OOP PROJECT DOCUMINTATION

Supervision: Dr. Sara Shehab

SUMMIT LOVERS TEAM:

Al-Hussein Ali
Hussein Shaban
Samar Ibrahim
Zainab Salah {T.L}
Ziad Mady

FCAI Places Project

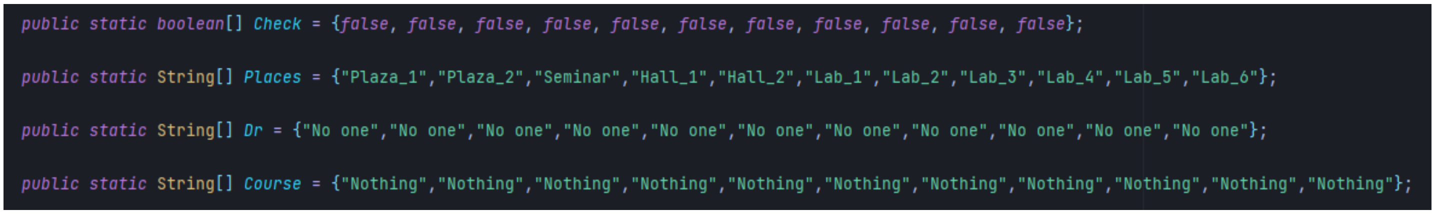
This project is a program that allows students and Drs to perform some operations on the places on campus, such as knowing the empty places at the present time, or testing whether a specific place is empty or not, and the Drs can reserve the place or cancel the reservation, and It is also possible to know the Dr in this place and the name of the subject he is teaching.

What is the aim of this Project?

The aim of this project is to help students and Drs to find an empty place to use that is suitable for their own needs.

How we did that?

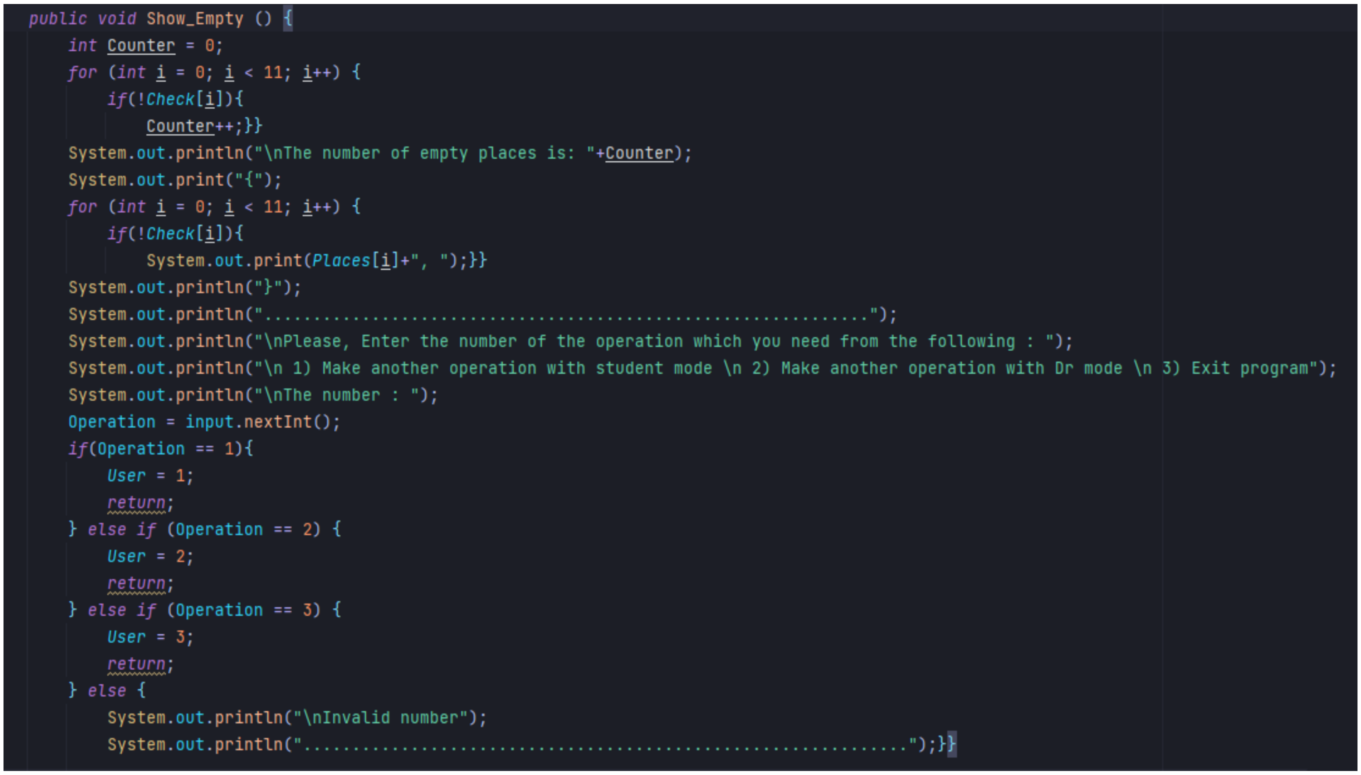
In the beginning, for the work of this program, there must be a database containing the names of the places and their status (empty or not), the names of the Drs in each place, and the names of the course, but since we were unable to link the database with the Java code, we have created 4 Arrays to store this data since it is limited As shown in the figure:



```java
public static boolean[] Check = {false, false, false, false, false, false, false, false, false, false, false};

public static String[] Places = {"Plaza_1","Plaza_2","Seminar","Hall_1","Hall_2","Lab_1","Lab_2","Lab_3","Lab_4","Lab_5","Lab_6"};

public static String[] Dr = {"No one","No one","No one","No one","No one","No one","No one","No one","No one","No one","No one"};

public static String[] Course = {"Nothing","Nothing","Nothing","Nothing","Nothing","Nothing","Nothing","Nothing","Nothing","Nothing","Nothing"};
```

After that, we wanted to finish the part for the students first and create a special class for them that includes checking a specific place or seeing the available places.

As for displaying the available places, we have created a special method to do that inside this class, and of course we made this class extend from the the class that contains the 4 arrays called place class to freely use these arrays as shown in the figure:

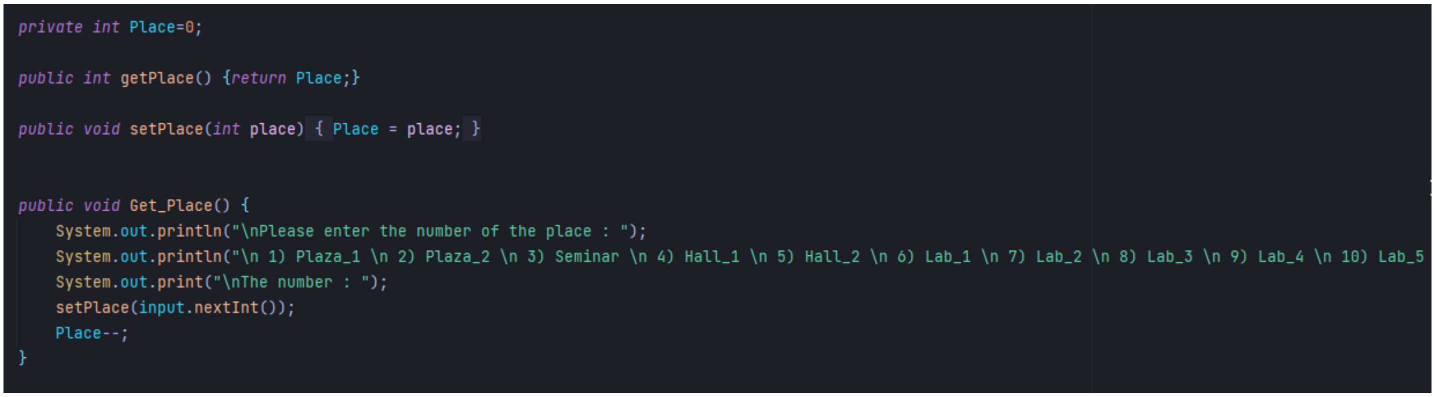


```java
public void Show_Empty () {
    int Counter = 0;
    for (int i = 0; i < 11; i++) {
        if(!Check[i]){
            Counter++;}}
    System.out.println("\nThe number of empty places is: "+Counter);
    System.out.print("{");
    for (int i = 0; i < 11; i++) {
        if(!Check[i]){
            System.out.print(Places[i]+", ");}}
    System.out.println("}");
    System.out.println(".............................................................");
    System.out.println("\nPlease, Enter the number of the operation which you need from the following : ");
    System.out.println("\n 1) Make another operation with student mode \n 2) Make another operation with Dr mode \n 3) Exit program");
    System.out.println("\nThe number : ");
    Operation = input.nextInt();
    if(Operation == 1){
        User = 1;
        return;
    } else if (Operation == 2) {
        User = 2;
        return;
    } else if (Operation == 3) {
        User = 3;
        return;
    } else {
        System.out.println("\nInvalid number");
        System.out.println("..............................................................");}}
```

