

Case :

By applying the principles of OOP (Inheritance, Polymorphism, Abstraction and Encapsulation), create a JAVA program for restaurant management, where:

- a. Restaurant had the following divisions:
 - i. Waiters
 - ii. Chef
 - iii. Financial Depart.
- b. There are 3 types of food and 2 types of drinks, including:
 - i. Food: Dessert, Main Course and Dessert
 - ii. Beverages: Iced dan Hot

The program runs by displaying menu options to the user, where the user here is divided into two parts, namely the customer user and the restaurant user management. User management can enter the system by implementing a login system which is carried out using the constructor() concept in the class (users and passwords are stored using private variables in the user management class). For customer user login is optional, can use the login mechanism or not.

After login process, then:

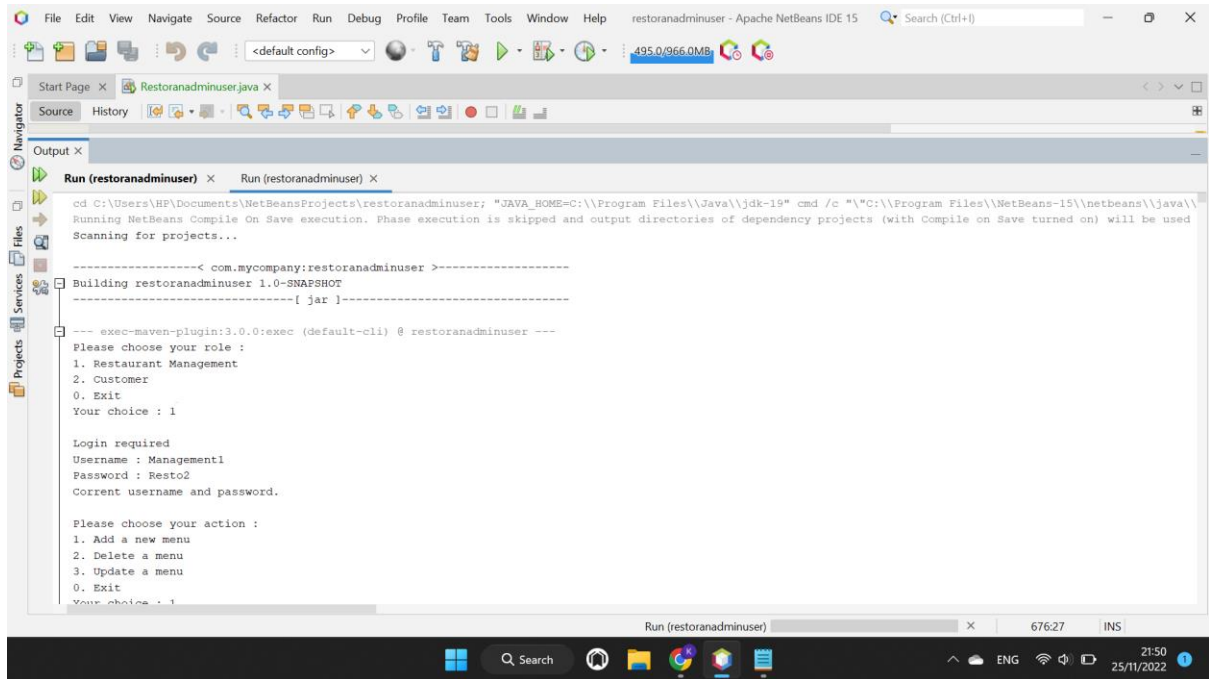
1. Customer can choose menu options, for example when the user chooses food, the menu will display 3 types of food choices, namely Desert, Main Course and Dessert, as well as drink choices. After that, the user can select the "pay" menu, where the output from the menu is the number of menus and orders that have been selected by the user along with the price and total to be paid.
2. For user management itself, they had a menu to add and delete food and drink menus, user management could also edit food and drink menus.

Hint:

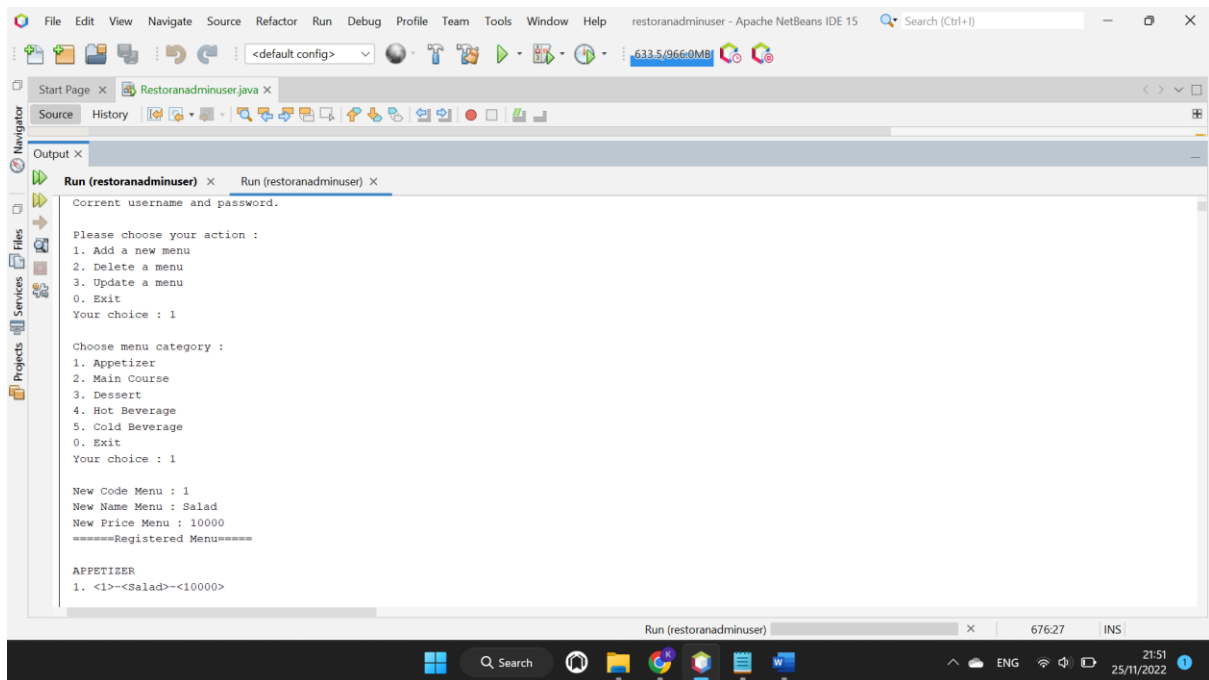
- To make it easier, create a function/method for each activity
- Use the mechanism of polymorphism, abstraction, or inheritance to minimize complexity and shorten code
- The use of a switch case is an interesting option to solve this case, but the use of if-else is also reliable
- There is no standard OOP design limit that you can use, everyone has their own approach

