Raha company
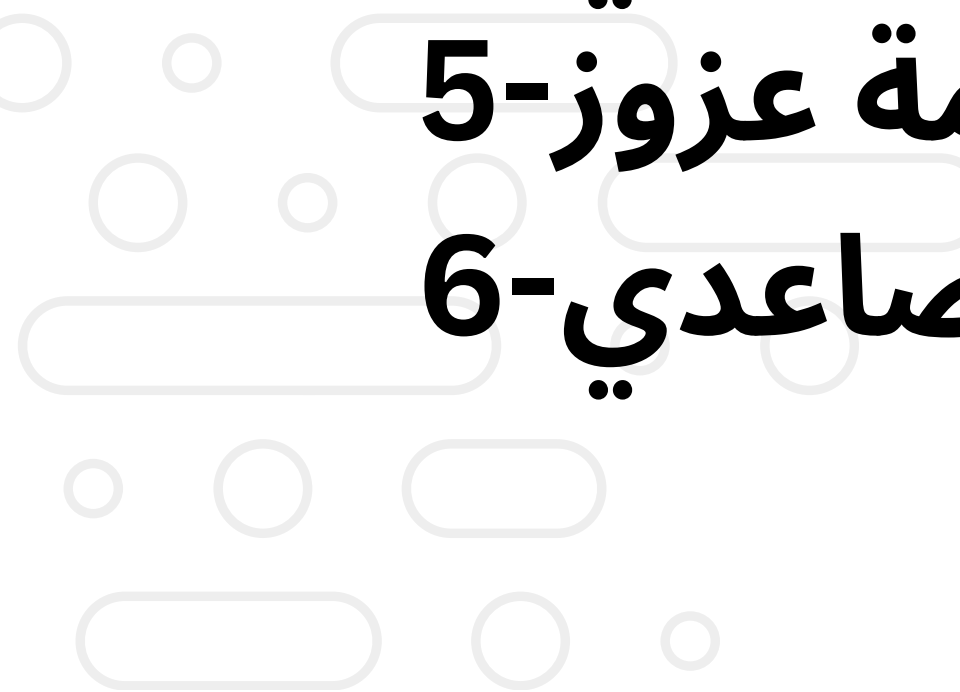
**Group members:**

1-فرح ابراهيم العتيبي
2-ريم احمد الحربي
3-ريماس عبد الستار ابوخليل
4-اروى سلطان السلمي
5-نجاة اسامة عزوز
6-لمى بندر الصاعدي

Description of the project

Our project is about application system raha company for domestic maids. Firstly, we ask the user to either log in or create an account if they haven't already. Then, we provide them with a list of services, which are either hourly maids or resident maids. The user can choose one of them. Let's assume the user chooses an hourly maid. After that, they can select one of the available packages: a single visit or a monthly subscription. With the monthly subscription, the maid visits every week for six days. As for resident maids, there's a variety of subscription contracts available, allowing the user to freely choose the payment method, either in installments or full payment. There are two ways for the user to pay for their order: cash or credit card. The user's order will be stored in an external file, including the order invoice and all the necessary information

Class Maid
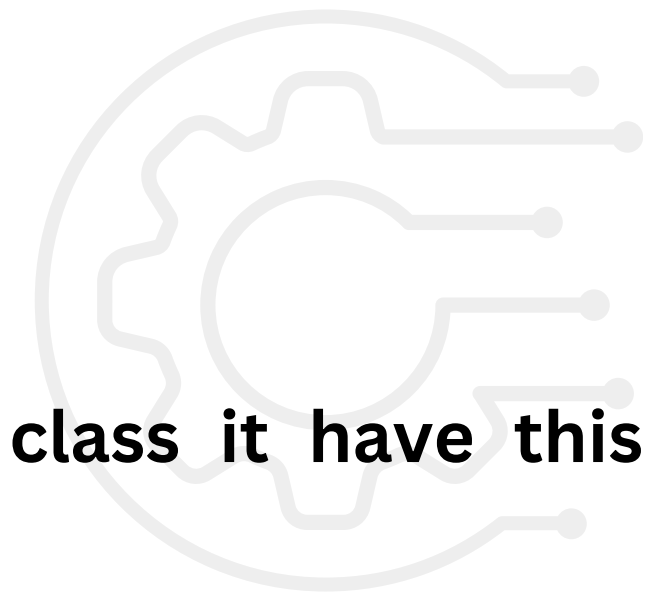
It provides the general characteristics of both ResidentMaid class and HourMaid class it have this attributes
1-int Nationality
2-int price
3-int Years_Experirnce
ana list that have all available
Nationality
private final String[]Nationality_list
with size 3

and have Default constructor Initializes nationality, price, and years of experience to 0.
and paramitrized constructor also have all setter and gettr for to access to attributes for
the methos it hava FillPrint_Nationality_list to dell with fill and display all nationality
and to string to print the class.
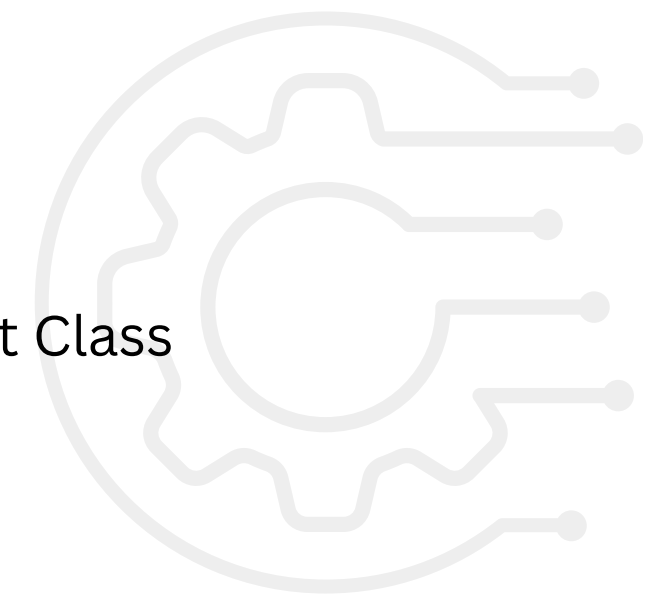
Class ResidentMaid

The ResidentMaid class allows the users to choose the duration of services and the pay type they prefer

Class UserInfo

It appears to the user when he opens the application, and he can choose to create a new account if he does not have a previous account and enter his information, such as his name, age, email, phone number, location, and password. If he has a previous account, he can log in using his email and password, and he can also log out if he wants when finished.

