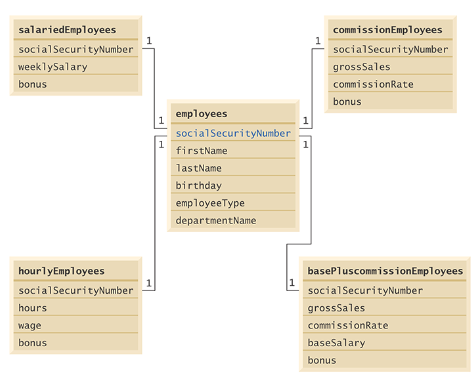
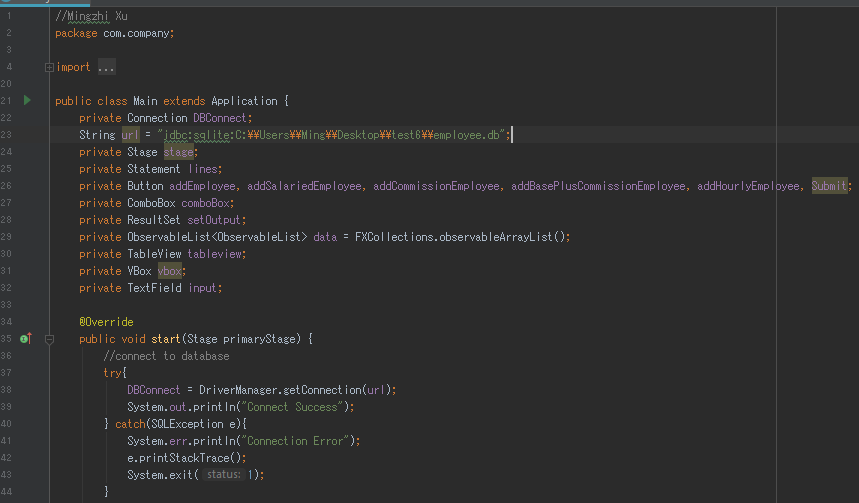
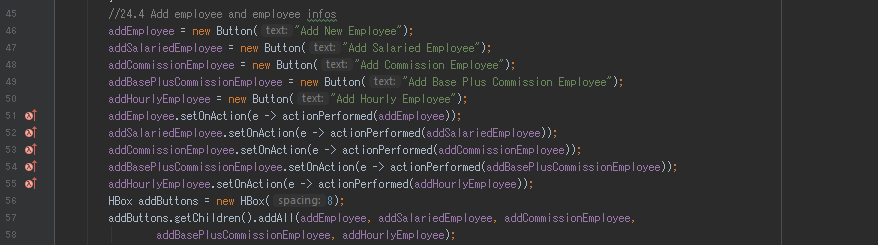
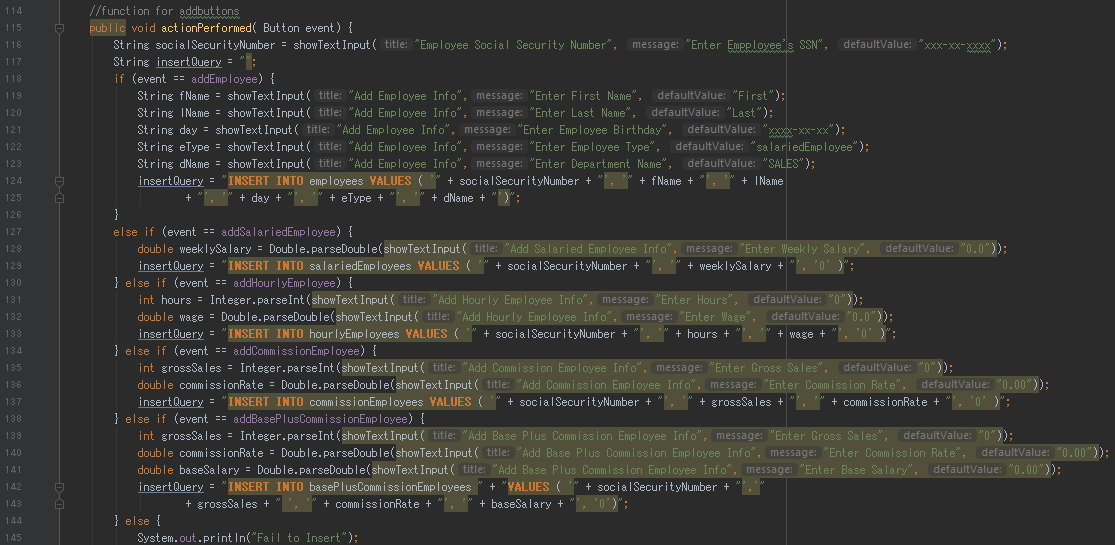
Mingzhi Xu