In this method, we create a variable called Counter to calculate the number of empty spaces and print it.

And 2 for loops were used, one to calculate the number and the second to print the empty places that were counted. Of course, it was possible to make them in only one, but this is how the number would be printed after displaying the places and not before, so we counted them only at the beginning, and then we printed them.

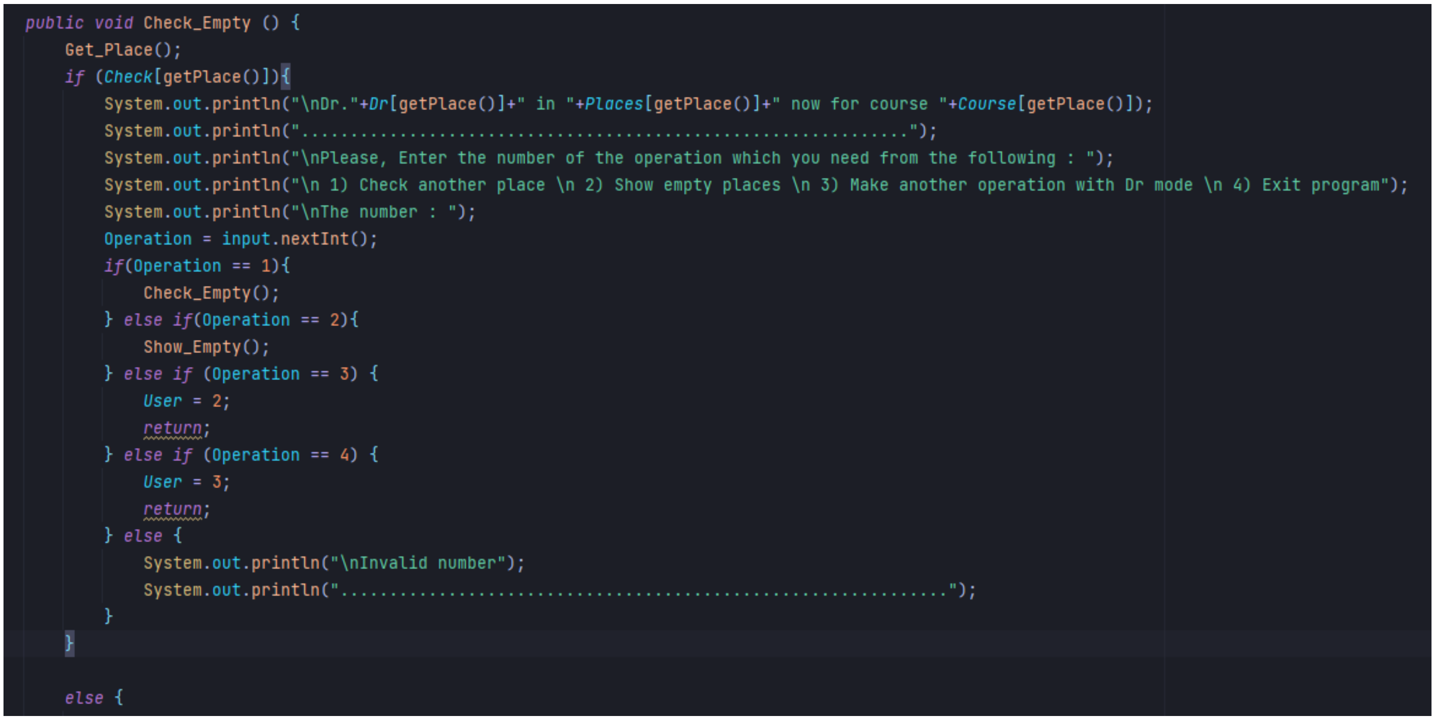
As for the second method that students can use, which is to check a specific place, we have created another method in the Student class to perform this method, but to check a place we must first know the place that the user wants to check, so we first created another method to find out the place that the user wants, but we put this method In the Place class because it is not specific to students only, as in the figure:



```
private int Place=0;

public int getPlace() {return Place;}

public void setPlace(int place) { Place = place; }


public void Get_Place() {
    System.out.println("\nPlease enter the number of the place : ");
    System.out.println("\n 1) Plaza_1 \n 2) Plaza_2 \n 3) Seminar \n 4) Hall_1 \n 5) Hall_2 \n 6) Lab_1 \n 7) Lab_2 \n 8) Lab_3 \n 9) Lab_4 \n 10) Lab_5
    System.out.print("\nThe number : ");
    setPlace(input.nextInt());
    Place--;
}
```

This method displays all the places on the university campus, and the user enters the number of the place he wants, and it is saved in a variable named Place, but the number is reduced by 1 because it will be dealt with in arrays that start counting from the number 0 and not 1.

