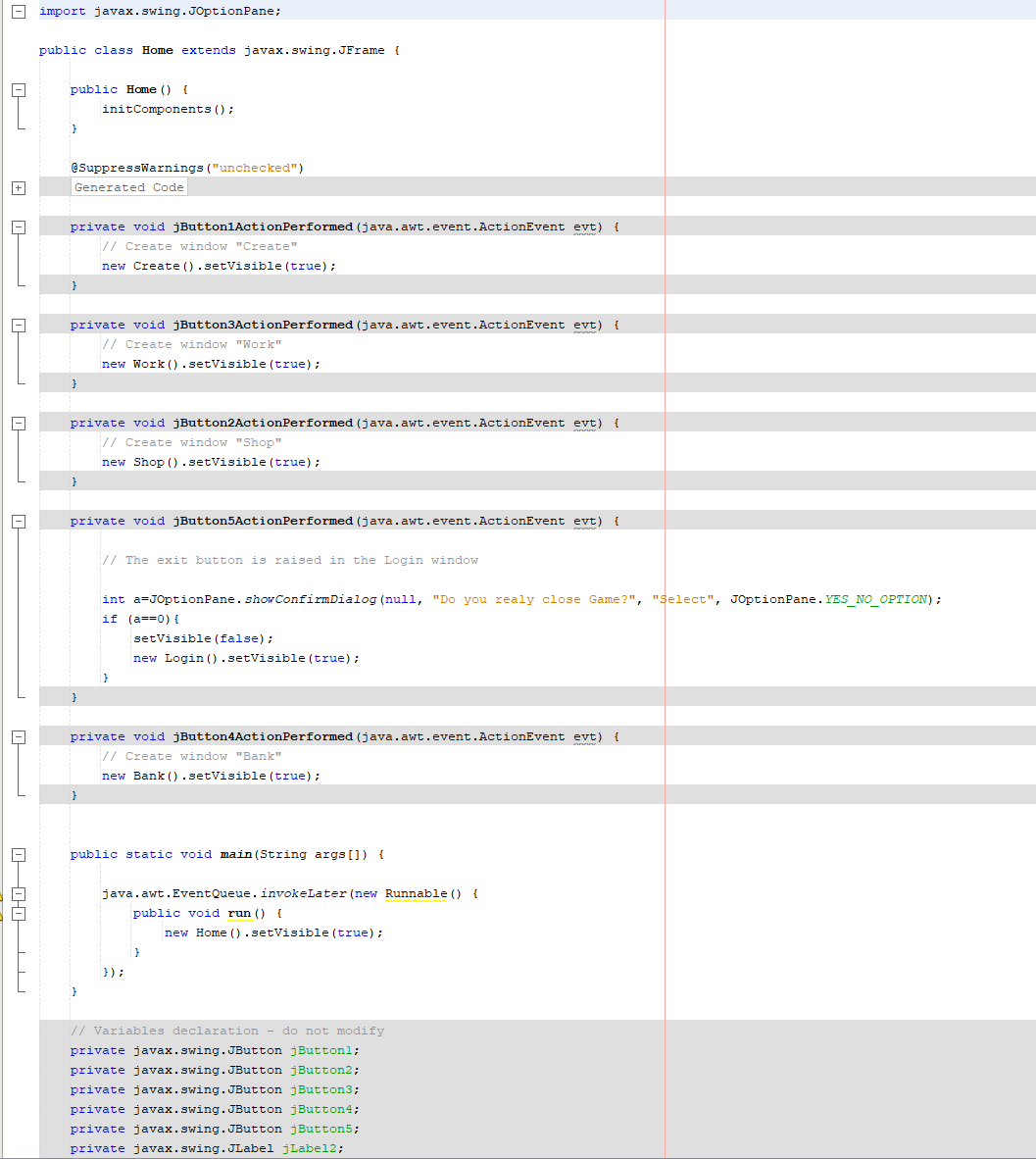
The main method of the Login class creates a window. In which are added buttons to exit the application (the window "do you really want to exit" appears) and authorization buttons. There are two text fields: Login and Password. The authorization button checks the correctness of the entered values, "Min" "min" - the right options. In case of unsuccessful authorization, a window will appear stating that "Invalid data have been entered". In the positive case, this window will close and a new Main class will be created.