CSC22100 Software Design Laboratory Fall 2018

Project Employee Database

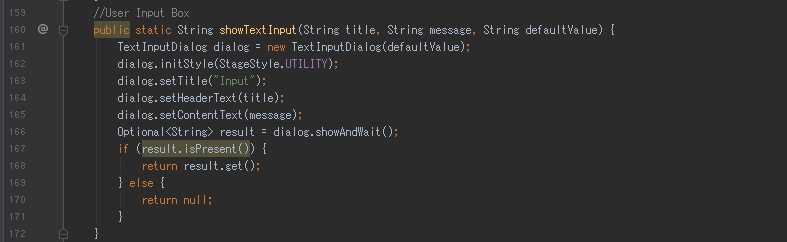
In this project, we will be creating a local database which will store information of the employees and design a user interface to manipulate and display the data from the database. The Employee database follows the hierarchy shown as below.

As we can see, we will have five tables in this Employee database, and the variable social security number exists in all five table which serves as the primary key to connect the tables to the employees table. We will be using SQLite, a relational database management system, to connect the database to the interface.

Now we will create an interface such that allows the user to add employees to the Employee table and payroll information to the appropriate table for each new employee. For example, for a salaried employee add the payroll information to the SalariedEmployees table.  This creates buttons for inserting new employees into the employees table and insert information of provided new employee social security number into the other tables corresponds to the employee’s type. 