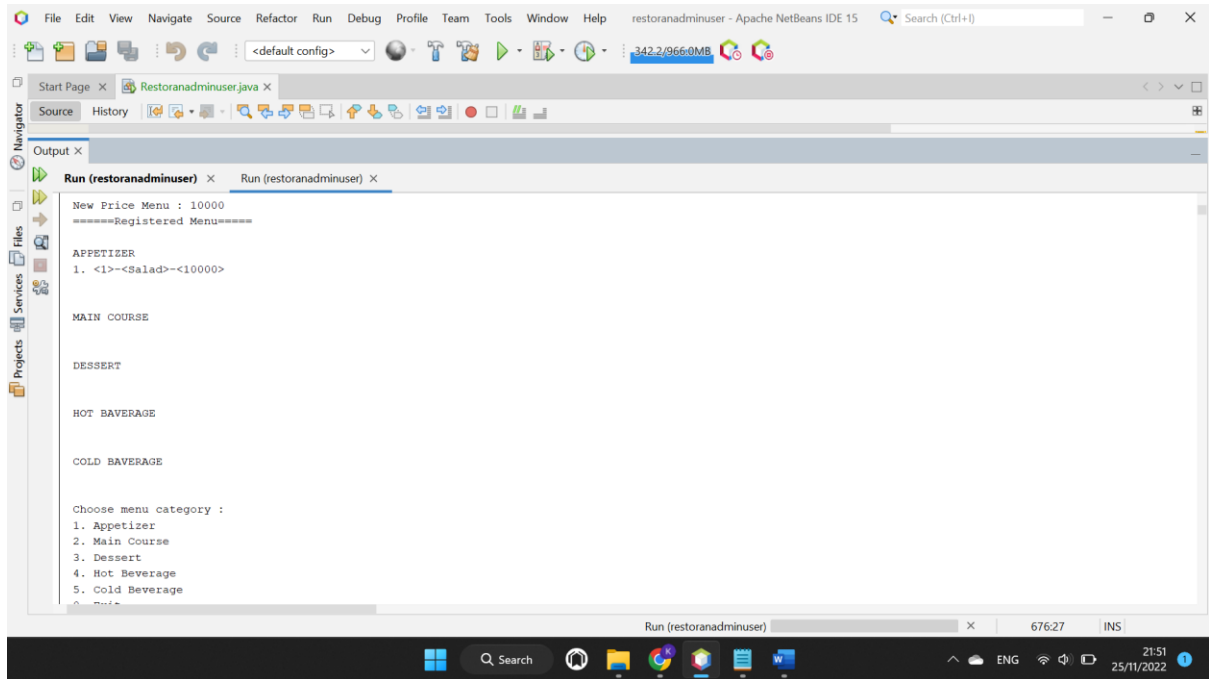
Screenshoot Program

Login sebagai Restaurant Management

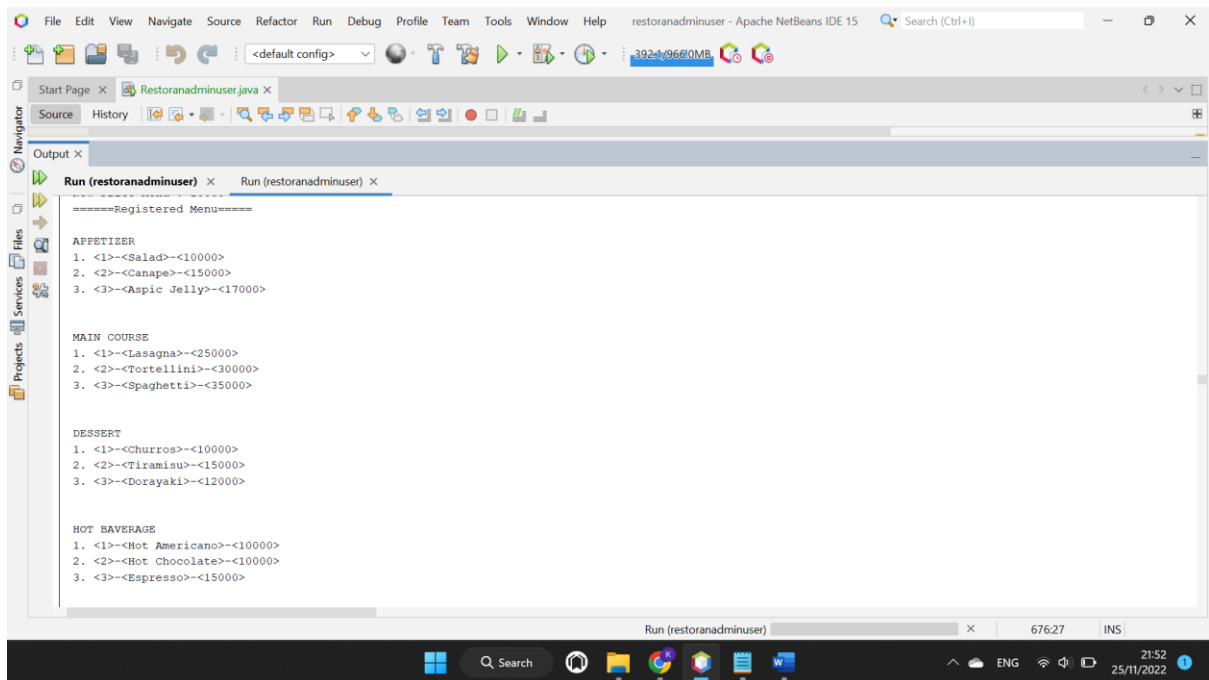


Proses insert

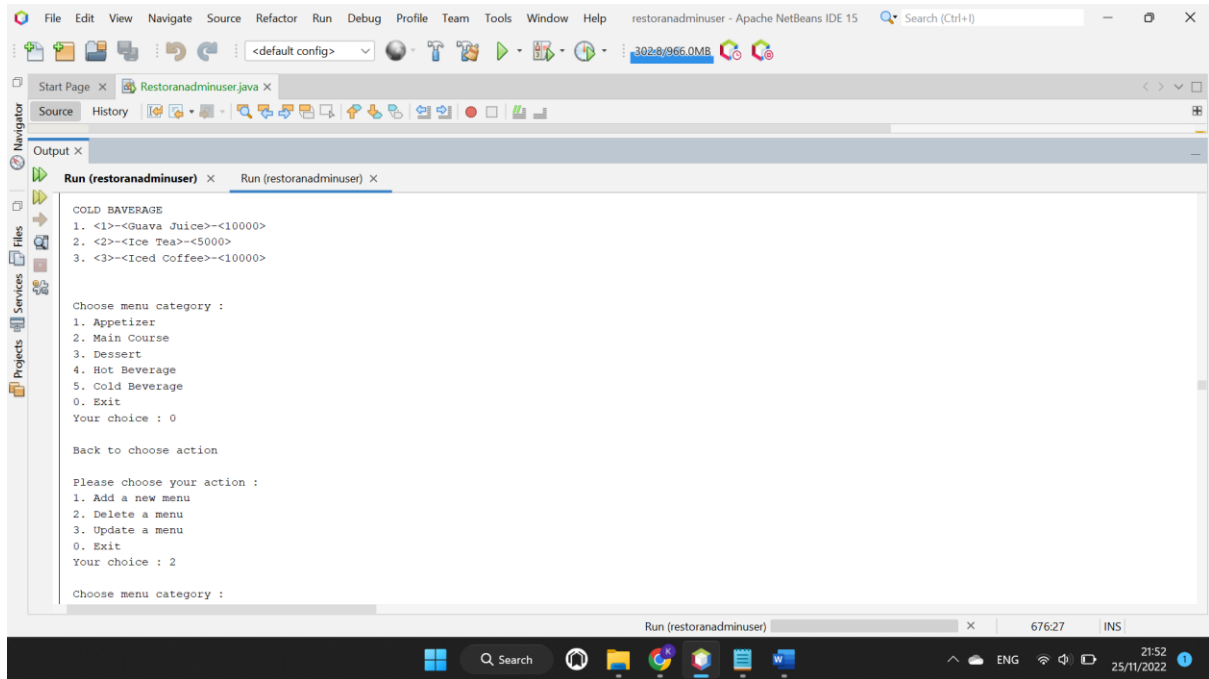




List setelah insert



Proses deleting



The screenshot shows the NetBeans IDE with the 'Run (restoranadminuser)' console window open. The output displays the initial menu structure for a restaurant management system. It starts with 'COLD BEVERAGE' and lists three items: 'Guava Juice' (ID 1, Price 10000), 'Ice Tea' (ID 2, Price 5000), and 'Iced Coffee' (ID 3, Price 10000). The user is prompted to 