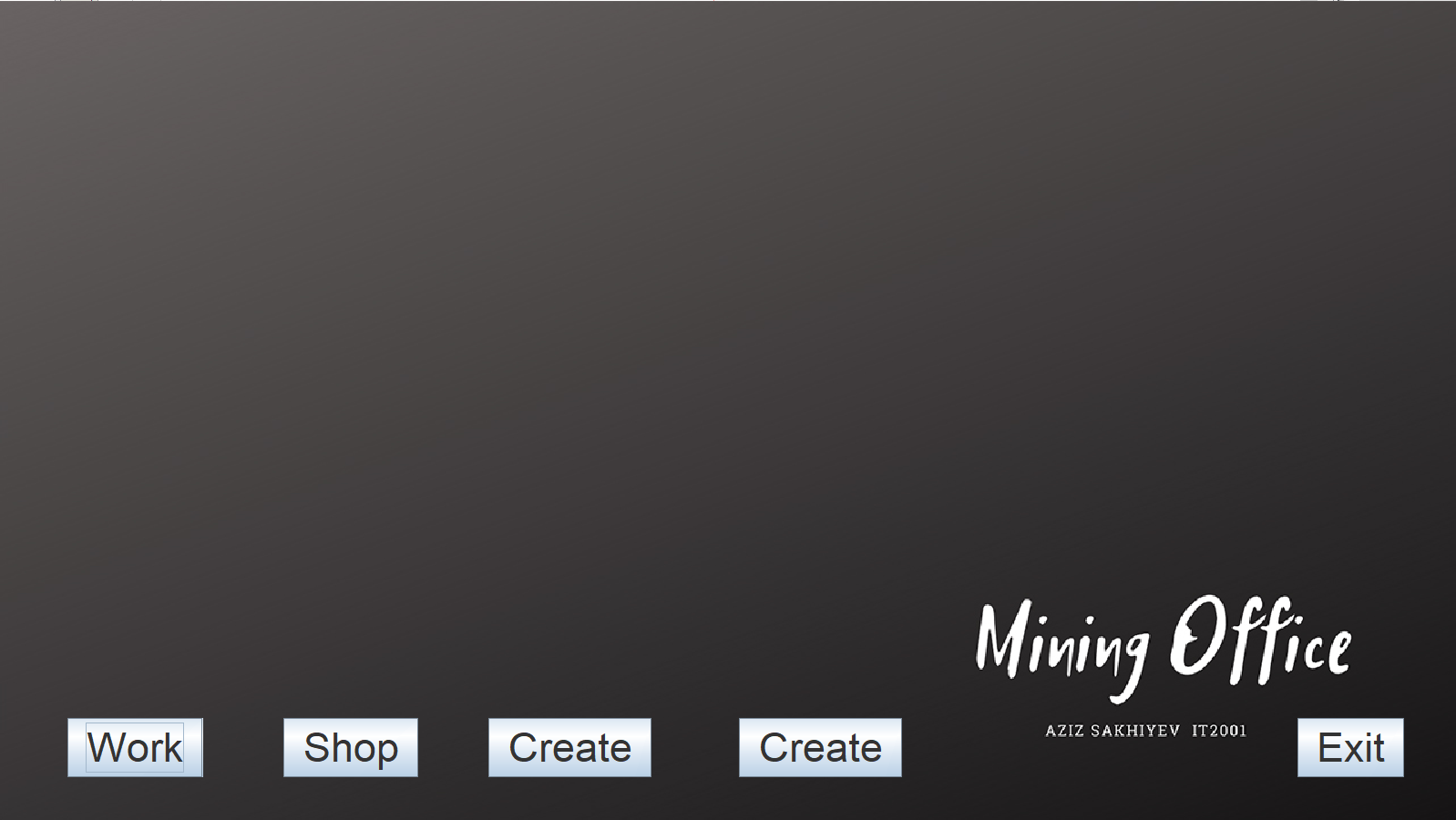


2.Home

This is the central class in which the Home window is created. The window has buttons that call new windows and redirect the user, and the old window is closed. There is also a close button that returns the Login window.





3. Connect

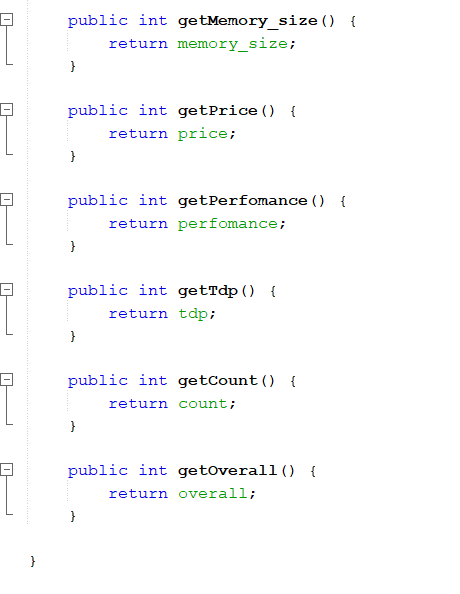
Used to connect to an external Postgresql database. It has a getConnection method which makes it easy to access the database in other classes.



4. Videocard

The video card class contains all the fields from the database, constructors, setters and getters required to transfer data from an sql query to a jTable.