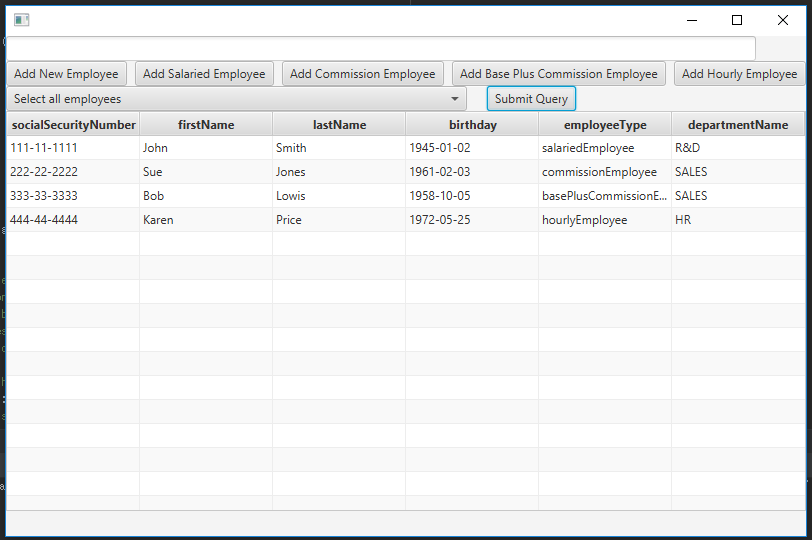


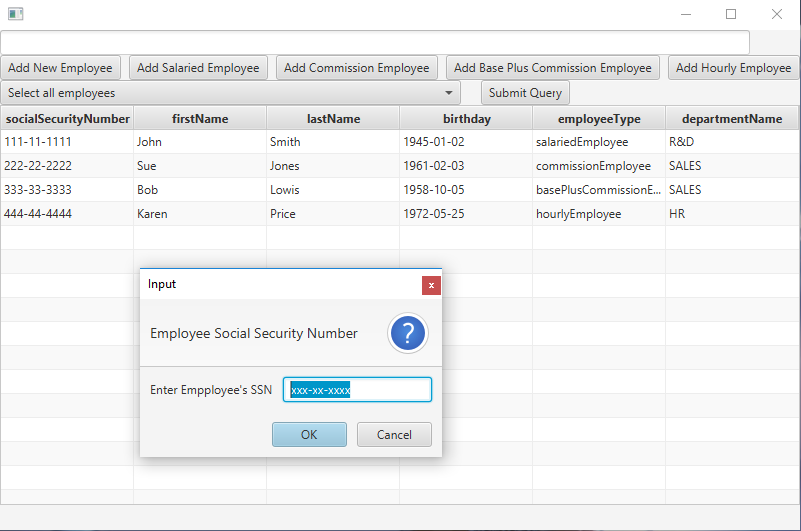
The buttons will perform the actionPerformed function to prompt for user input using the showTextInput function for variables shown in the Employee database hierarchy. Then the user input will be inserted into the prepared statements which will be executed to insert information into the database.

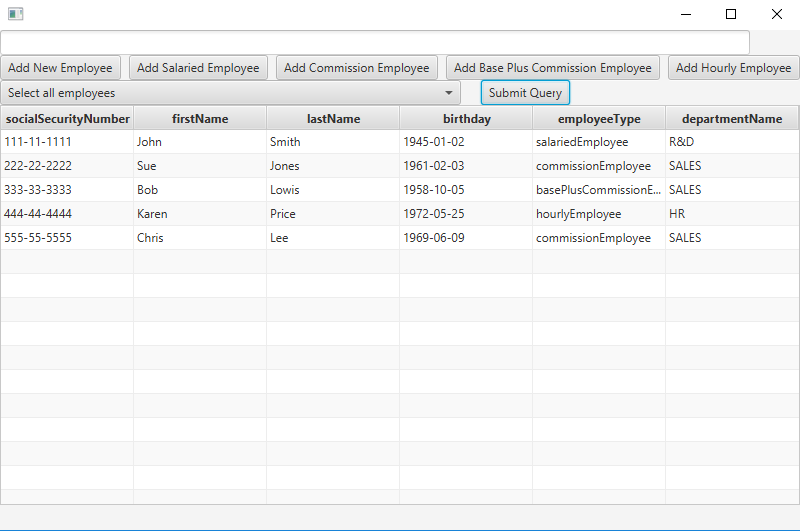


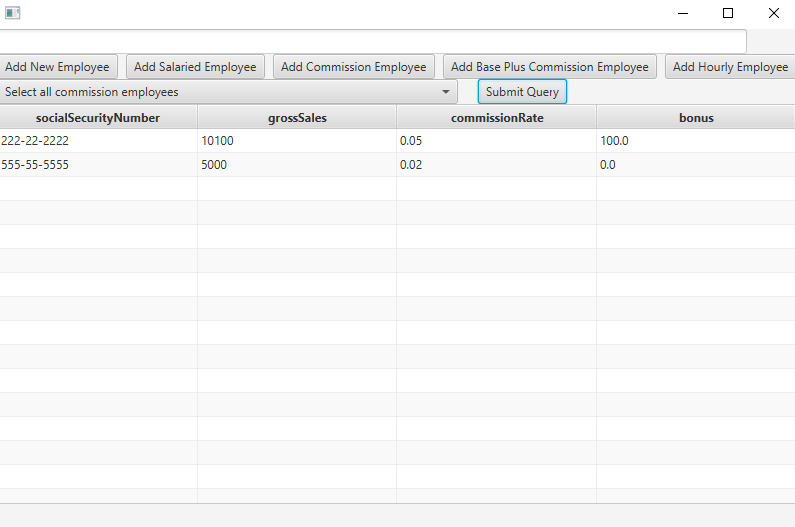
The function showTextInput works similarly to the JOptionPane in swing but since we are using Javafx, we are not able to use JOptionPane.