'Choose menu category' and selects '0' for 'Exit'. Then, the user is prompted to 'Please choose your action' and selects '2' for 'Update a menu'. The console output is as follows:

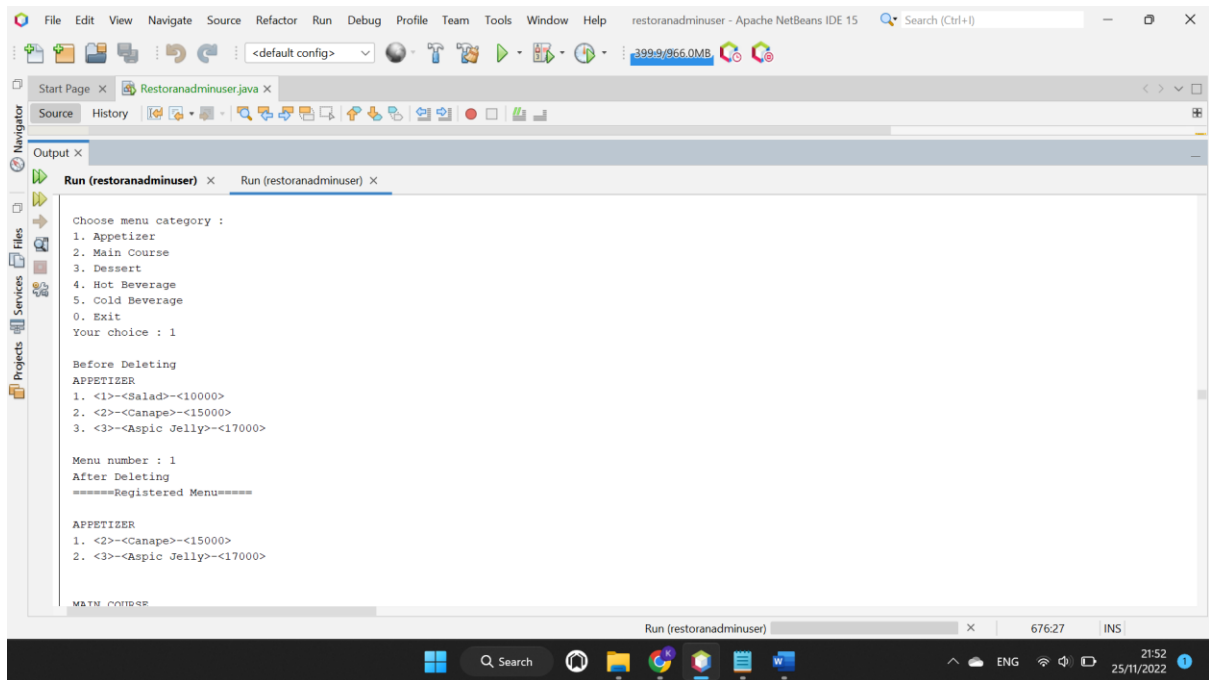
```
COLD BEVERAGE
1. <1>~<Guava Juice>~<10000>
2. <2>~<Ice Tea>~<5000>
3. <3>~<Iced Coffee>~<10000>

Choose menu category :
1. Appetizer
2. Main Course
3. Dessert
4. Hot Beverage
5. Cold Beverage
0. Exit
Your choice : 0

Back to choose action

Please choose your action :
1. Add a new menu
2. Delete a menu
3. Update a menu
0. Exit
Your choice : 2

Choose menu category :
```