ClassCreditCard

CreditCard class that extend Payment Class to represents a credit card payment method and it extend Payment Class

1- instance variables:
creditCardNum: store the credit card number
cardHolderName: store the card holder name
cvv :store the cvv

2- The class has two constructors:
  - The default constructor sets all instance variables to default values
  - The parameterized constructor takes four arguments: credit card number, cardholder's name, CVV, and a price. and sets values

3- getter and setter methods for the instance variables

4-makePayment method: is overridden from the superclass Payment, prints a message indicating that a payment was made with the credit card number and the amount paid.

5- toString method :is overridden from the Object class. It returns a string representation of the CreditCard object

6-equals method :is overridden to compare two CreditCard objects for equality. It checks if the credit card number of the current object is equal to the credit card number of the object passed as a parameter.

Class Cash

Cash class that extends Payment class and represents a payment made in cash.

instance variable:
  pricePaid: store the amount of cash paid

The class has two constructors:
    - The default constructor sets the pricePaid variable to 0.0
    - The parameterized constructor takes two arguments: the amount of cash paid (pricePaid) and the total price of.
      the service and sets these values

getter and setter methods for the pricePaid variable

The makePayment method:is overridden from the superclass Payment, calculates the remaining amount and prints a message indicating the amount paid, the service price, and the remaining amount.

toString method :is overridden from the Object class. It returns a string representation of the CreditCard object

equals method :is overridden to compare two Cash objects for equality. It checks if pricePaid of the current object is equal to the pricePaid of the object passed as a parameter.
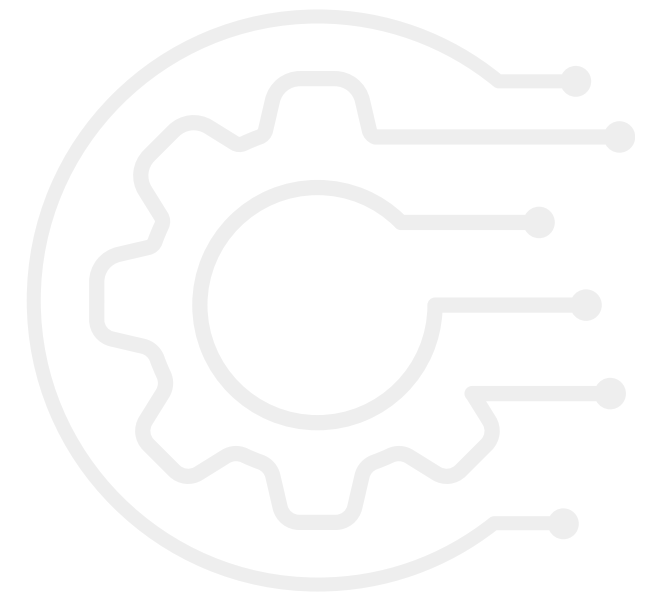
Class Payment

CreditCard class that extend Payment Class to represents a credit card payment method and it extend Payment Class

1- instance variables:
creditCardNum: store the credit card number
cardHolderName: store the card holder name
cvv :store the cvv

2- The class has two constructors:
 - The default constructor sets all instance variables to default values
 - The parameterized constructor takes four arguments: credit card number, cardholder's name, CVV, and a price. and sets values

3- getter and setter methods for the instance variables

4-makePayment method: is overridden from the superclass Payment, prints a message indicating that a payment was made with the credit card number and the amount paid.

5- toString method :is overridden from the Object class. It returns a string representation of the CreditCard object

6-equals method :is overridden to compare two CreditCard objects for equality. It checks if the credit card number of the current object is equal to the credit card number of the object passed as a parameter.

Class HourMaid

The HourMaid class extends the Maid class. It has the following attributes:

1-String Duration
2-String Package
3-String[] hourlyservice
4-String[] vist_duration
The class has two constructors: a default constructor and a parameterized constructor. It also includes setter and getter methods for the Duration and Package attributes. Additionally, it provides methods to display all available durations and hourly services. Finally, it overrides the toString method to print the class information.
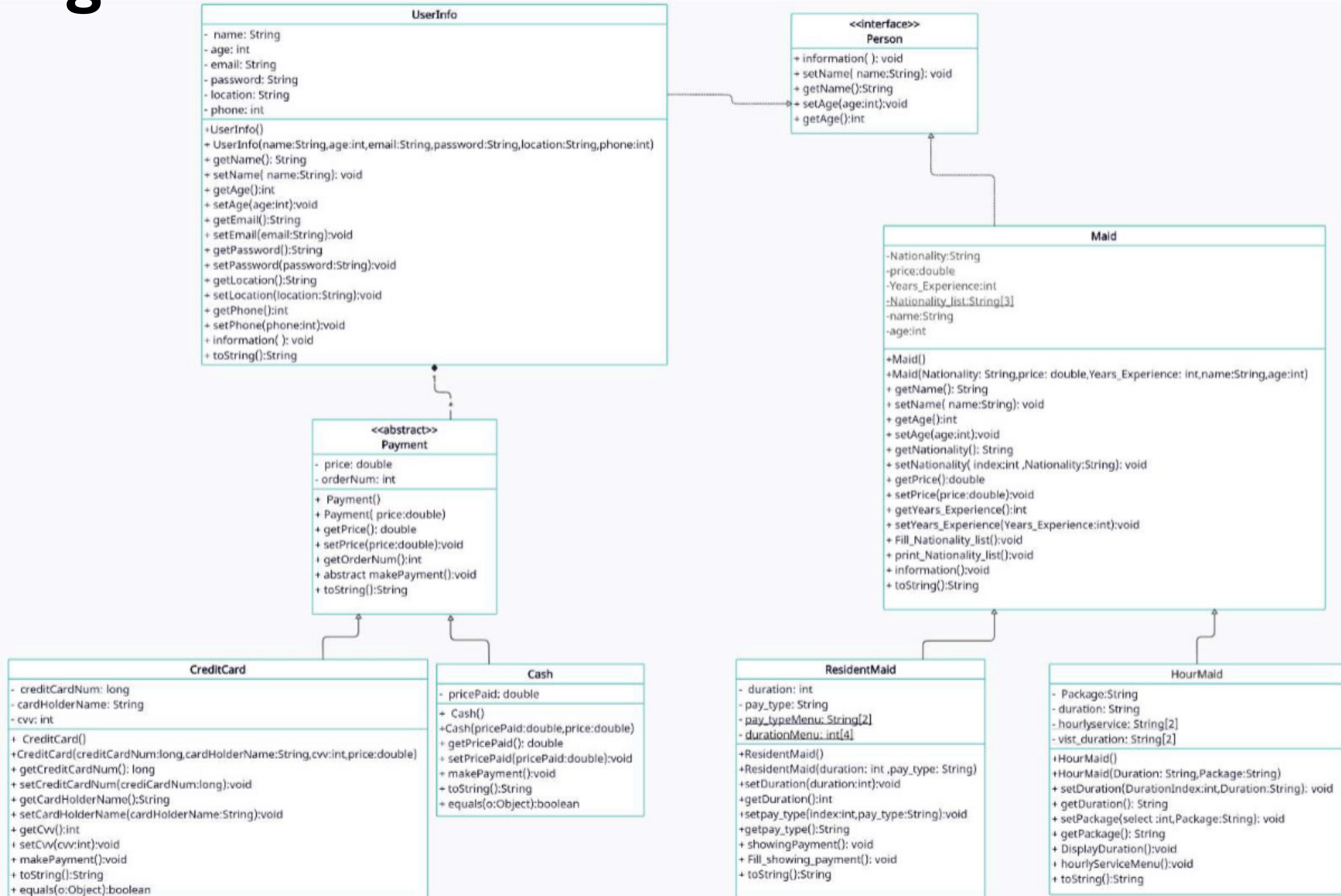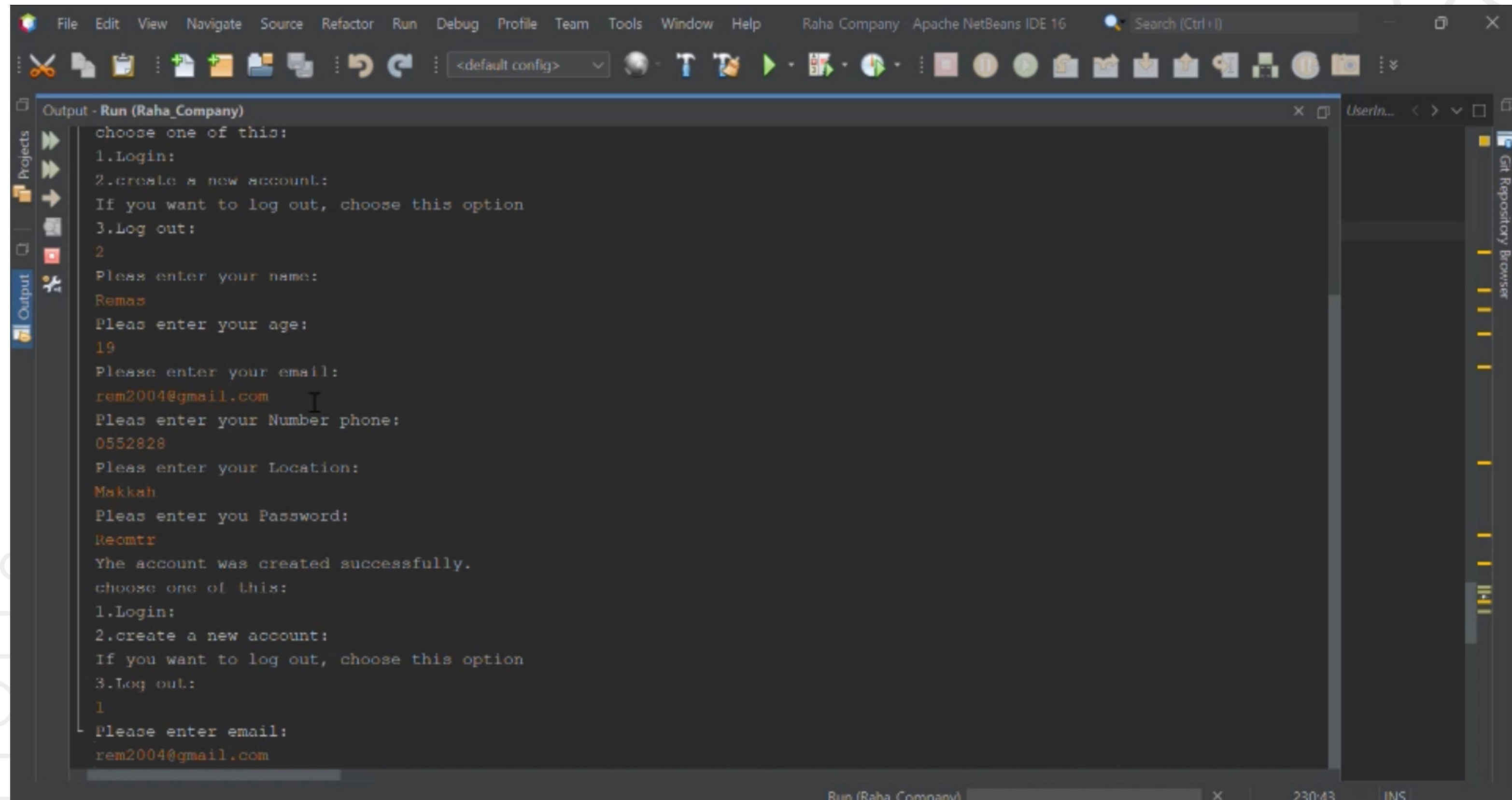
Person interface
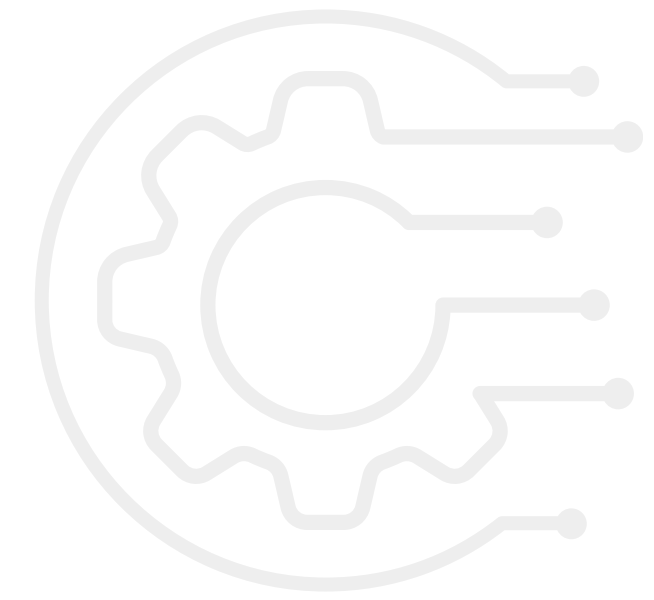
It contains 3 abstract methods set and gate name and set and age and one more Information its for print just geneal information for security , each of thia methods has a specific functionality process in the classes that implemented by it Maid and UserInfo.
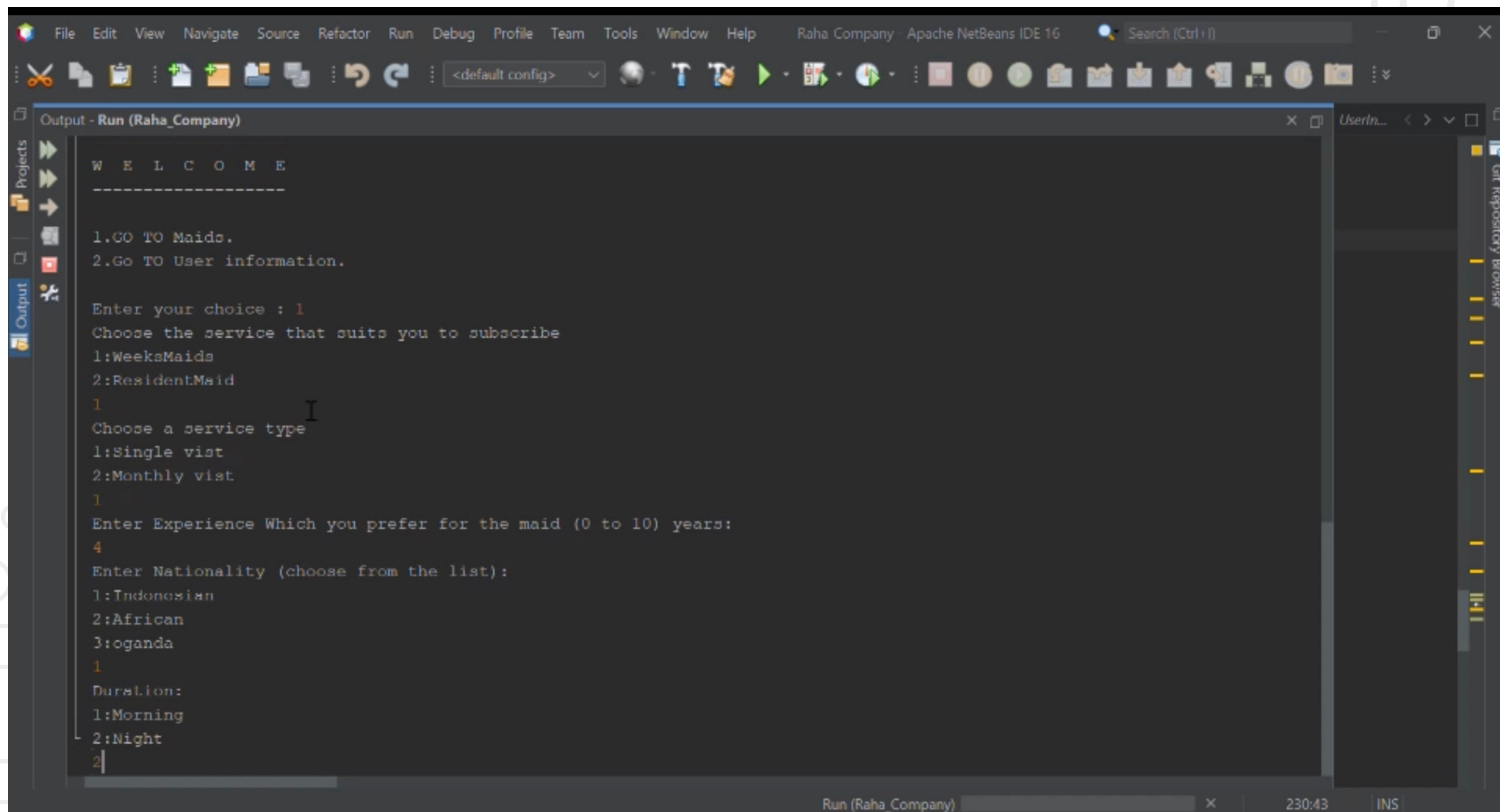
Class diagrams