Result for 24.4

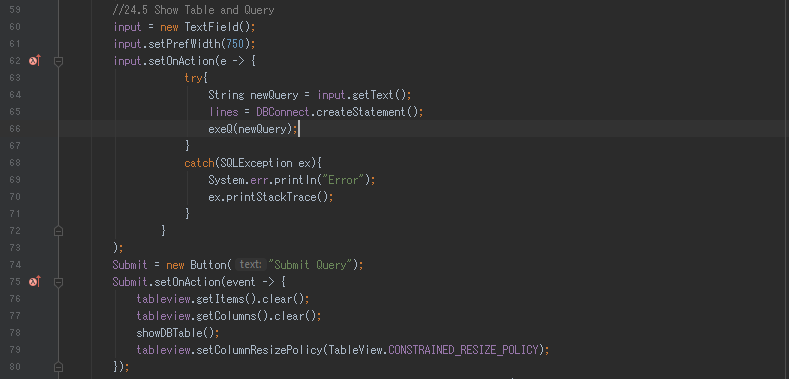


Now we will insert a new employee with the social security number 555-55-5555, first name Chris, last name Lee, birthday 1969-06-09, employee type commissionEmployee, and in the sales department. A window will pop out to prompt for those user inputs.



After that we want to insert payroll information for the new employee, so we look at the employee type which is commissionEmployee, so we click on the button Add Commission Employee and same thing as add employee a pop out window will prompt for user input.

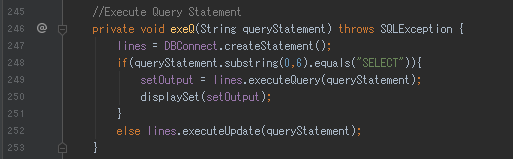
Next, we will modify the interface to contain a combo box and a text area to allow the user to perform a query that is either selected from the combo box or input into the text area. Some predefined queries would include 1) Select all employees working in the department SALES. 2) Select hourly employees working over 30 hours. 3) Select all commission employees in descending order of the commission rate.



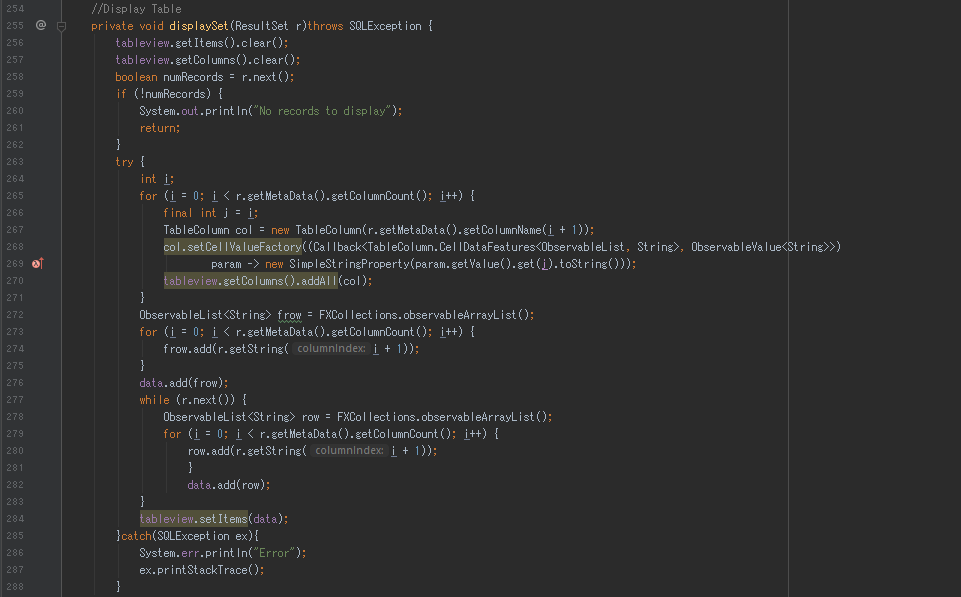
Here we will create a text field which is used for the text area for user input of their own queries and a submit button that will display the table based on the preselected queries in the combo box. Below is list for the combo box and layouts of the interface.