The screenshot shows the NetBeans IDE with the 'Run (restoranadminuser)' console window open. The output displays the menu structure after deleting the 'Appetizer' category. It shows 'Before Deleting APPETIZER' with three items: 'Salad' (ID 1, Price 10000), 'Canape' (ID 2, Price 15000), and 'Aspic Jelly' (ID 3, Price 17000). The user is prompted to 'Menu number' and selects '1'. Then, the user is prompted to 'After Deleting' and the output shows the updated menu structure for 'APPETIZER' with two items: 'Canape' (ID 2, Price 15000) and 'Aspic Jelly' (ID 3, Price 17000). The console output is as follows:

```
Choose menu category :
1. Appetizer
2. Main Course
3. Dessert
4. Hot Beverage
5. Cold Beverage
0. Exit
Your choice : 1

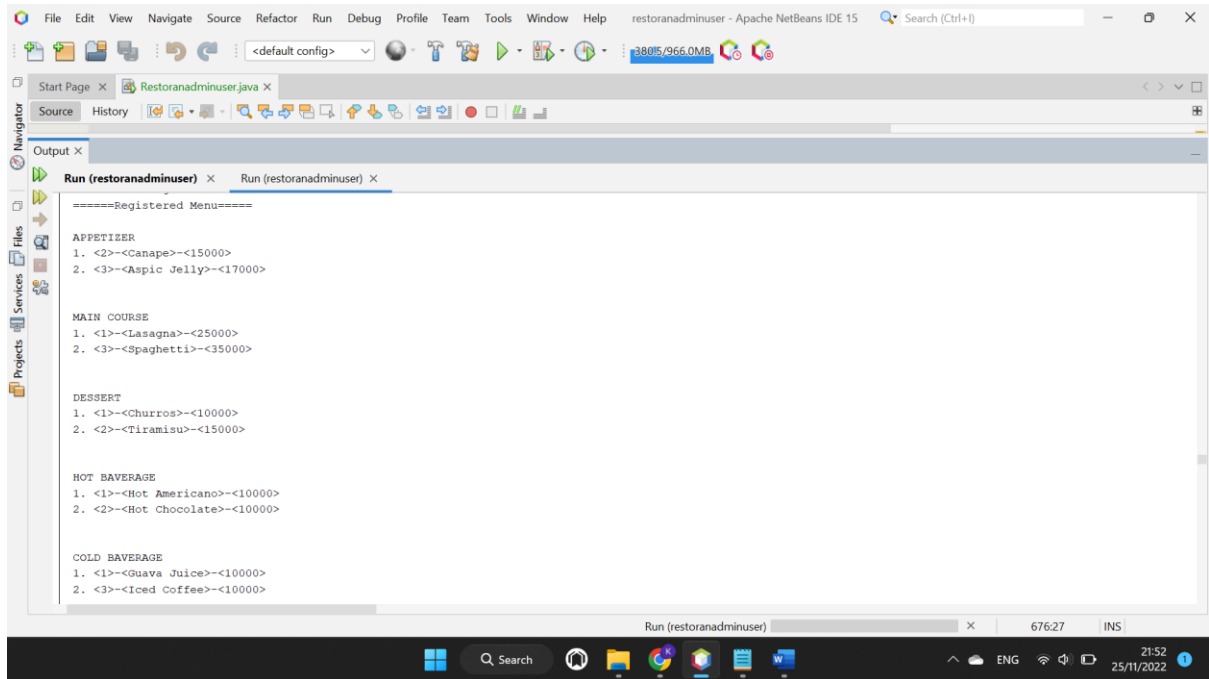
Before Deleting
APPETIZER
1. <1>~<Salad>~<10000>
2. <2>~<Canape>~<15000>
3. <3>~<Aspic Jelly>~<17000>

Menu number : 1
After Deleting
=====Registered Menu=====

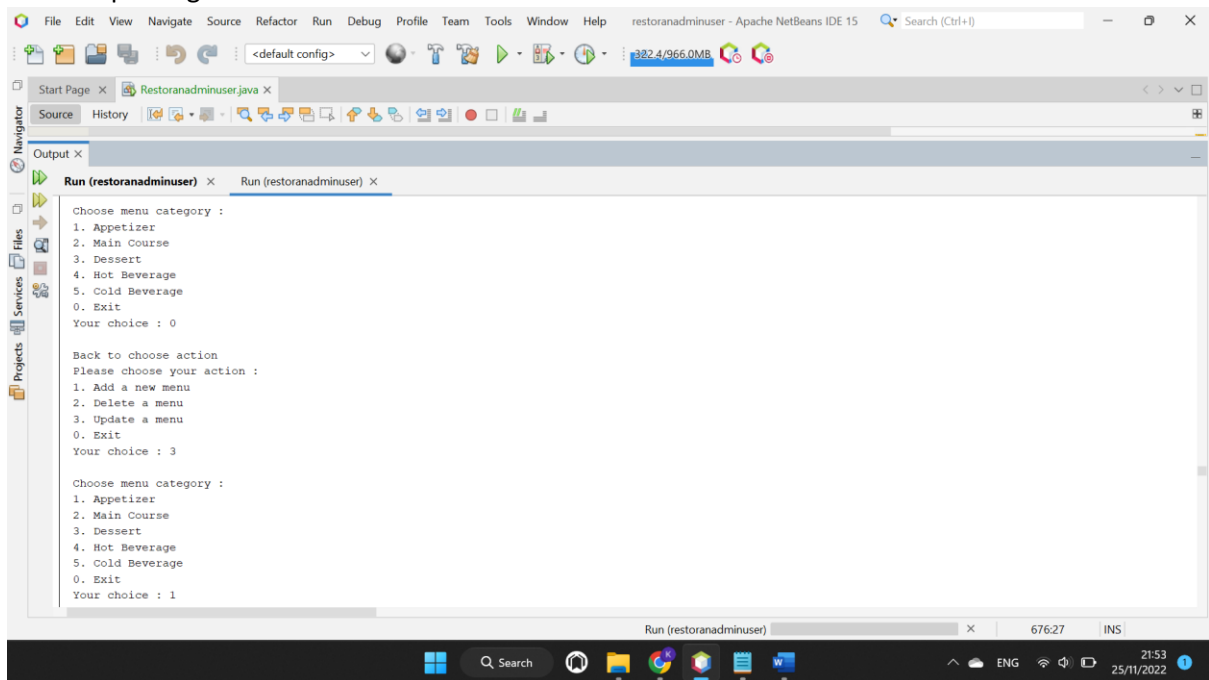
APPETIZER
1. <2>~<Canape>~<15000>
2. <3>~<Aspic Jelly>~<17000>

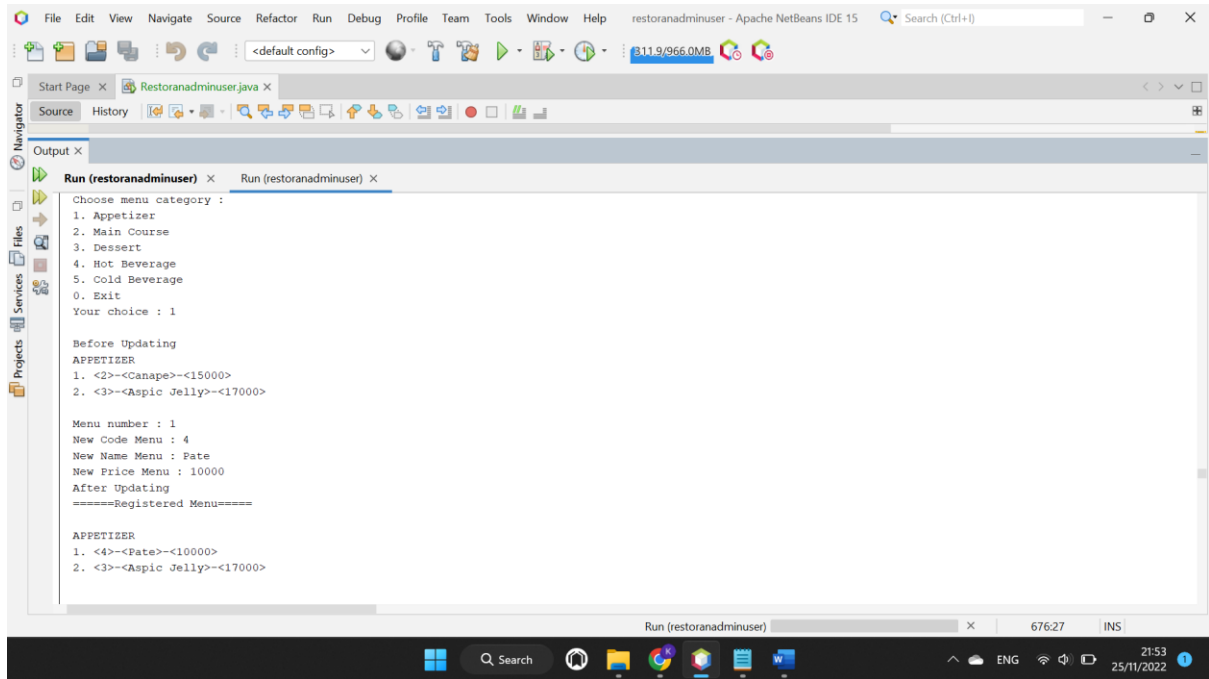
MAIN COURSE
```