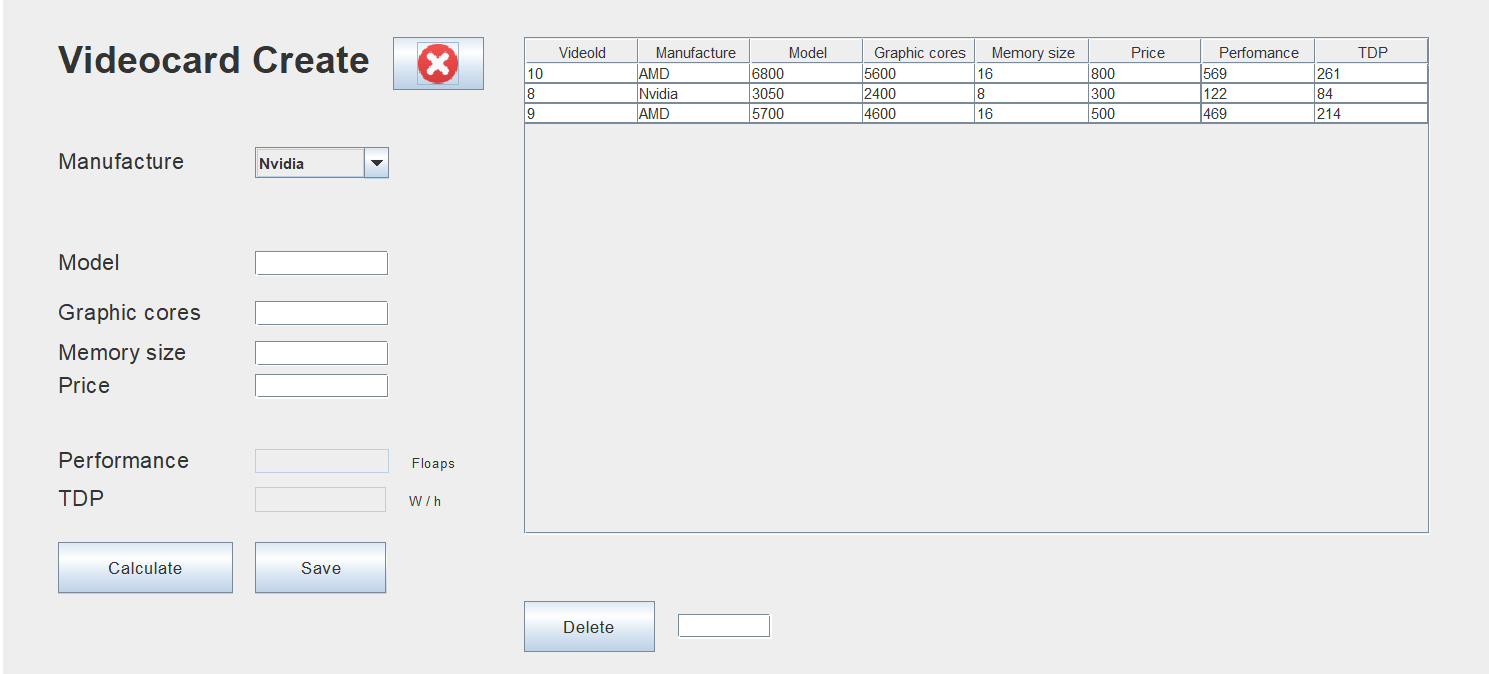


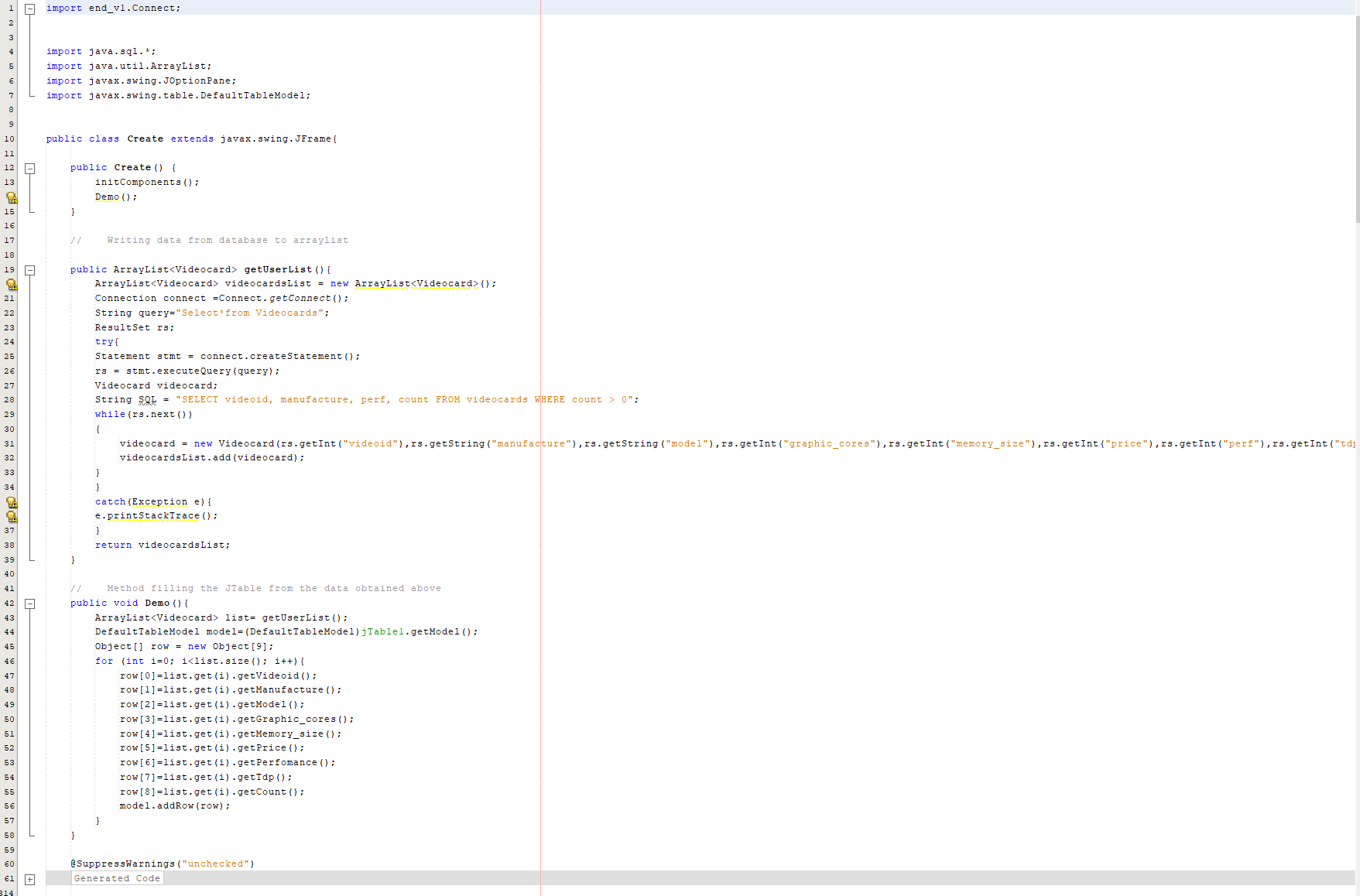
5. Create

The create class contains a table based on an sql query that collects information into an ArrayList <videocard> (receiving a set of elements from this query), then a method is described that sorts the resulting sheet into a table.