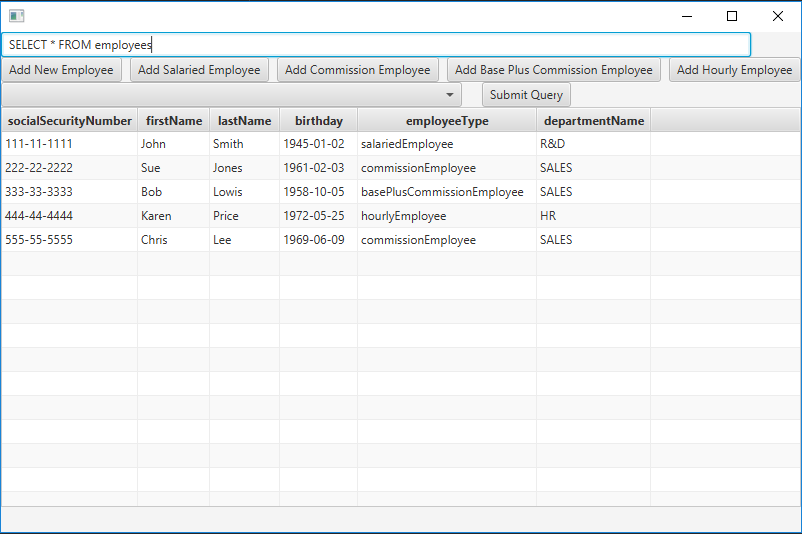
This contains prepared statements for the selected queries that exists in the combo box.

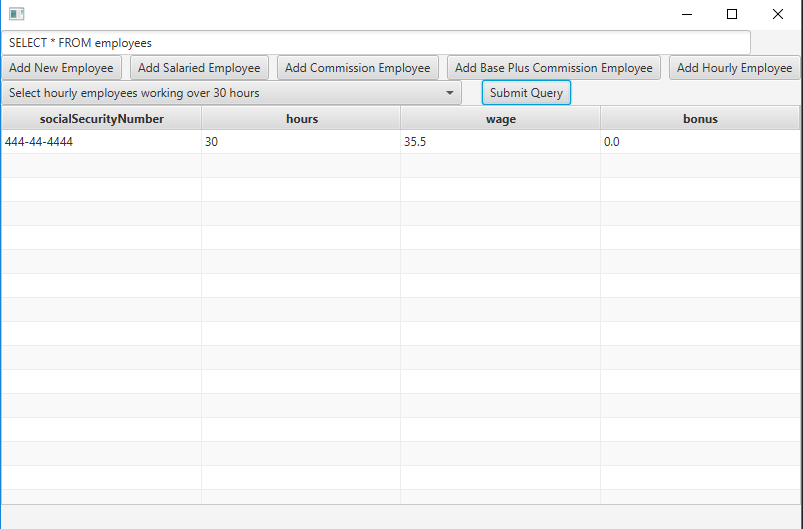
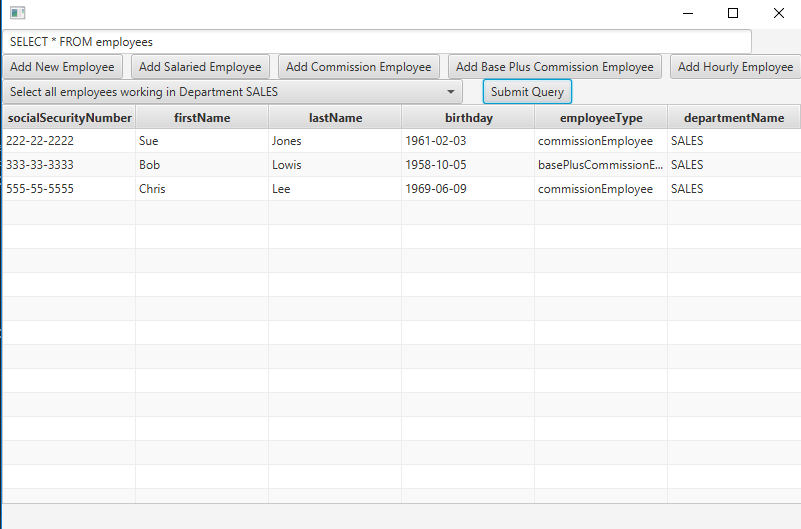
The function exeQ was created so there is not needing to keep calling create statement and execute.