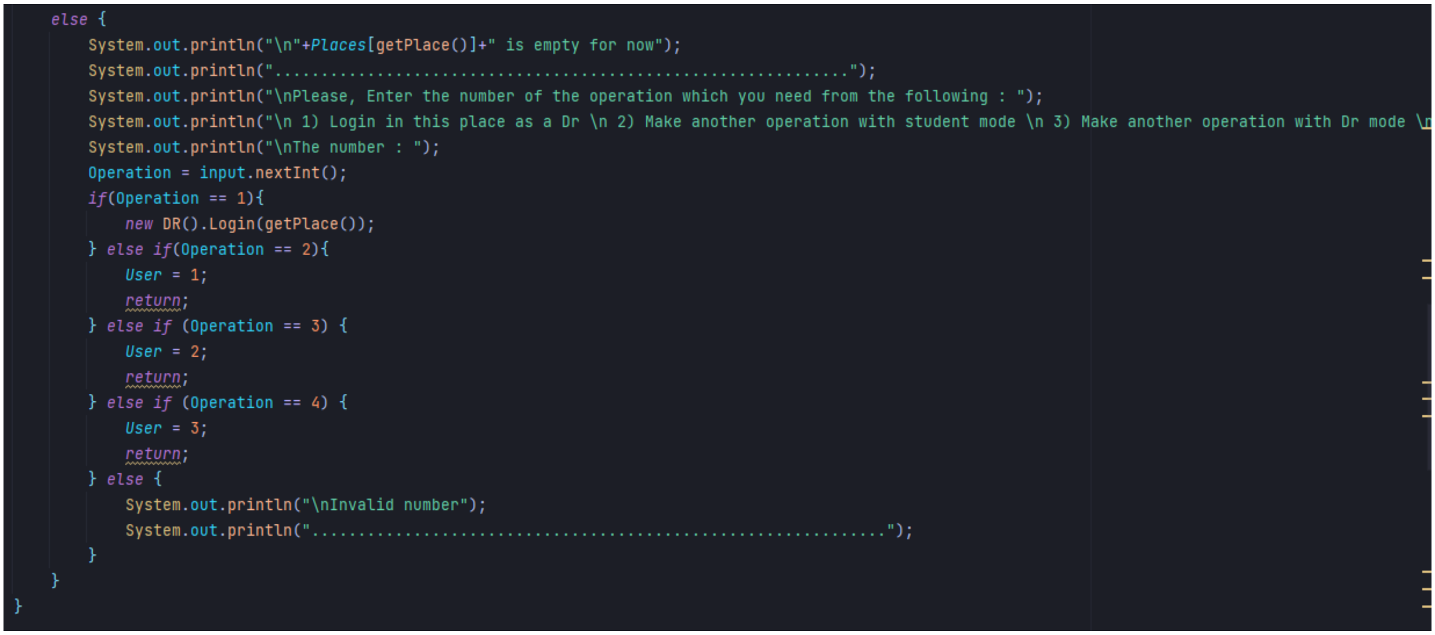
Now, by referring to the Check_Empty method in the Student class, first the Get_Place method is called from the Place class, and then the checking process is done by the Check array as shown:



```
public void Check_Empty () {
    Get_Place();
    if (Check[getPlace()]){
        System.out.println("\nDr."+Dr[getPlace()]+" in "+Places[getPlace()]+" now for course "+Course[getPlace()]);
        System.out.println("...............................................................");
        System.out.println("\nPlease, Enter the number of the operation which you need from the following : ");
        System.out.println("\n 1) Check another place \n 2) Show empty places \n 3) Make another operation with Dr mode \n 4) Exit program");
        System.out.println("\nThe number : ");
        Operation = input.nextInt();
        if(Operation == 1){
            Check_Empty();
        } else if(Operation == 2){
            Show_Empty();
        } else if (Operation == 3) {
            User = 2;
            return;
        } else if (Operation == 4) {
            User = 3;
            return;
        } else {
            System.out.println("\nInvalid number");
            System.out.println("...............................................................");
        }
    }

    else {
```

In this part, using if and Check array, if the value inside the array is true, this means that the place is currently occupied for your reason. A sentence will be printed containing that the place is reserved by this Dr who studies this course, and this is by outputting the values corresponding to this place in other arrays and it is given the user has other options, including checking another place and showing the currently available places.

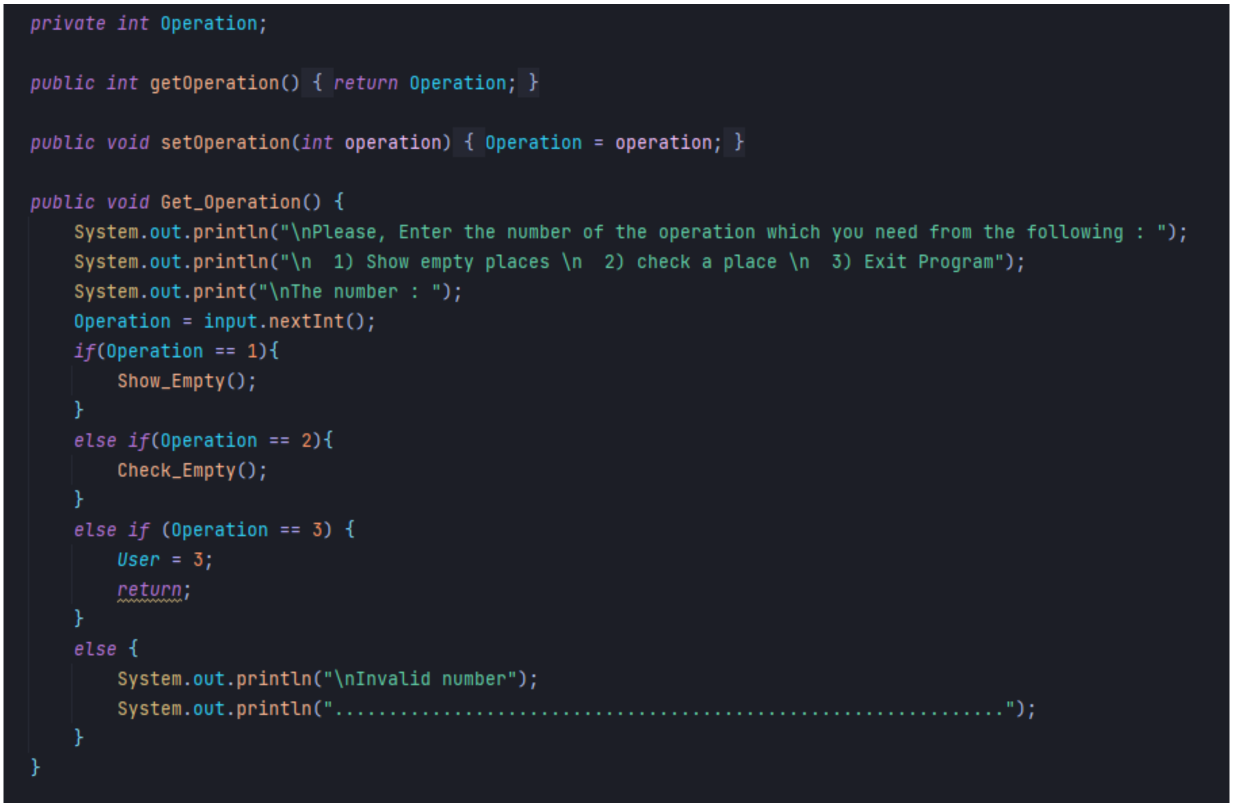
On the other hand, if the value inside the Check array is false, this means that the place is empty, in this case, a sentence is printed stating that this place is empty, and some options are given to the user, including reserving the place as a Dr, as shown in the figure:



```java
    else {
        System.out.println("\n"+Places[getPlace()]+" is empty for now");
        System.out.println("..............................................................");
        System.out.println("\nPlease, Enter the number of the operation which you need from the following : ");
        System.out.println("\n 1) Login in this place as a Dr \n 2) Make another operation with student mode \n 3) Make another operation with Dr mode \n
        System.out.println("\nThe number : ");
        Operation = input.nextInt();
        if(Operation == 1){
            new DR().Login(getPlace());
        } else if(Operation == 2){
            User = 1;
            return;
        } else if (Operation == 3) {
            User = 2;
            return;
        } else if (Operation == 4) {
            User = 3;
            return;
        } else {
            System.out.println("\nInvalid number");
            System.out.println("..............................................................");
        }
    }
}
```