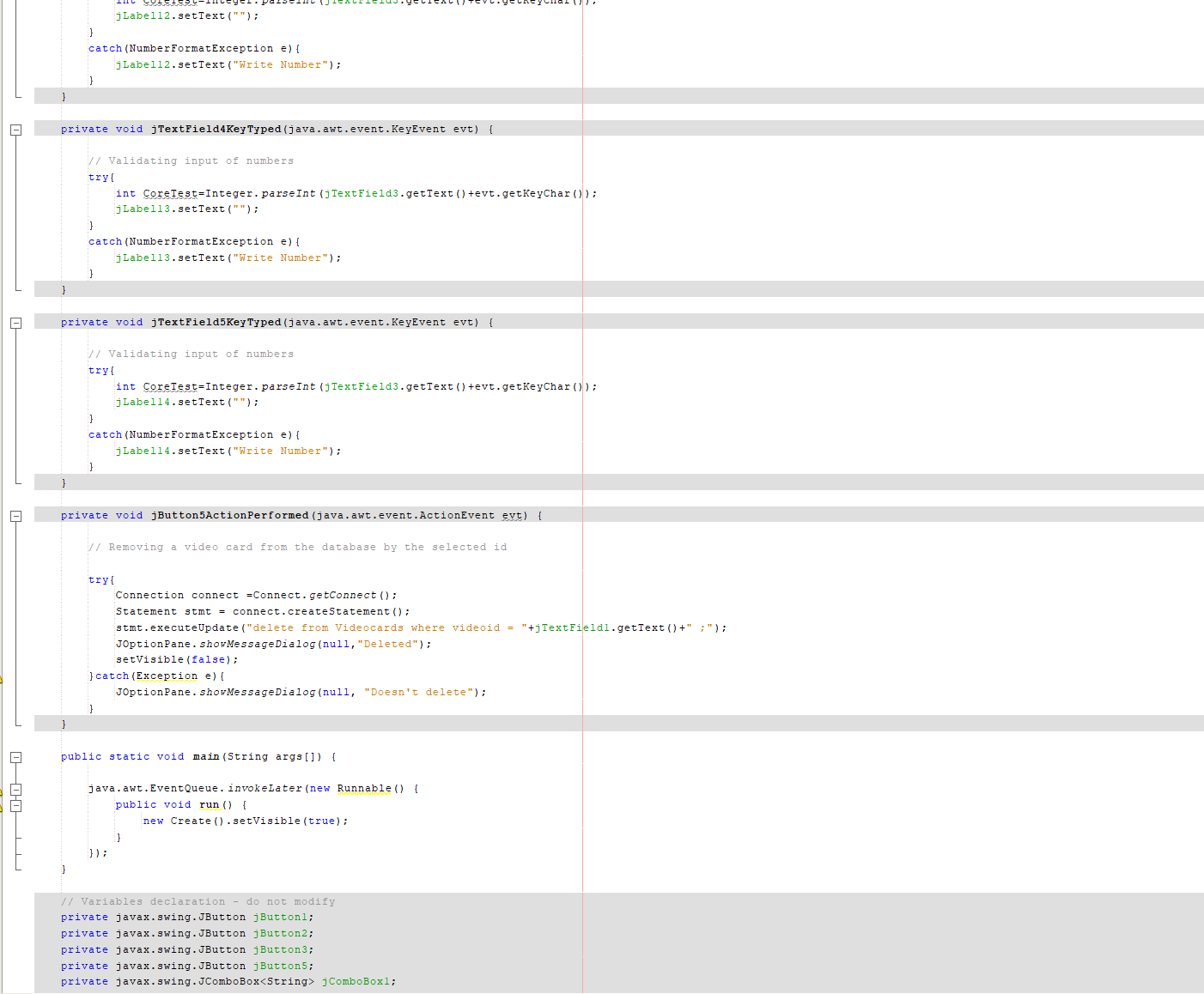
This class contains fields, buttons and their methods. When you enter the available fields in the interface in the correct format (some have a check for entering only numbers), it becomes possible to call the static methods of the "calculator" class to calculate the rest of the parameters from the table. There is a button that allows you to send all the entered information about the video card to the database, data extraction from the fields is based on the jTextField.getText () methods.

There is a dump, a method that allows you to remove the view of a video card from the table by a given id.