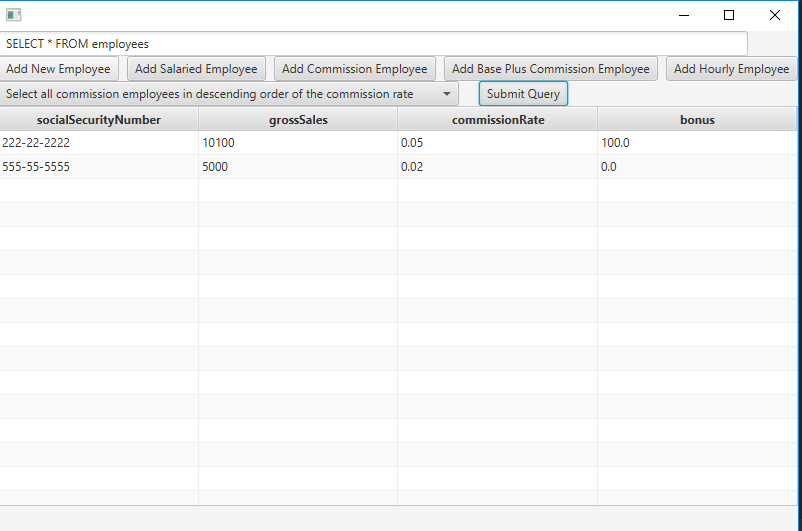


The displaySet will take data from the database and insert into table view which will be displayed on the interface.

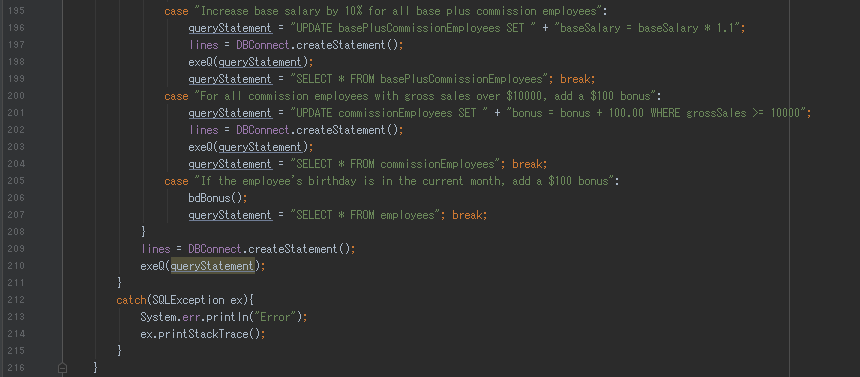
Result for 24.5



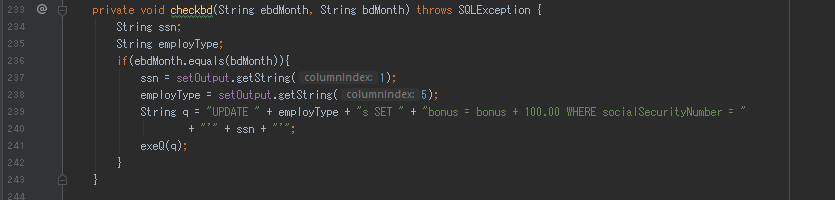
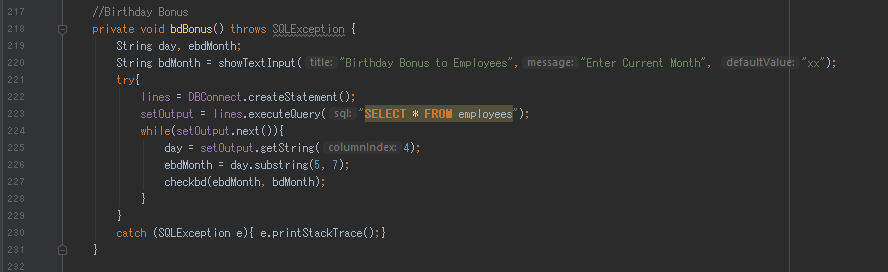




Next We will modify the interface to perform the following tasks, increase base salary by 10% for all base plus commission employees, if the employee’s birthday is in the current month add a $100 bonus, and for all commission employees with gross sales over $10000, add a $100 bonus. These are going to be implemented as predefined queries, so it will be in the combo box as you can see from the query list from above and similarly, we will put these cases into showDBTable function.



However, the case which adds bonus to employee if birthday is in the current month requires you to define the month of the current month, so the function bdBonus is created to prompt for user input for current month and takes data from database to compare and execute the bonus into the payroll information.



Result for 24.6