UserInfo

- name: String
- age: int
- email: String
- password: String
- location: String
- phone: int

+UserInfo()
+ UserInfo(name:String,age:int,email:String,password:String,location:String,phone:int)
+ getName(): String
+ setName(name:String): void
+ getAge():int
+ setAge(age:int):void
+ getEmail():String
+ setEmail(email:String):void
+ getPassword():String
+ setPassword(password:String):void
+ getLocation():String
+ setLocation(location:String):void
+ getPhone():int
+ setPhone(phone:int):void
+ information( ): void
+ toString():String

<<interface>>
Person

+ information( ): void
+ setName(name:String): void
+ getName():String
+ setAge(age:int):void
+ getAge():int

Maid

-Nationality:String
-price:double
-Years_Experience:int
-Nationality_list:String[3]
-name:String
-age:int

+Maid()
+Maid(Nationality: String,price: double,Years_Experience: int,name:String,age:int)
+ getName(): String
+ setName(name:String): void
+ getAge():int
+ setAge(age:int):void
+ getNationality(): String
+ setNationality(index:int ,Nationality:String): void
+ getPrice():double
+ setPrice(price:double):void
+ getYears_Experience():int
+ setYears_Experience(Years_Experience:int):void
+ Fill_Nationality_list():void
+ print_Nationality_list():void
+ information():void
+ toString():String

<>
Payment

- price: double
- orderNum: int

+ Payment()
+ Payment(price:double)
+ getPrice(): double
+ setPrice(price:double):void
+ getOrderNum():int
+ abstract makePayment():void
+ toString():String

CreditCard

- creditCardNum: long
- cardHolderName: String
- cvv: int

+ CreditCard()
+CreditCard(creditCardNum:long,cardHolderName:String,cvv:int,price:double)
+ getCreditCardNum(): long
+ setCreditCardNum(crediCardNum:long):void
+ getCardHolderName():String
+ setCardHolderName(cardHolderName:String):void
+ getCvv():int
+ setCvv(cvv:int):void
+ makePayment():void
+ toString():String
+ equals(o:Object):boolean