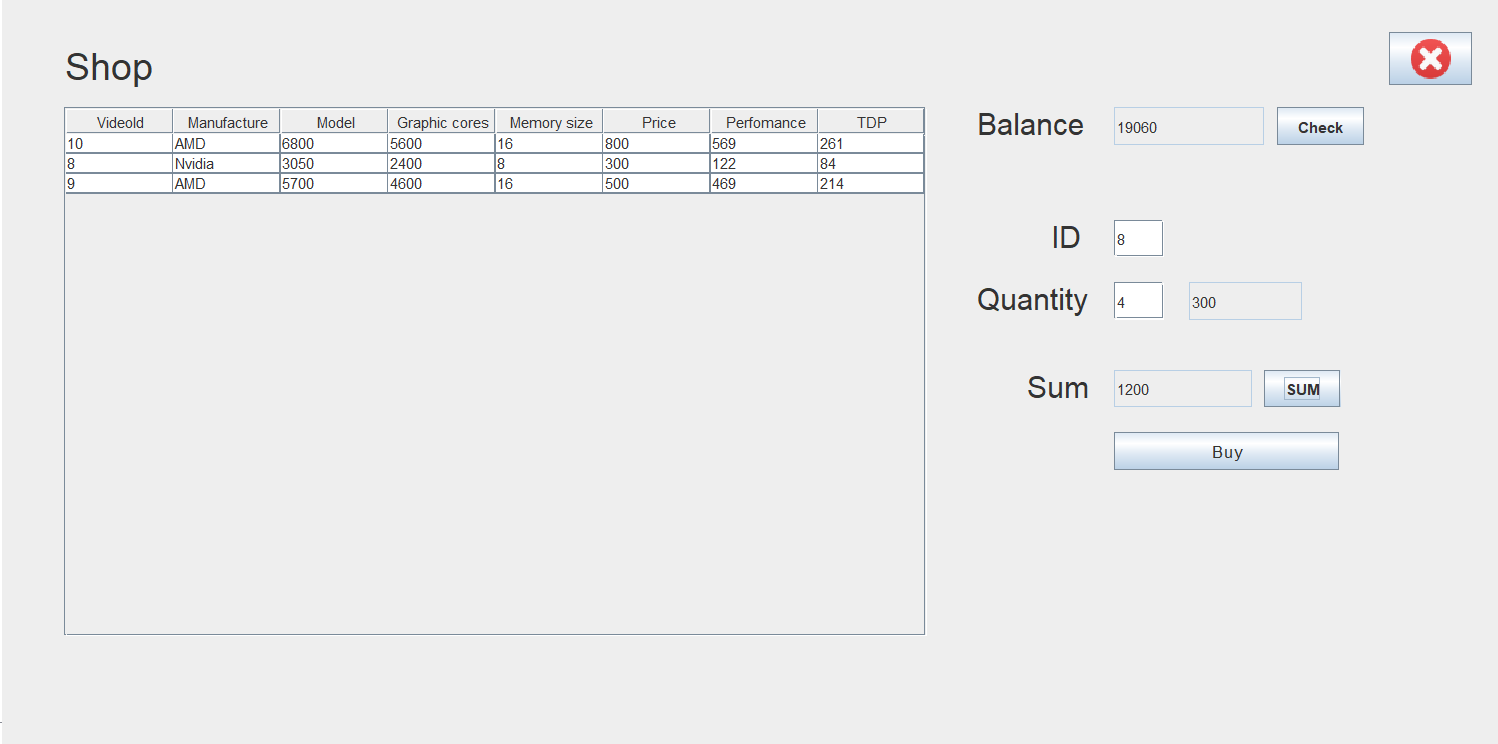




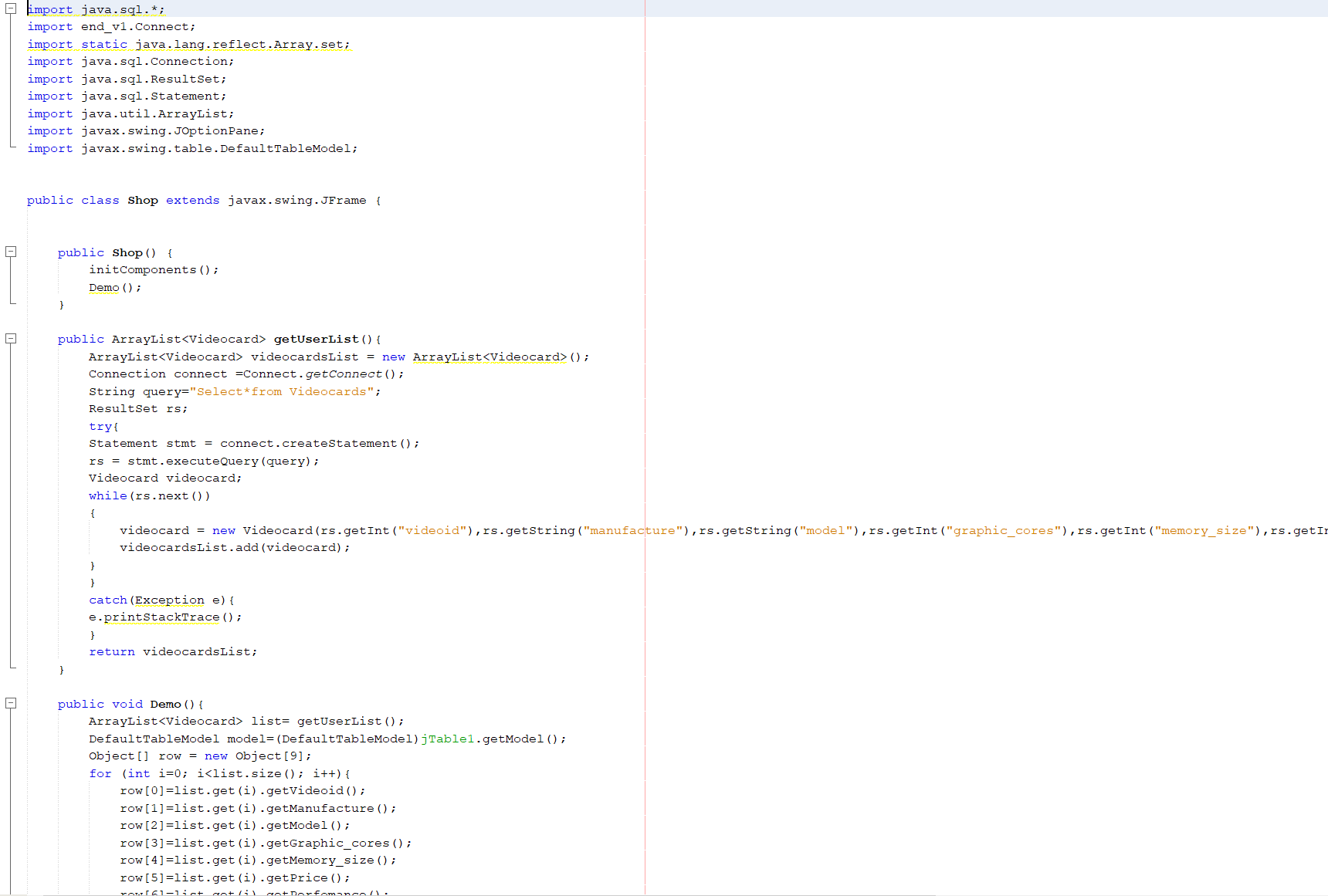


6. Shop

The store class contains a table based on an sql query that collects information into an ArrayList <videocard> (receiving a set of elements from this query), then a method is described that sorts the resulting sheet into a table.