List menu setelah deleting

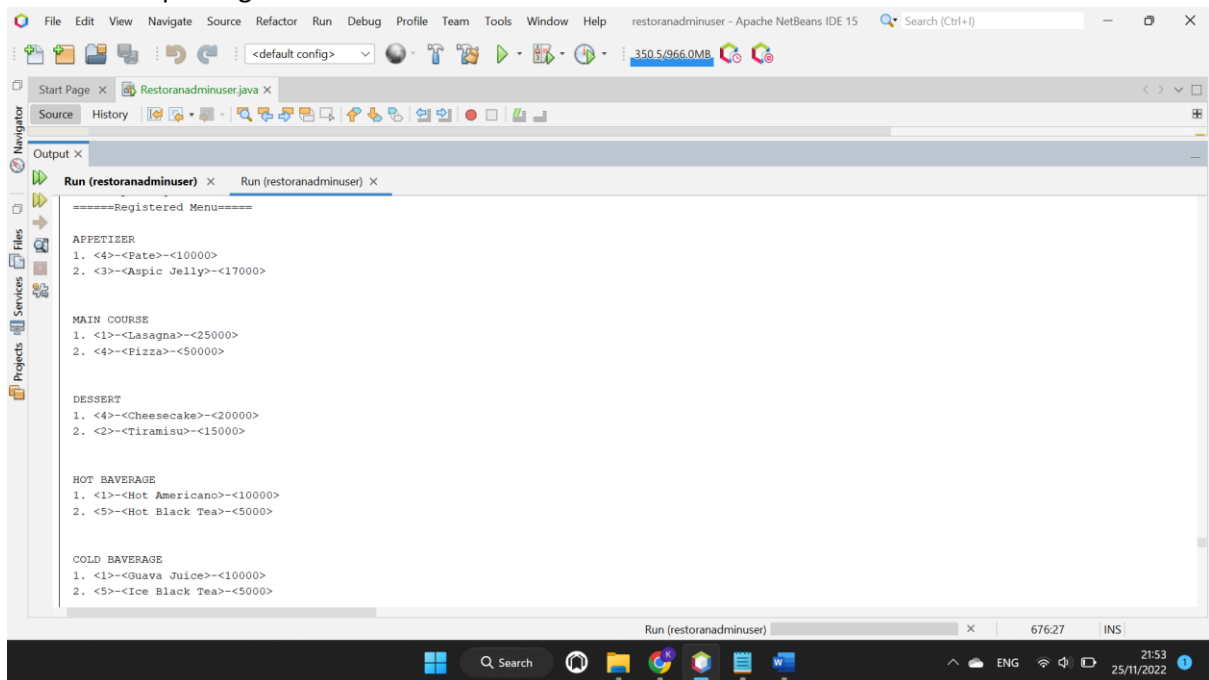


Proses Updating

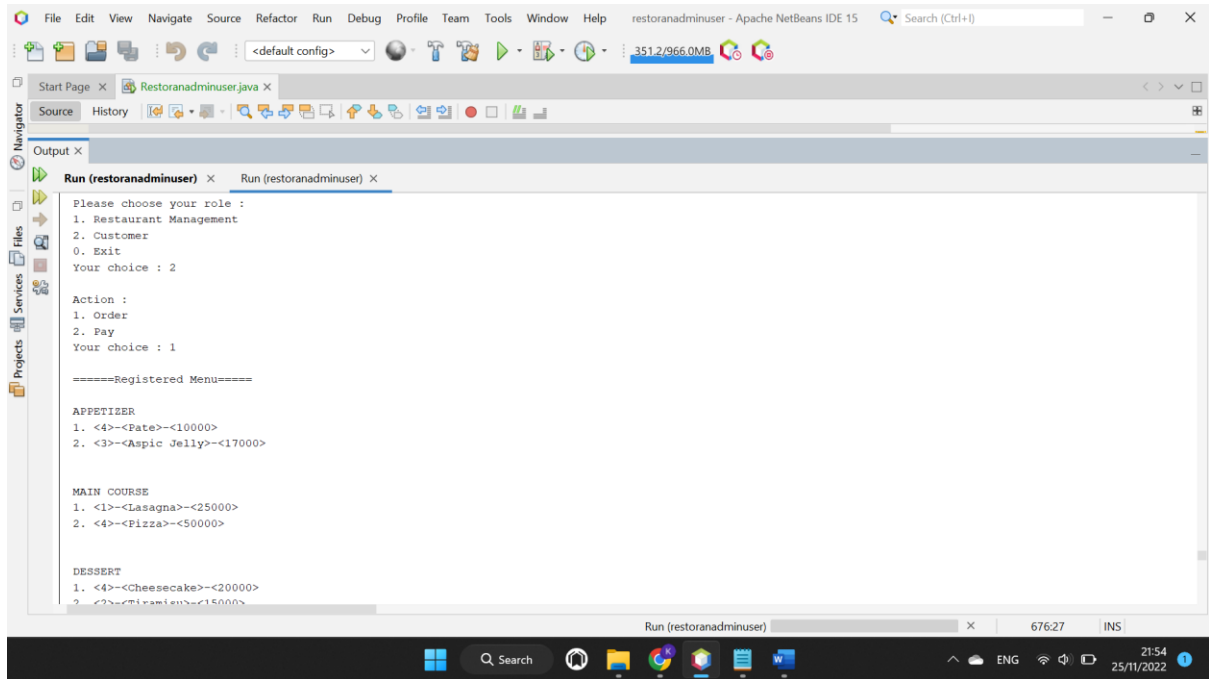