Cash

- pricePaid: double

+ Cash()
+Cash(pricePaid:double,price:double)
+ getPricePaid(): double
+ setPricePaid(pricePaid:double):void
+ makePayment():void
+ toString():String
+ equals(o:Object):boolean

ResidentMaid

- duration: int
- pay_type: String
- pay_typeMenu: String[2]
- durationMenu: int[4]

+ResidentMaid()
+ResidentMaid(duration: int ,pay_type: String)
+setDuration(duration:int):void
+getDuration():int
+setpay_type(index:int,pay_type:String):void
+getpay_type():String
+ showingPayment(): void
+ Fill_showing_payment(): void
+ toString():String

HourMaid

- Package:String
- duration: String
- hourlyservice: String[2]
- vist_duration: String[2]

+HourMaid()
+HourMaid(Duration: String,Package:String)
+ setDuration(DurationIndex:int,Duration:String): void
+ getDuration(): String
+ setPackage(select :int,Package:String): void
+ getPackage(): String
+ DisplayDuration():void
+ hourlyServiceMenu():void
+ toString():String

The user here created an account and logged in and chose to go to Maid and chose a WeekMaid service of the type of single visit and with four years of experience of Indonesian nationality she comes at night on Monday. The value of the service appeared to her and she chose to pay by credit card and it was printed for her a message was paid from the credit card
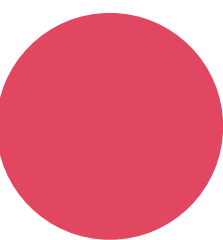


```
Output - Run (Raha_Company)

choose one of this:
1.Login:
2.create a new account:
If you want to log out, choose this option
3.Log out:
2
Pleas enter your name:
Remas
Pleas enter your age:
19
Please enter your email:
rem2004@gmail.com
Pleas enter your Number phone:
0552828
Pleas enter your Location:
Makkah
Pleas enter you Password:
Reomtr
Yhe account was created successfully.
choose one of this:
1.Login:
2.create a new account:
If you want to log out, choose this option
3.Log out:
1
Please enter email:
rem2004@gmail.com
```