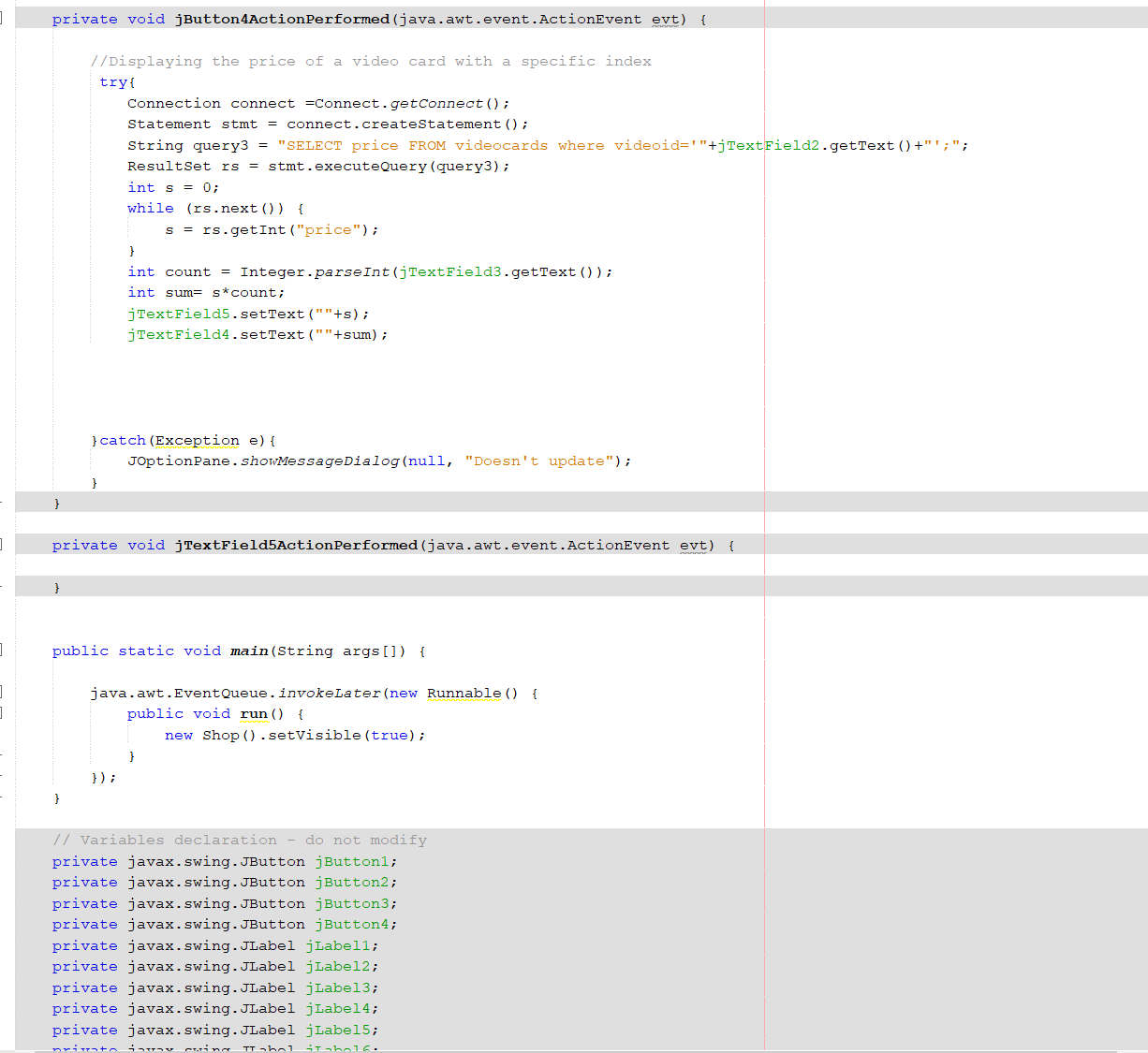


This class contains fields, buttons and their methods. When you enter the available fields in the interface in the correct format (some have a check for entering only numbers), it becomes possible to call the static methods of the "calculator" class to calculate the rest of the parameters. It is possible to check the user's balance from the "Human" table of the database. There is a system