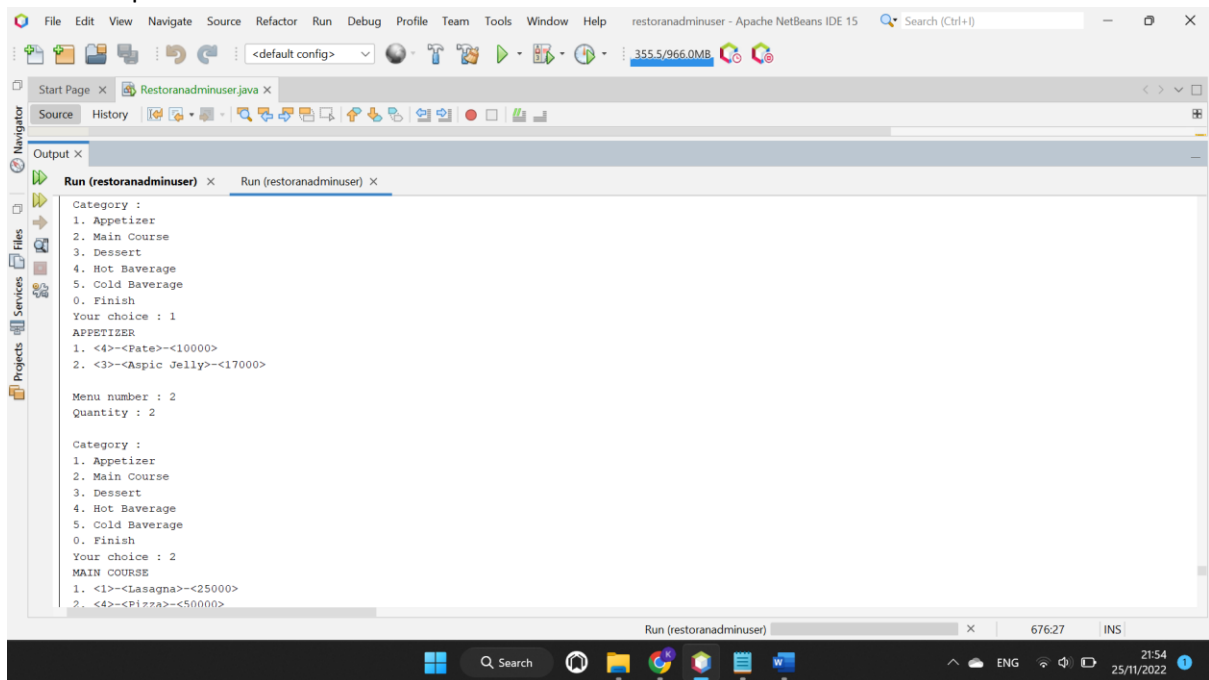
List setelah updating



Masuk sebagai customer



Pemilihan pesanan



Pembayaran