Output - Run (Raha_Company)   UserIn...
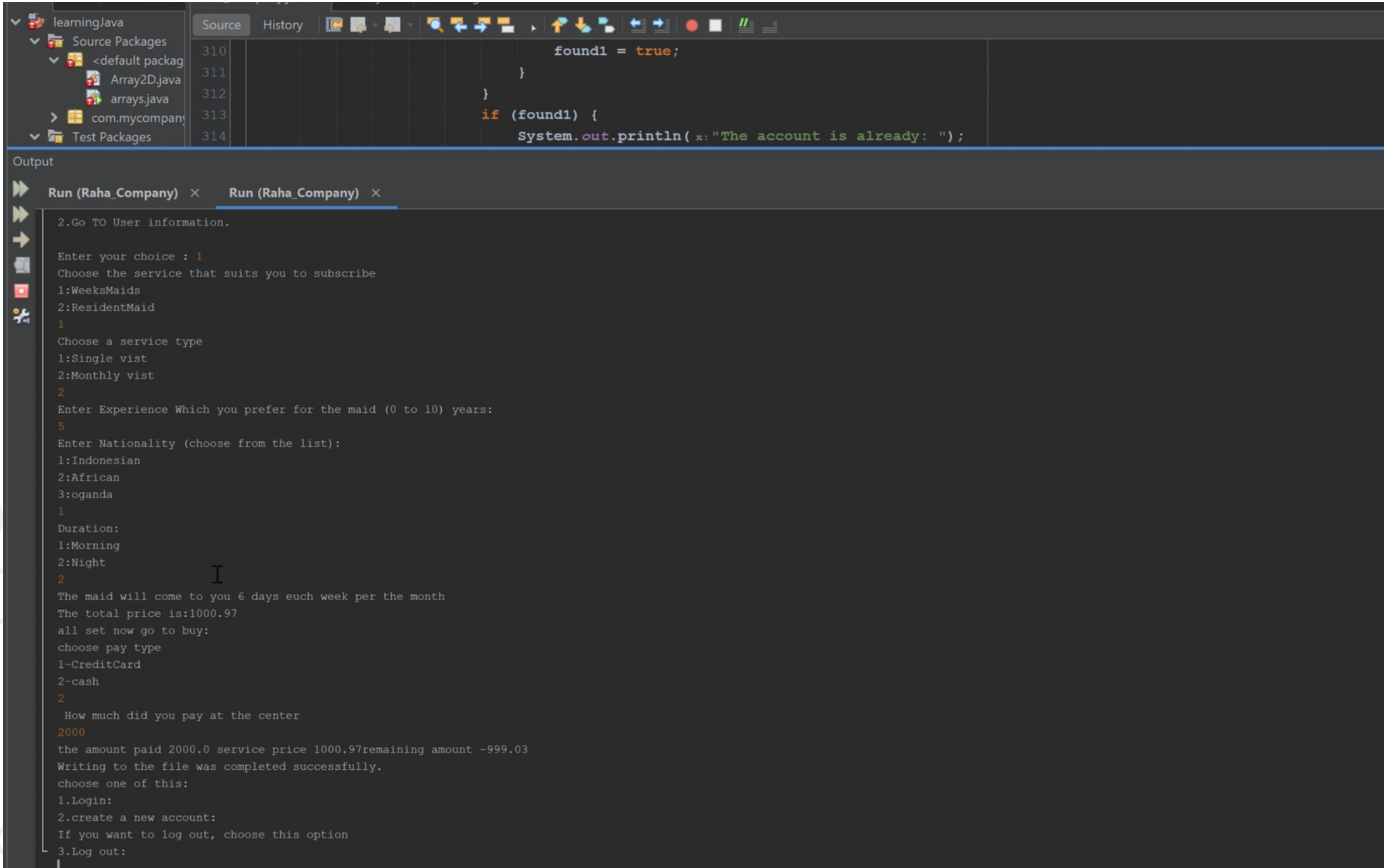
```
W   E   L   C   O   M   E
--------------------


1.GO TO Maids.
2.Go TO User information.


Enter your choice : 1
Choose the service that suits you to subscribe
1:WeeksMaids
2:ResidentMaid
1
Choose a service type
1:Single vist
2:Monthly vist
1
Enter Experience Which you prefer for the maid (0 to 10) years:
4
Enter Nationality (choose from the list):
1:Indonesian
2:African
3:oganda
1
Duration:
1:Morning
2:Night
2
```

```
sunday
monday
tusday
Wednesday
Thursday
Monday
The total price is:150.5
all set now go to buy:
choose pay type
1-CreditCard
2-cash
1


enter CreditCard Numper:7743100


enter CardHolderName:Remaskhs


cvv numper;
324
150.5 was paid from the credit card 7743100
Writing to the file was completed successfully.
choose one of this:
1.Login:
2.create a new account:
If you want to log out, choose this option
3.Log out:
```
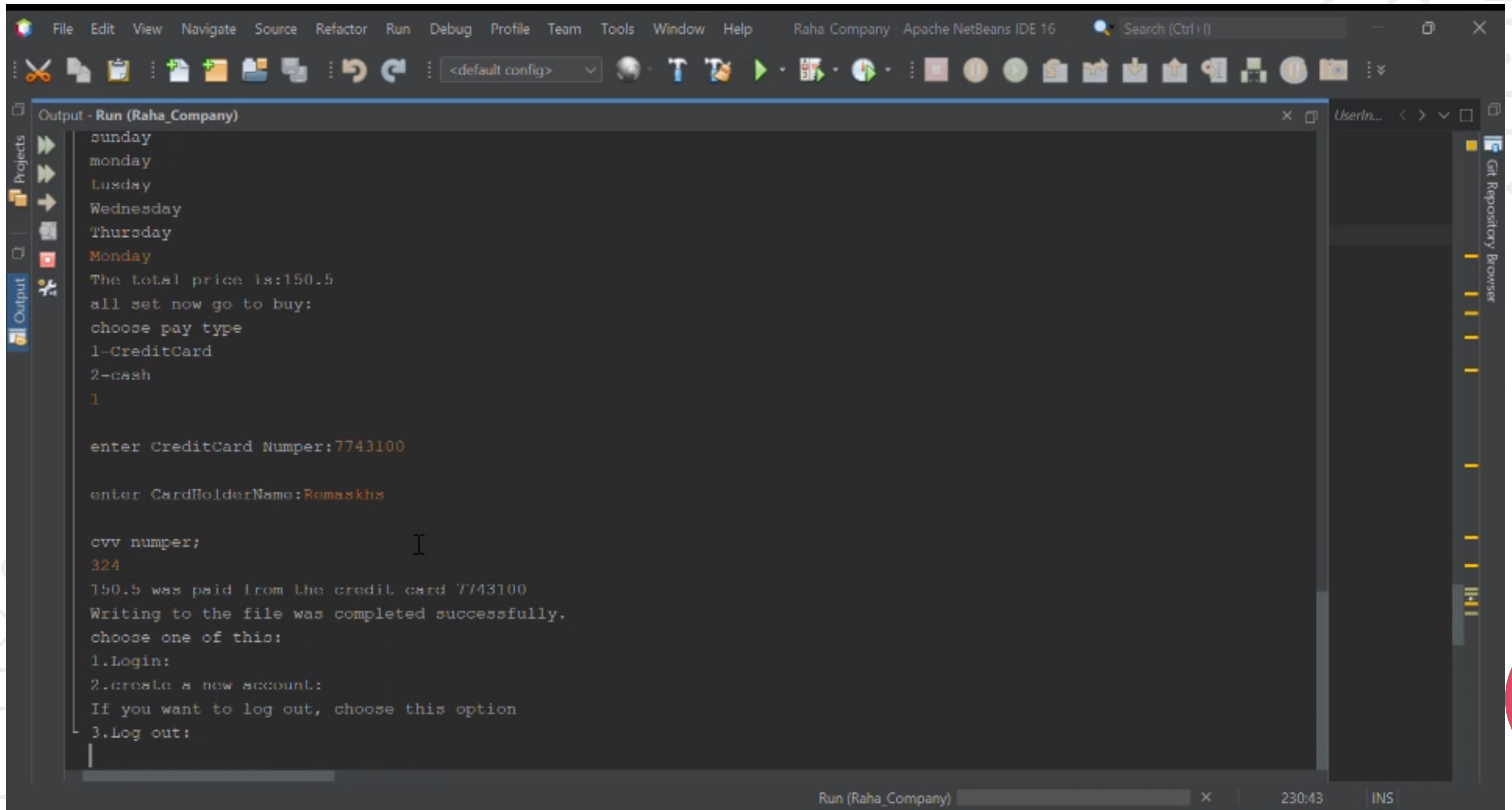
Here she chose a WeekMaid service of the type of monthly visit, with five years of experience, of Indonesian nationality, she comes at night, 6 days a week per the month. The value of the service appeared to her and she made the choice to pay in cash. She paid an amount and the rest was returned to her.
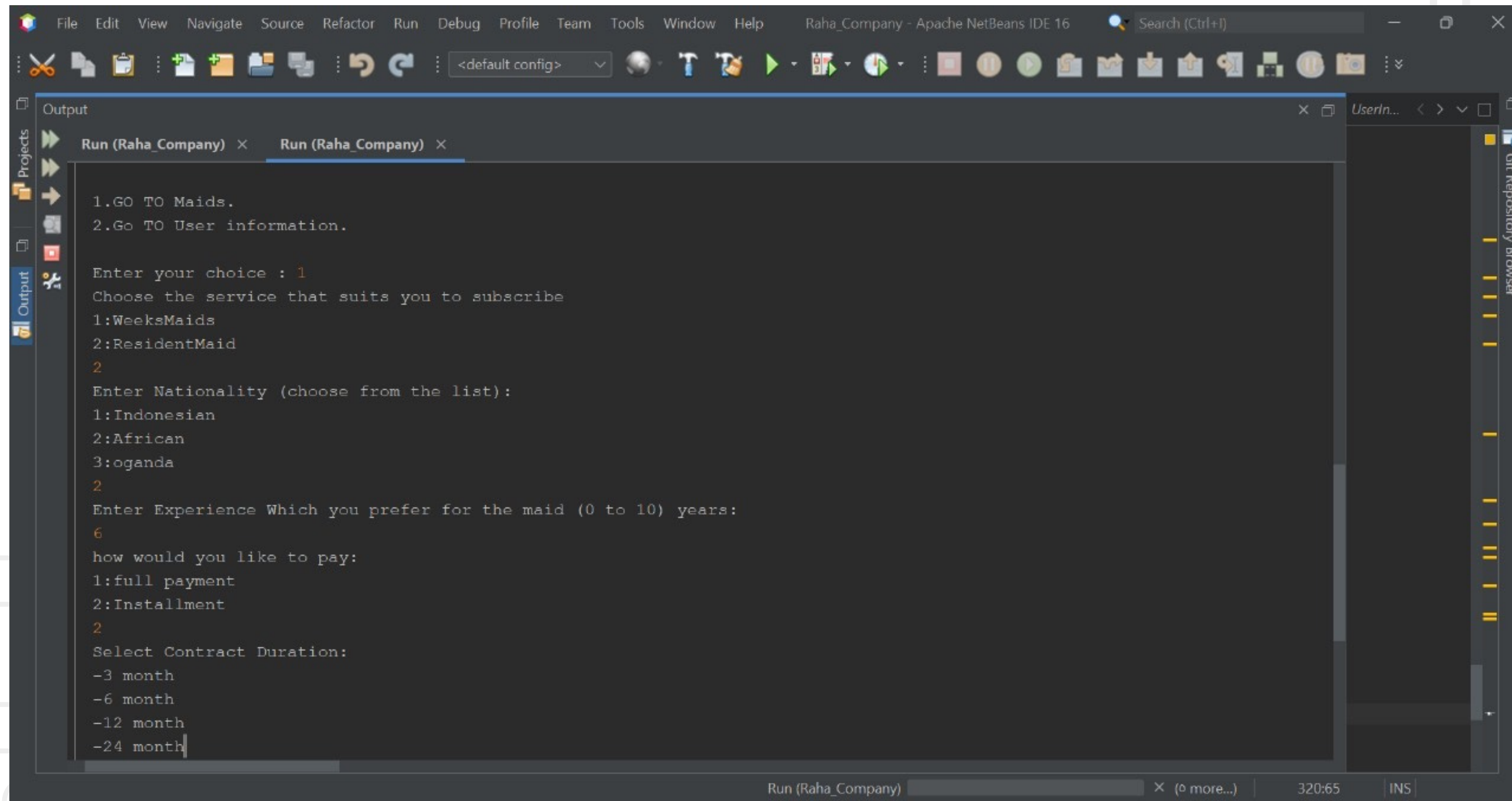

```java
                found1 = true;
            }
        }
        if (found1) {
            System.out.println( x: "The account is already: ");
```

```
Output

Run (Raha_Company) ×    Run (Raha_Company) ×

2.Go TO User information.

Enter your choice : 1
Choose the service that suits you to subscribe
1:WeeksMaids
2:ResidentMaid
1
Choose a service type
1:Single vist
2:Monthly vist
2
Enter Experience Which you prefer for the maid (0 to 10) years:
5
Enter Nationality (choose from the list):
1:Indonesian
2:African
3:oganda
1
Duration:
1:Morning
2:Night
2
The maid will come to you 6 days euch week per the month
The total price is:1000.97
all set now go to buy:
choose pay type
1-CreditCard
2-cash
2
 How much did you pay at the center
2000
the amount paid 2000.0 service price 1000.97remaining amount -999.03
Writing to the file was completed successfully.
choose one of this:
1.Login:
2.create a new account:
If you want to log out, choose this option
3.Log out:
```