for buying a given number of video cards of a certain number. Digging the amount calls a method that calculates the total cost of the order. The Buy button calls a method that allows you to purchase an item for a certain amount; there is a user balance check that does not allow purchasing an item if the order amount is greater than the balance.



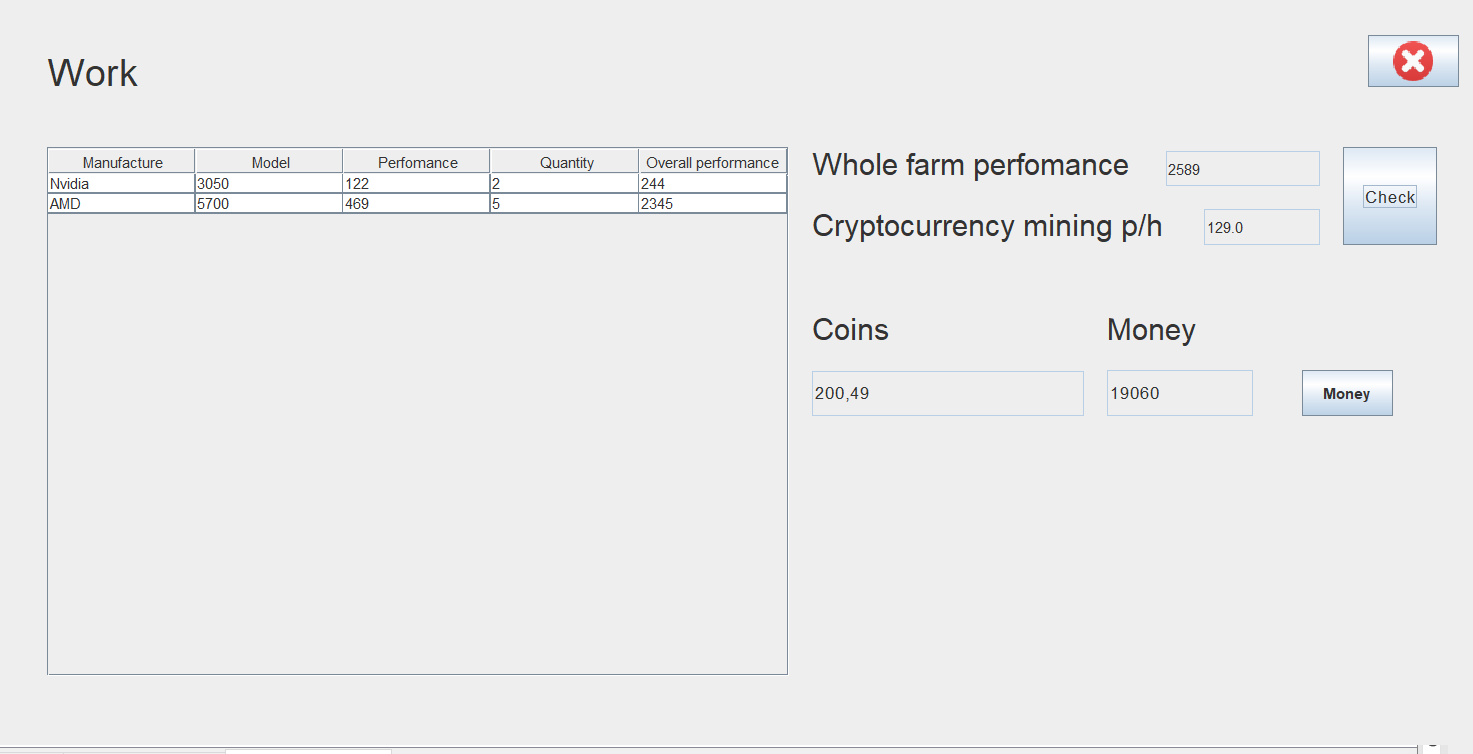
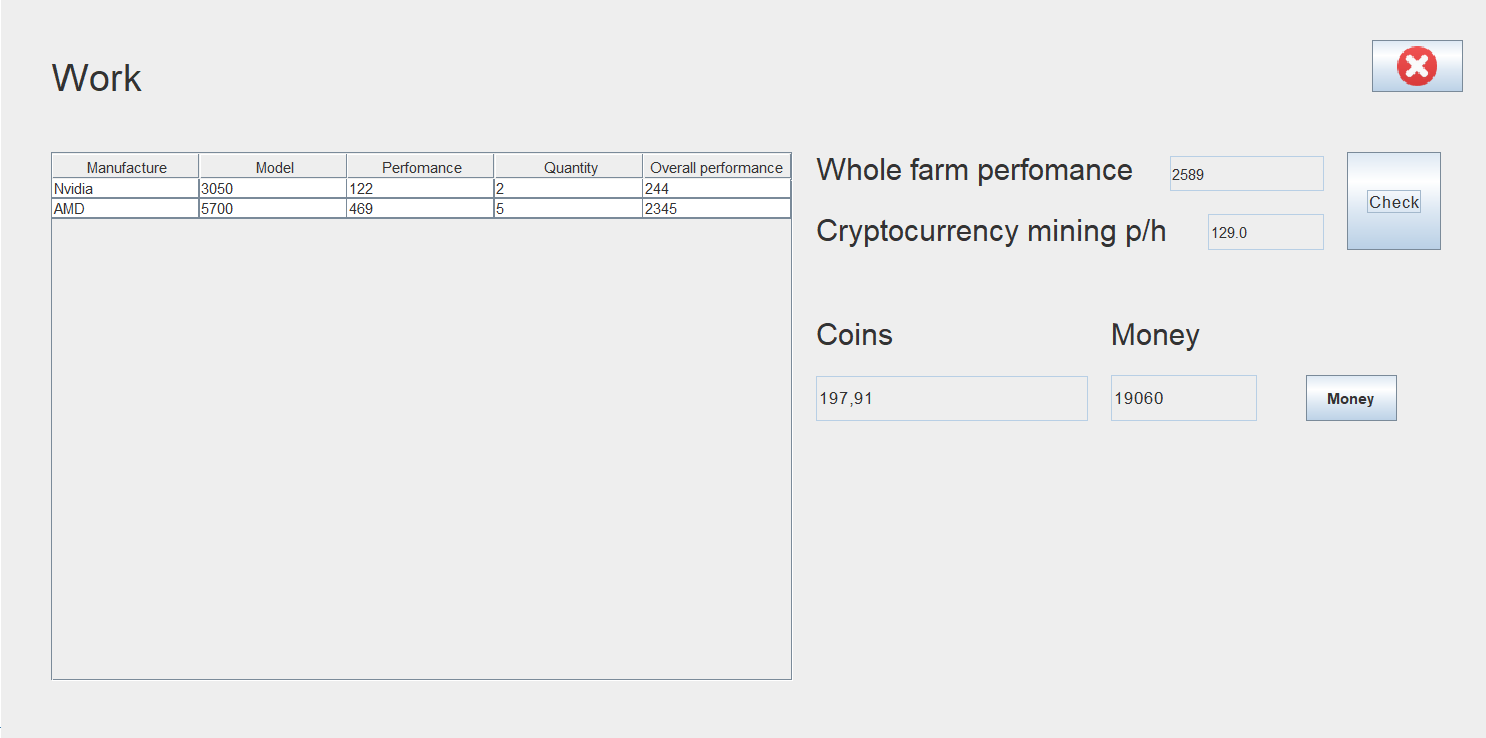