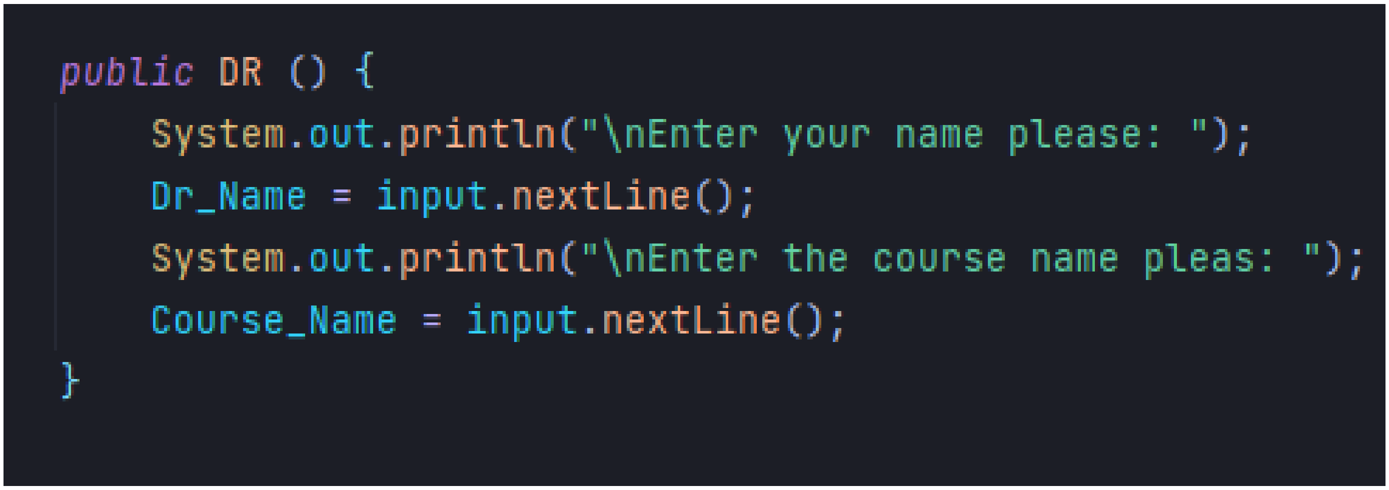
So how do we know what the student wants to do, whether it is showing available places or examining a specific place?

Therefore, we created a function to find out the operation that the student wants to do. In this function, the student chooses the operation that he wants to perform, or if he wants to exit the program by saving the operation number in a variable named Operation, as in the form:



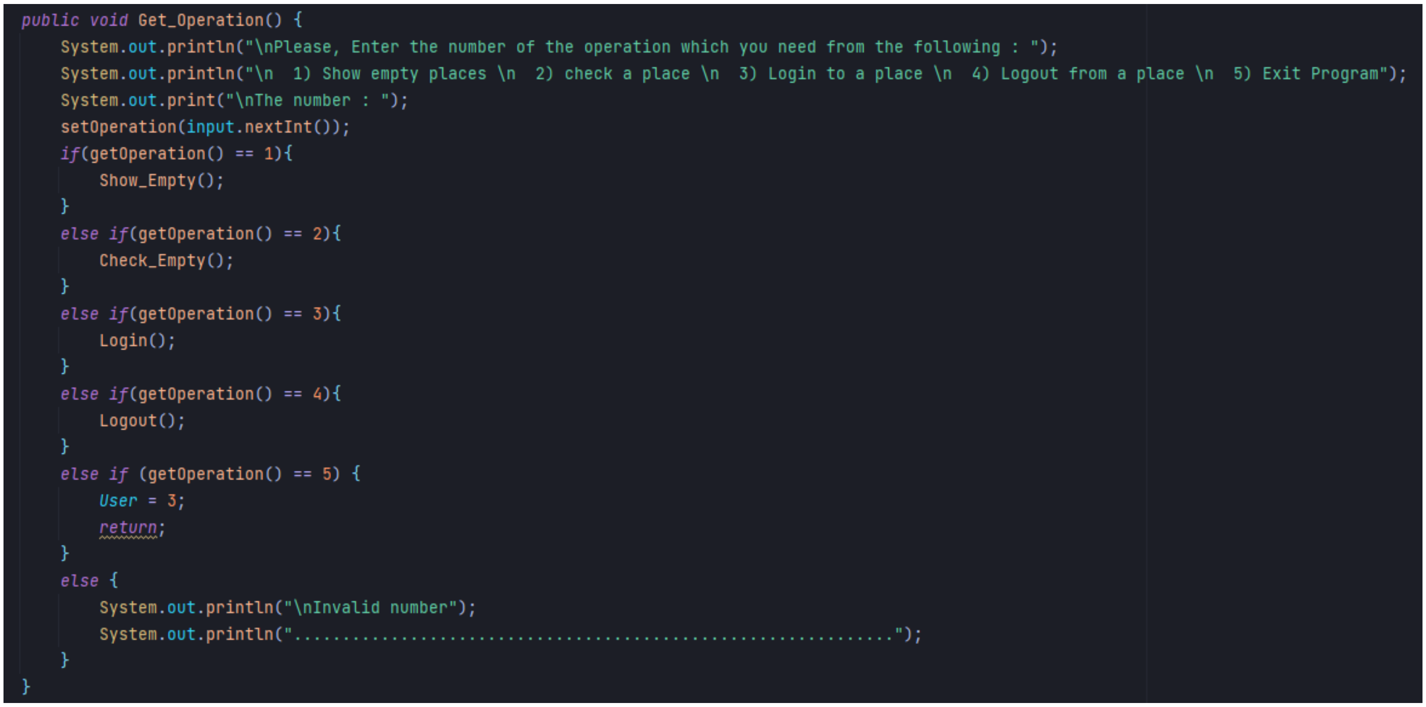
```java
private int Operation;

public int getOperation() { return Operation; }

public void setOperation(int operation) { Operation = operation; }

public void Get_Operation() {
    System.out.println("\nPlease, Enter the number of the operation which you need from the following : ");
    System.out.println("\n  1) Show empty places \n  2) check a place \n  3) Exit Program");
    System.out.print("\nThe number : ");
    Operation = input.nextInt();
    if(Operation == 1){
        Show_Empty();
    }
    else if(Operation == 2){
        Check_Empty();
    }
    else if (Operation == 3) {
        User = 3;
        return;
    }
    else {
        System.out.println("\nInvalid number");
        System.out.println("..............................................................");
    }
}
```

Thus, the two methods that the student will need have been made, and since the Dr will also have these methods in addition to other methods, we have created a class with the name DR, which is extend from the Student class, and a constructor has been added to it to enter the name of the Dr and the name of the course he teaches, as in the form:



```java
public DR () {
    System.out.println("\nEnter your name please: ");
    Dr_Name = input.nextLine();
    System.out.println("\nEnter the course name pleas: ");
    Course_Name = input.nextLine();
}
```

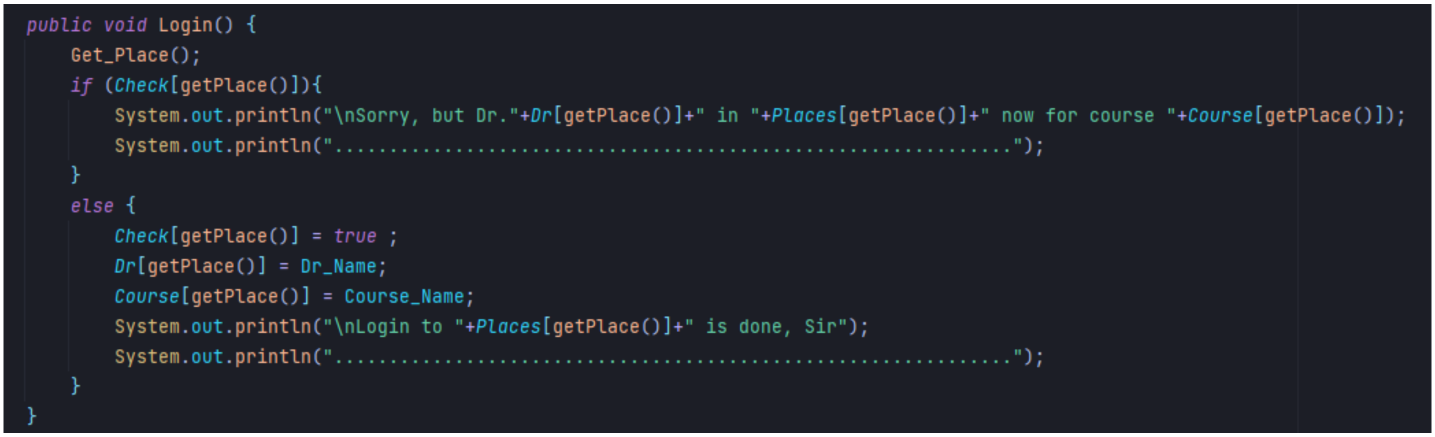
Since the operations that the Dr will perform are different from those performed by the student, we have override the Get_Operation method in the Student class and added the reservation and cancel reservation operations for the operations that can be selected as shown in the figure:



```java
public void Get_Operation() {
    System.out.println("\nPlease, Enter the number of the operation which you need from the following : ");
    System.out.println("\n  1) Show empty places \n  2) check a place \n  3) Login to a place \n  4) Logout from a place \n  5) Exit Program");
    System.out.print("\nThe number : ");
    setOperation(input.nextInt());
    if(getOperation() == 1){
        Show_Empty();
    }
    else if(getOperation() == 2){
        Check_Empty();
    }
    else if(getOperation() == 3){
        Login();
    }
    else if(getOperation() == 4){
        Logout();
    }
    else if (getOperation() == 5) {
        User = 3;
        return;
    }
    else {
        System.out.println("\nInvalid number");
        System.out.println("..............................................................");
    }
}
```