Here she chose the service of a resident maid of Indonesian nationality with six years of experience. She chose payment Installment, chose the service for a period of 6 months. The value of the service appeared to her and she chose payment in cash. She paid an amount and She received a message about how much she paid and how much remains to be paid.
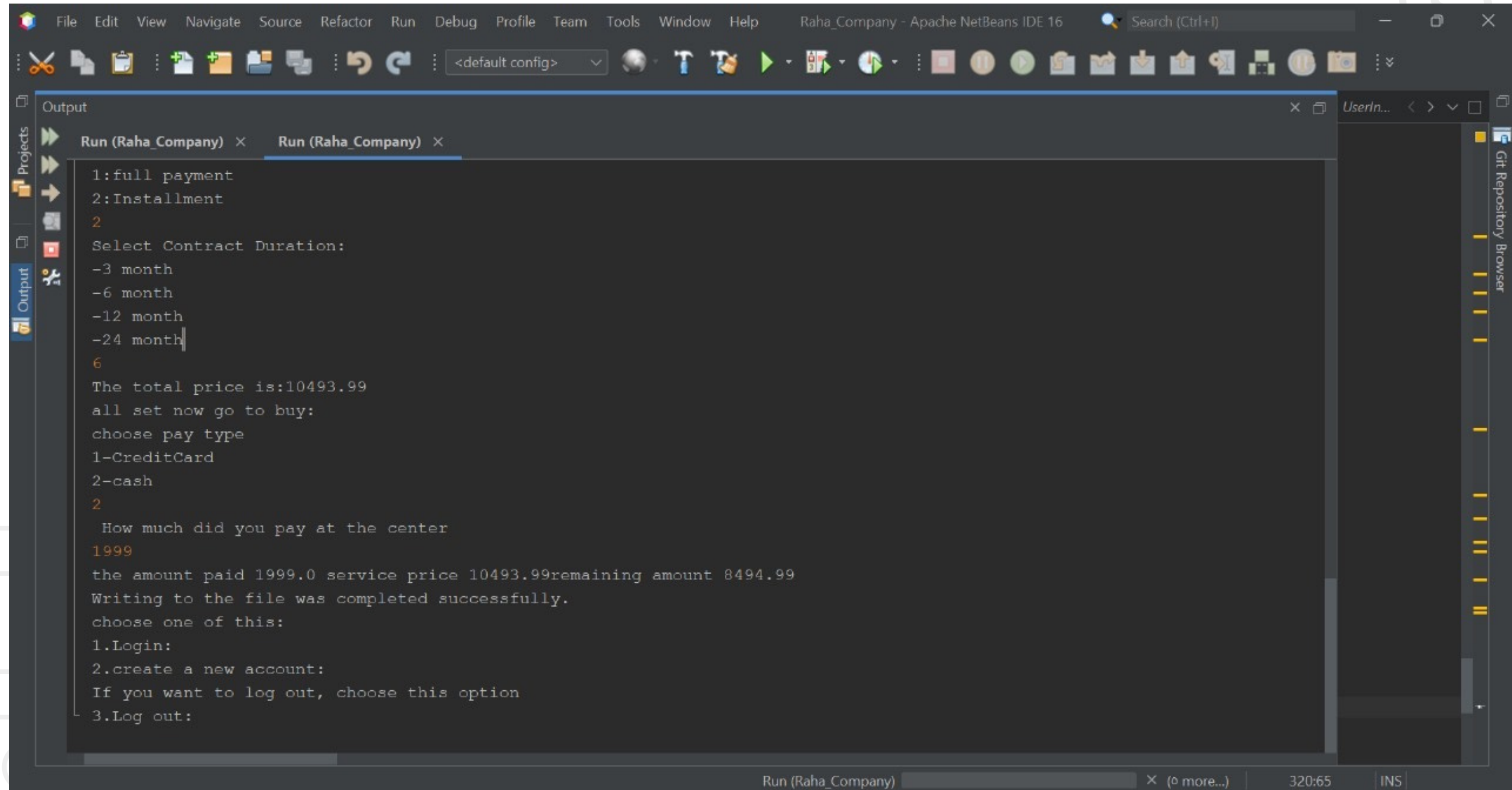
Output