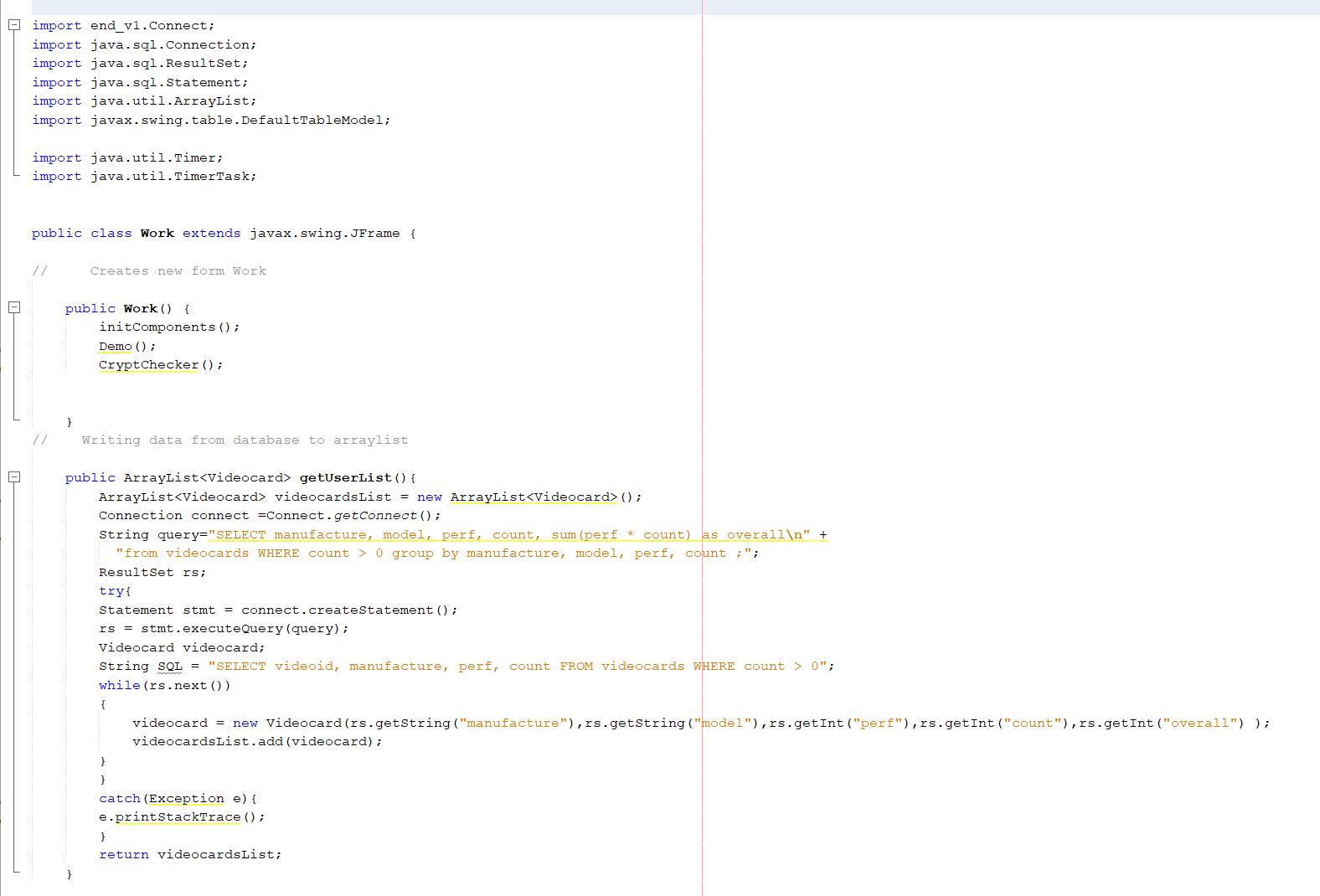




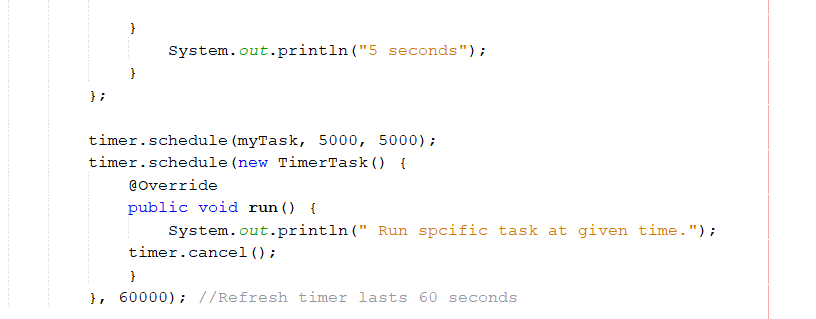
7. Work

The structure of this class resembles the past, with the exception that a metad is created here, which works for 60 seconds (test) and is updated every 5 seconds (util. timer and timerTask are used), this allows you to see the mined cryptocurrency in real time, new values ​​immediately are sent to the database and from there are returned to the program. For calculation, static methods from the calculator class are used.







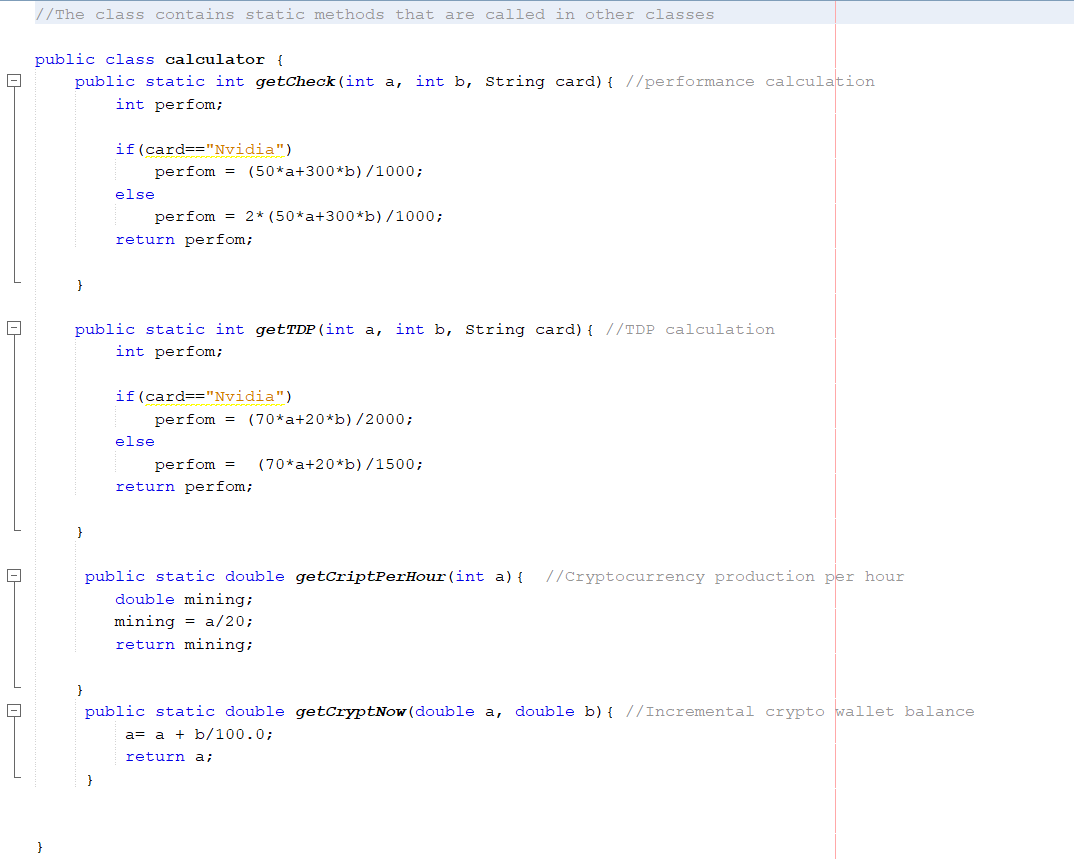






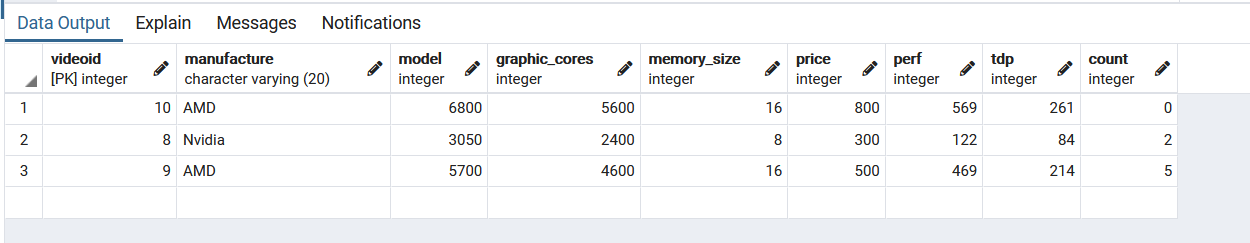
8. Calculator

Contains static methods needed to calculate some parameters. Used in other classes.



DATABASE

Videocards



Human