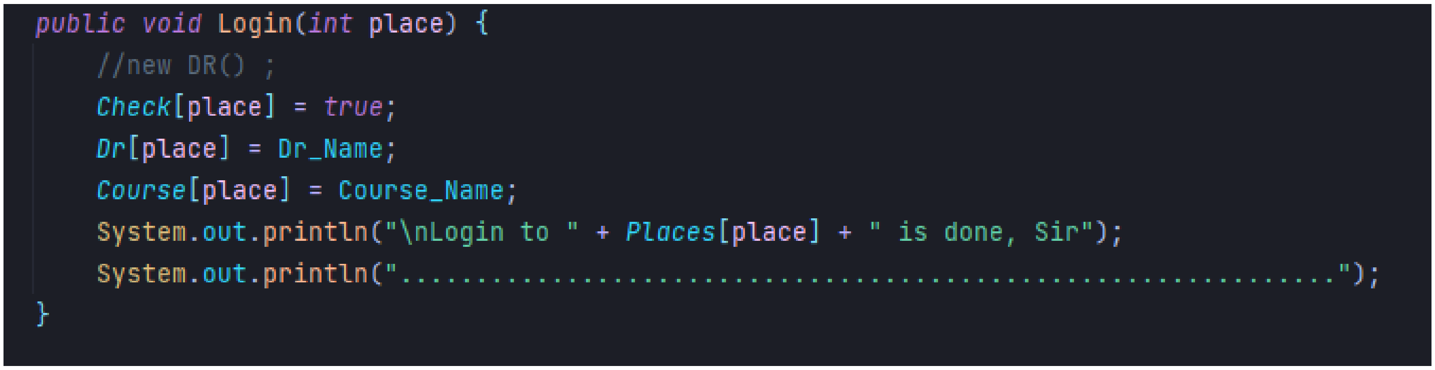
Now, as for booking the place, first we need to specify a place, and after selecting it, we need to check if it is empty or not.

If the hall is full, a sentence will be printed, and if it is empty, it will be made full and a message stating that will be printed, as shown in the figure:



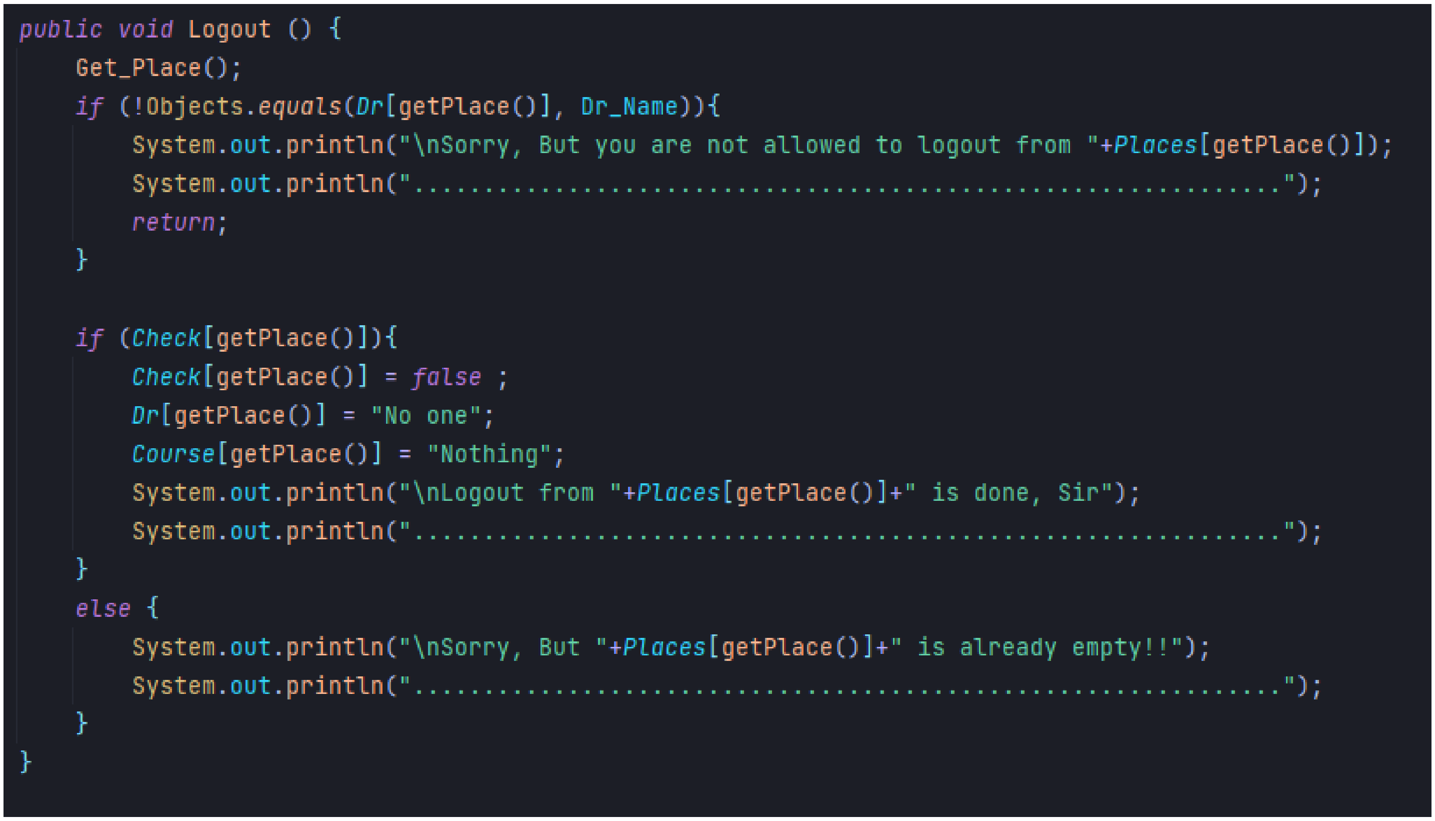
```
public void Login() {
    Get_Place();
    if (Check[getPlace()]){
        System.out.println("\nSorry, but Dr."+Dr[getPlace()]+" in "+Places[getPlace()]+" now for course "+Course[getPlace()]);
        System.out.println("..............................................................");
    }
    else {
        Check[getPlace()] = true ;
        Dr[getPlace()] = Dr_Name;
        Course[getPlace()] = Course_Name;
        System.out.println("\nLogin to "+Places[getPlace()]+" is done, Sir");
        System.out.println("..............................................................");
    }
}
```