Run (Raha_Company) ×   Run (Raha_Company) ×

```
1.GO TO Maids.
2.Go TO User information.

Enter your choice : 1
Choose the service that suits you to subscribe
1:WeeksMaids
2:ResidentMaid
2
Enter Nationality (choose from the list):
1:Indonesian
2:African
3:oganda
2
Enter Experience Which you prefer for the maid (0 to 10) years:
6
how would you like to pay:
1:full payment
2:Installment
2
Select Contract Duration:
-3 month
-6 month
-12 month
-24 month
```
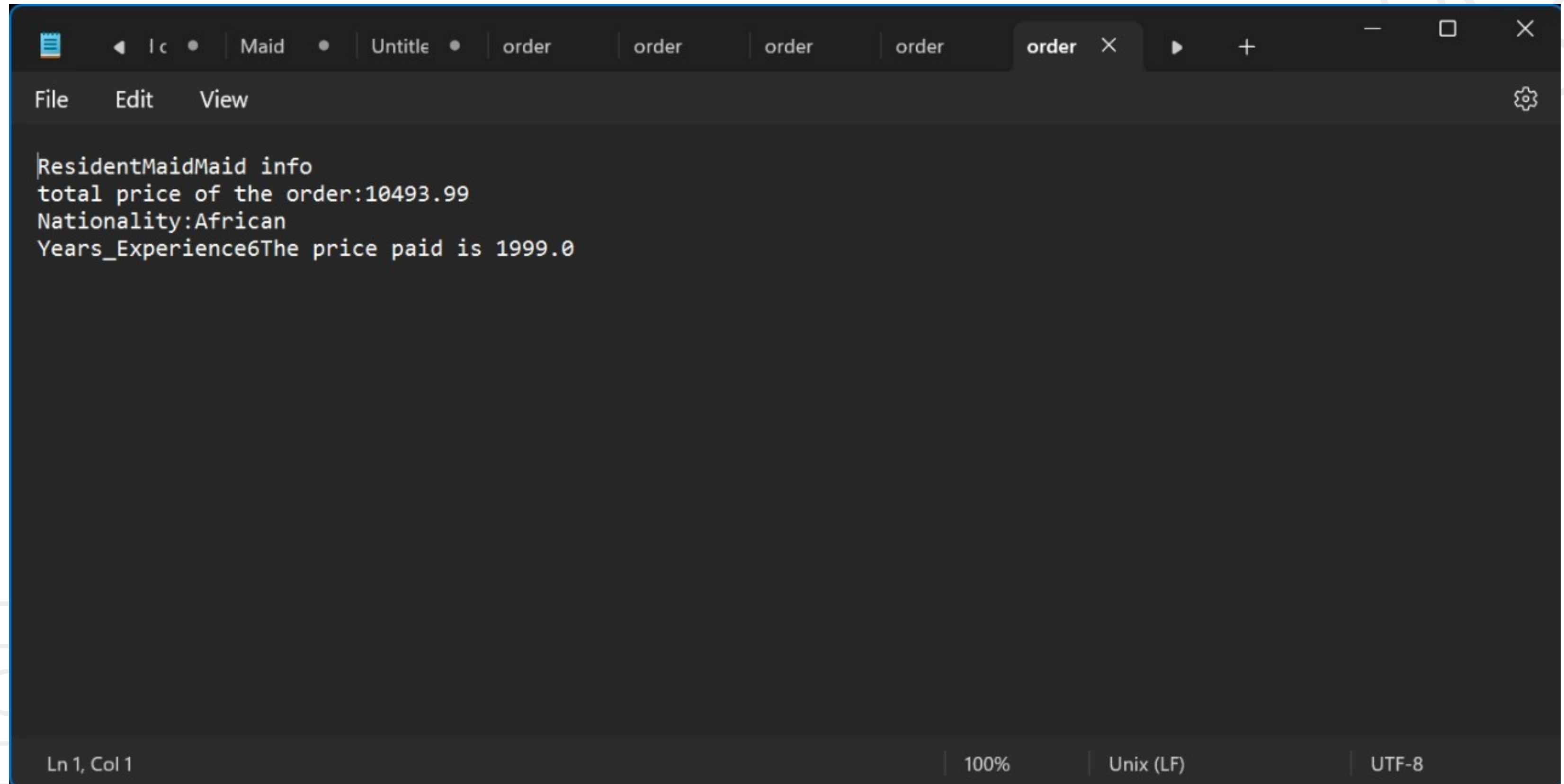
Run (Raha_Company)   × (0 more...)   320:65   INS

```
File  Edit  View  Navigate  Source  Refactor  Run  Debug  Profile  Team  Tools  Window  Help          Raha_Company - Apache NetBeans IDE 16        Search (Ctrl+I)

                                            <default config>

Output

  Run (Raha_Company)  ✕     Run (Raha_Company)  ✕

    1:full payment
    2:Installment
    2
    Select Contract Duration:
    -3 month
    -6 month
    -12 month
    -24 month
    6
    The total price is:10493.99
    all set now go to buy:
    choose pay type
    1-CreditCard
    2-cash
    2
     How much did you pay at the center
    1999
    the amount paid 1999.0 service price 10493.99remaining amount 8494.99
    Writing to the file was completed successfully.
    choose one of this:
    1.Login:
    2.create a new account:
    If you want to log out, choose this option
    3.Log out:

                                          Run (Raha_Company)                    ✕  (o more...)              320:65    INS
```

Sample picture of the file read


```
ResidentMaidMaid info
total price of the order:10493.99
Nationality:African
Years_Experience6The price paid is 1999.0
```

THANK YOU