This is good, but since we put the reservation option in the Check method, if the place is empty, then it does not need to specify the location again, so we overloaded the Login method, where in the second method it takes a parameter of type int and it expresses the place that the method needs and takes it from a method Check and performs the rest of the operations as the previous method, as shown in the figure:



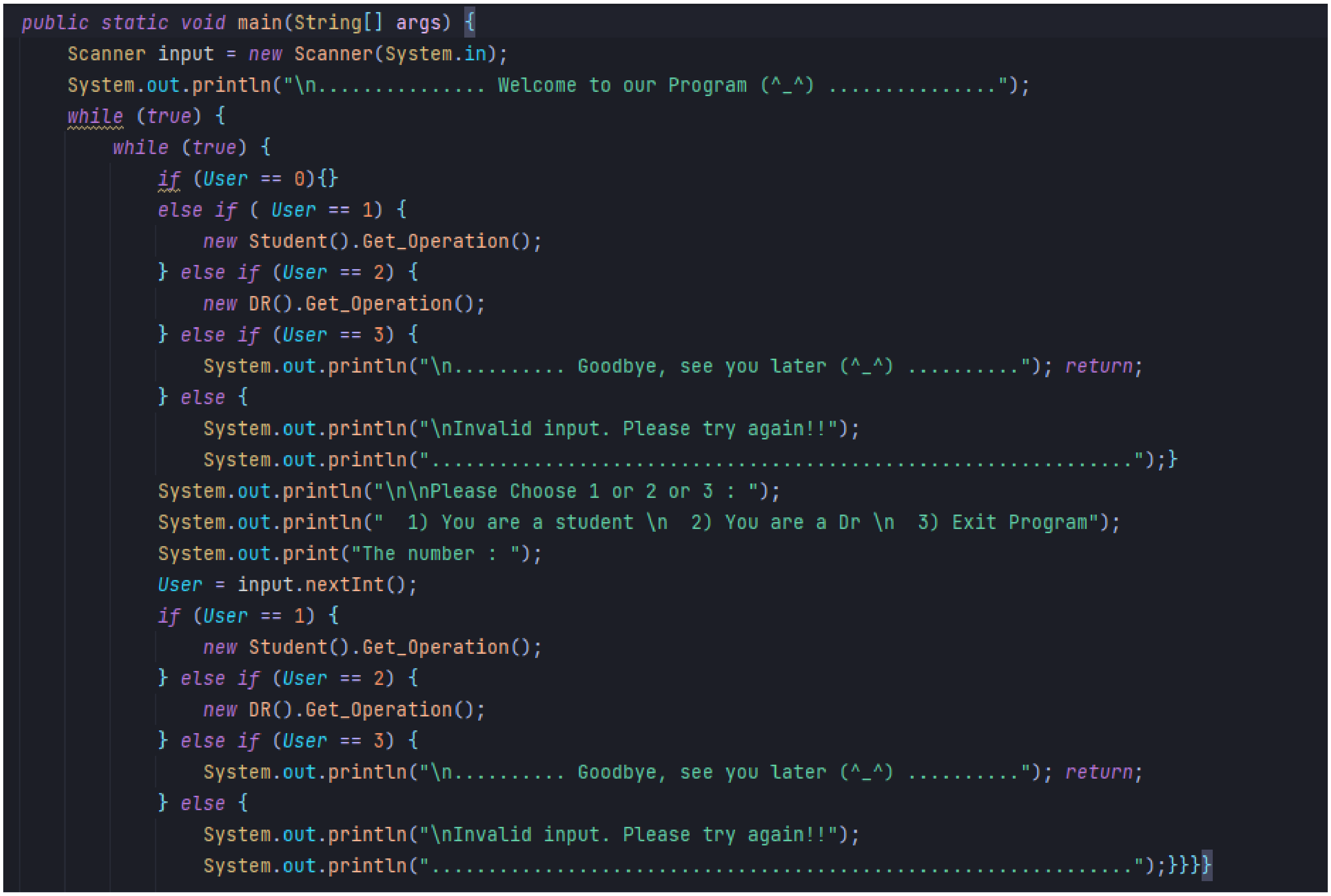
```
public void Login(int place) {
    //new DR() ;
    Check[place] = true;
    Dr[place] = Dr_Name;
    Course[place] = Course_Name;
    System.out.println("\nLogin to " + Places[place] + " is done, Sir");
    System.out.println("..............................................................");
}
```

Then, for canceling the reservation, we made a Logout method, which also needs to specify the place and check if this was the place empty, then a message will be printed to inform the user that this place is already empty, and if the place is full, then it must be confirmed that the person who cancels the reservation is the same one who logged in, and this will be done by comparing the name of the Dr who is trying to cancel the reservation with the name of the Dr who is currently in the place whose data has been saved in the Dr array in the Place class, and if it is confirmed that it is the same Dr, the reservation will be canceled. Otherwise, a message will be printed stating that this user is not allowed to cancel This reservation is as shown in the figure:



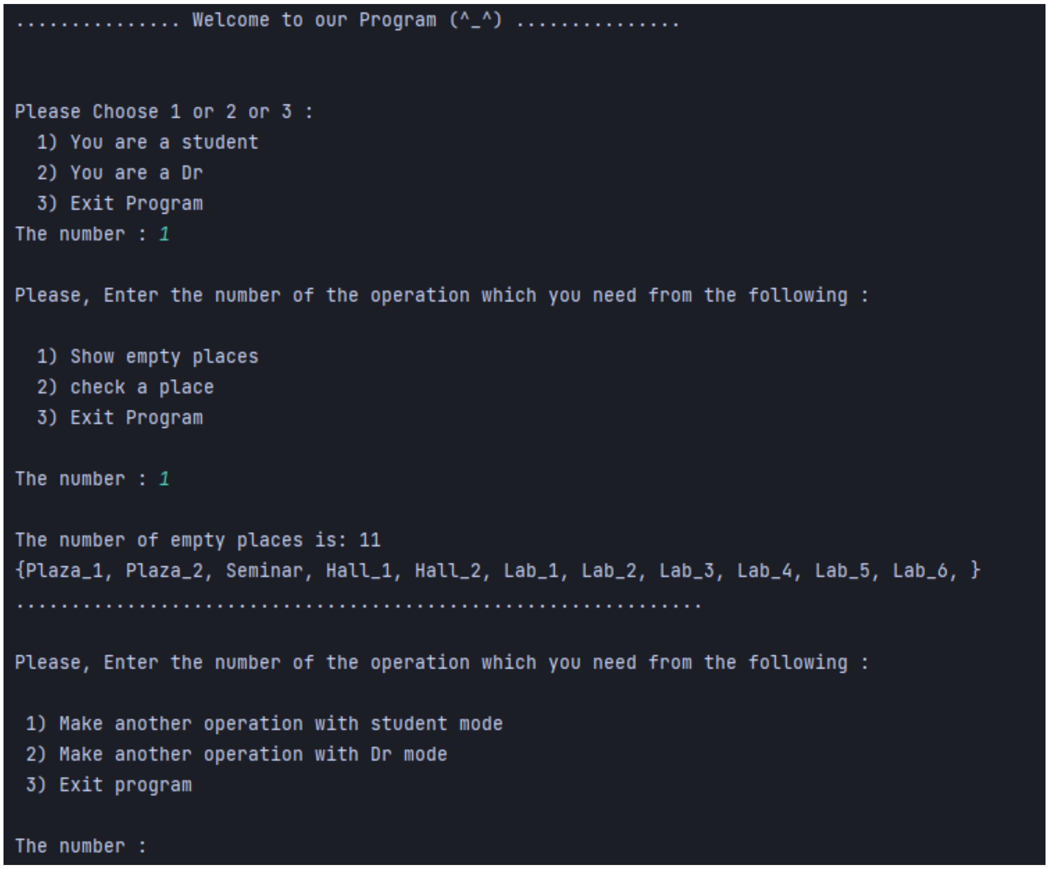
```
public void Logout () {
    Get_Place();
    if (!Objects.equals(Dr[getPlace()], Dr_Name)){
        System.out.println("\nSorry, But you are not allowed to logout from "+Places[getPlace()]);
        System.out.println("..............................................................");
        return;
    }

    if (Check[getPlace()]){
        Check[getPlace()] = false ;
        Dr[getPlace()] = "No one";
        Course[getPlace()] = "Nothing";
        System.out.println("\nLogout from "+Places[getPlace()]+" is done, Sir");
        System.out.println("..............................................................");
    }
    else {
        System.out.println("\nSorry, But "+Places[getPlace()]+" is already empty!!");
        System.out.println("..............................................................");
    }
}
```

Now that the required methods have been compiled to perform the specified methods, a start must be made for the program, which was made in the main method in Place class, where it defines the user, whether he is a student or a Dr, and saves it in a variable named User, and the possible cases for this variable are placed in the beginning, because if the user wants to exit from the middle of the program, he cannot exit directly, as in the Python language, which contains a ready command called “exit()”, but he can exit the current method and return to the main method, so this condition was added at the beginning, because if he wanted to exit from it in the middle the program will be converted to this method after changing the value of the User variable to 3 and extracting it from here in this way, as in the figure:

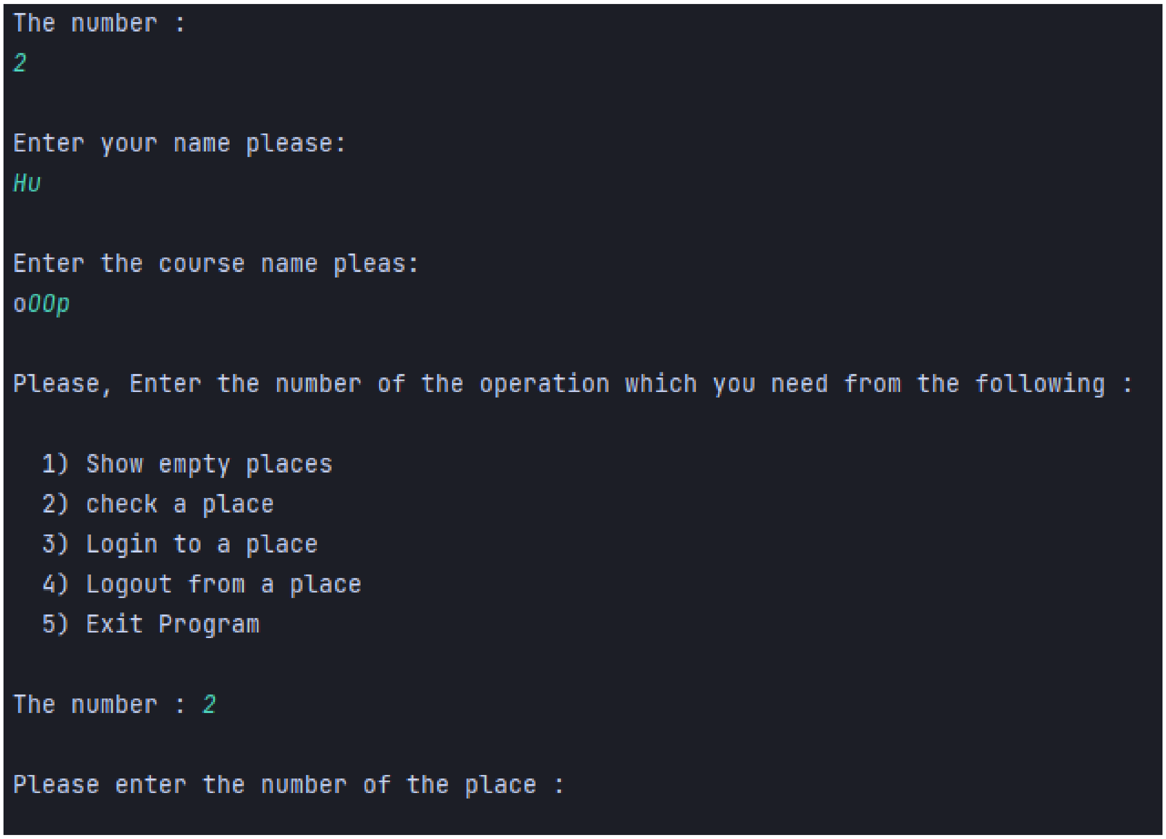


```java
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.println("\n............... Welcome to our Program (^_^) ...............");
    while (true) {
        while (true) {
            if (User == 0){}
            else if ( User == 1) {
                new Student().Get_Operation();
            } else if (User == 2) {
                new DR().Get_Operation();
            } else if (User == 3) {
                System.out.println("\n.......... Goodbye, see you later (^_^) .........."); return;
            } else {
                System.out.println("\nInvalid input. Please try again!!");
                System.out.println("..............................................................");}
            System.out.println("\n\nPlease Choose 1 or 2 or 3 : ");
            System.out.println("  1) You are a student \n  2) You are a Dr \n  3) Exit Program");
            System.out.print("The number : ");
            User = input.nextInt();
            if (User == 1) {
                new Student().Get_Operation();
            } else if (User == 2) {
                new DR().Get_Operation();
            } else if (User == 3) {
                System.out.println("\n.......... Goodbye, see you later (^_^) .........."); return;
            } else {
                System.out.println("\nInvalid input. Please try again!!");
                System.out.println("..............................................................");}}}}
```

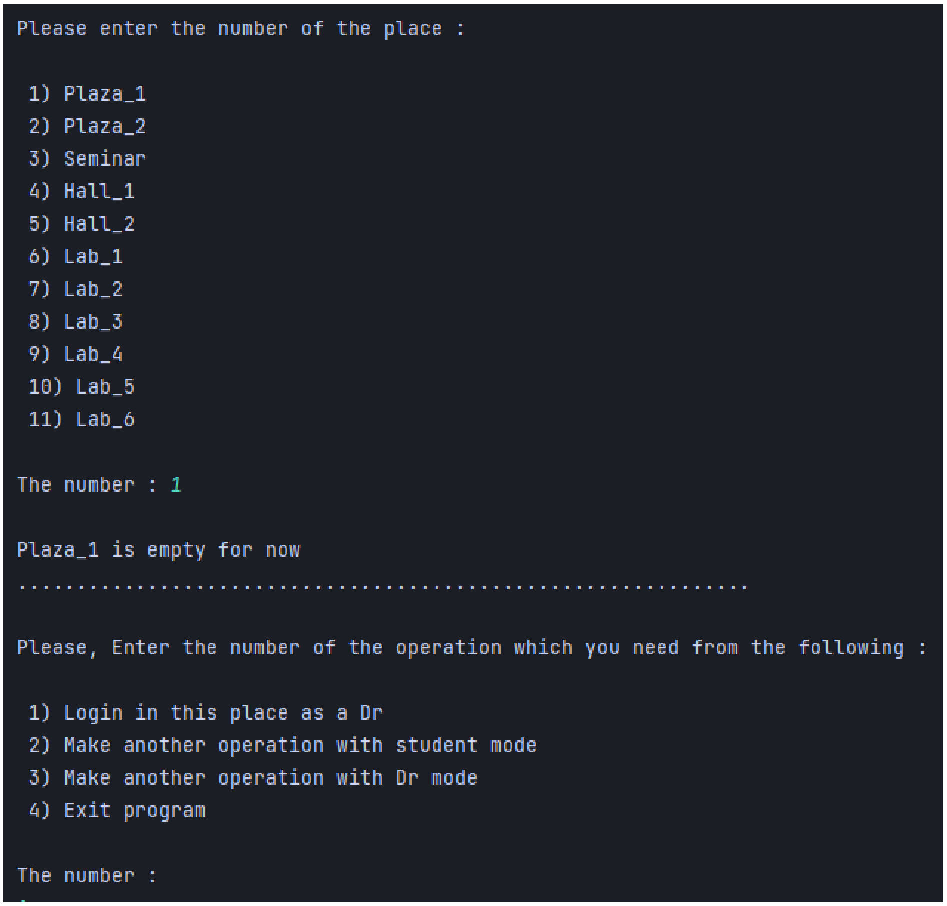
Now these are some pictures that show the work of the program and the output of some of its important parts:



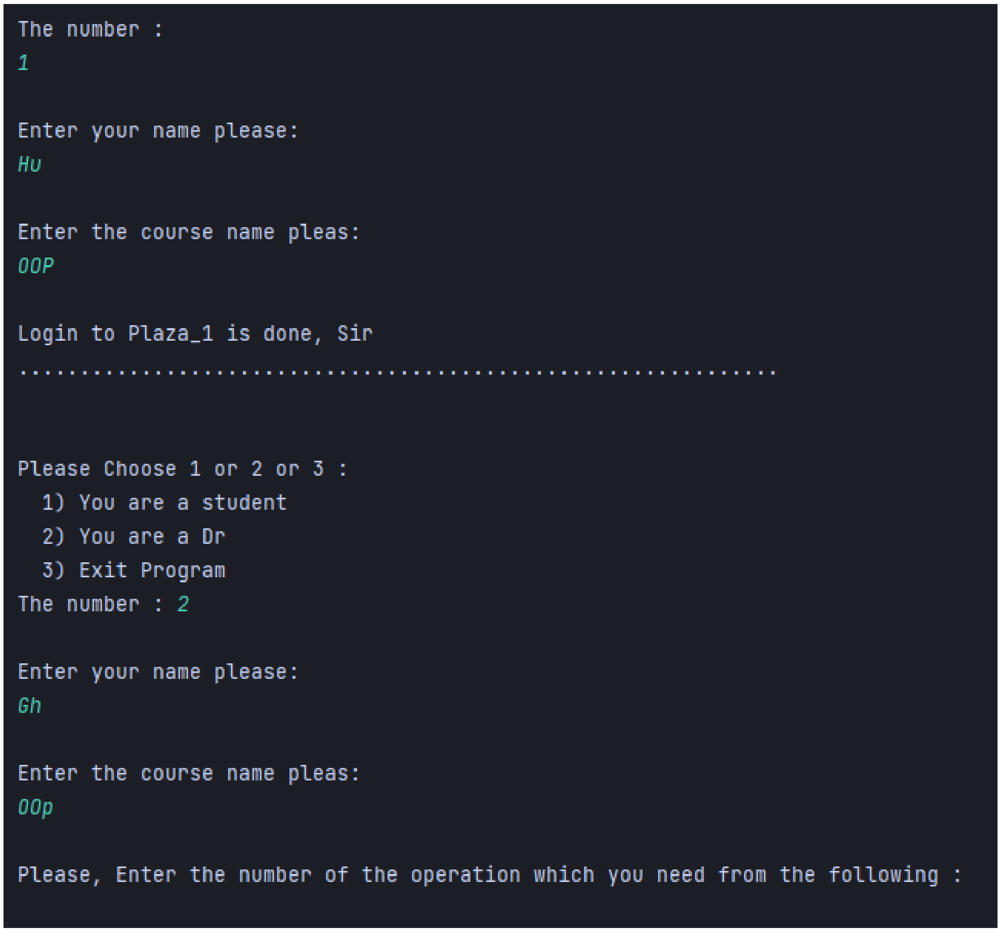
```
............... Welcome to our Program (^_^) ..............



Please Choose 1 or 2 or 3 :
  1) You are a student
  2) You are a Dr
  3) Exit Program
The number : 1

Please, Enter the number of the operation which you need from the following :

  1) Show empty places
  2) check a place
  3) Exit Program

The number : 1

The number of empty places is: 11
{Plaza_1, Plaza_2, Seminar, Hall_1, Hall_2, Lab_1, Lab_2, Lab_3, Lab_4, Lab_5, Lab_6, }
..............................................................

Please, Enter the number of the operation which you need from the following :

 1) Make another operation with student mode
 2) Make another operation with Dr mode
 3) Exit program

The number :
```



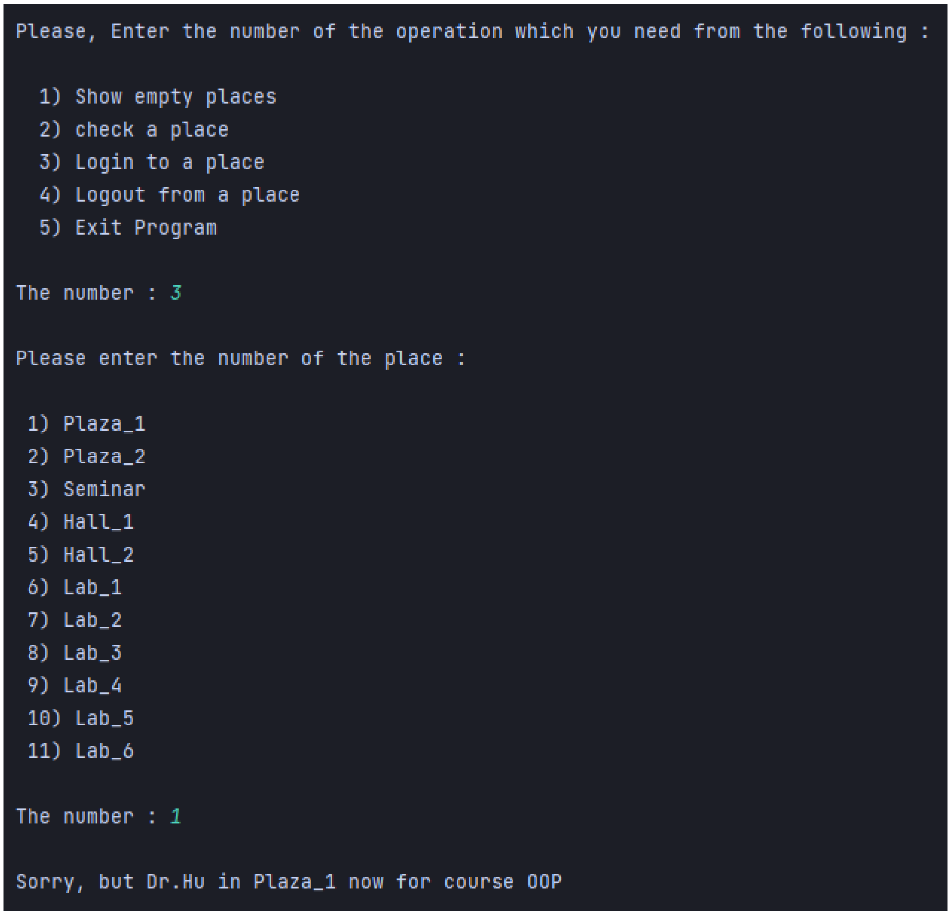
```
The number :
2

Enter your name please:
Hu

Enter the course name pleas:
oOOp

Please, Enter the number of the operation which you need from the following :

  1) Show empty places
  2) check a place
  3) Login to a place
  4) Logout from a place
  5) Exit Program

The number : 2

Please enter the number of the place :
```



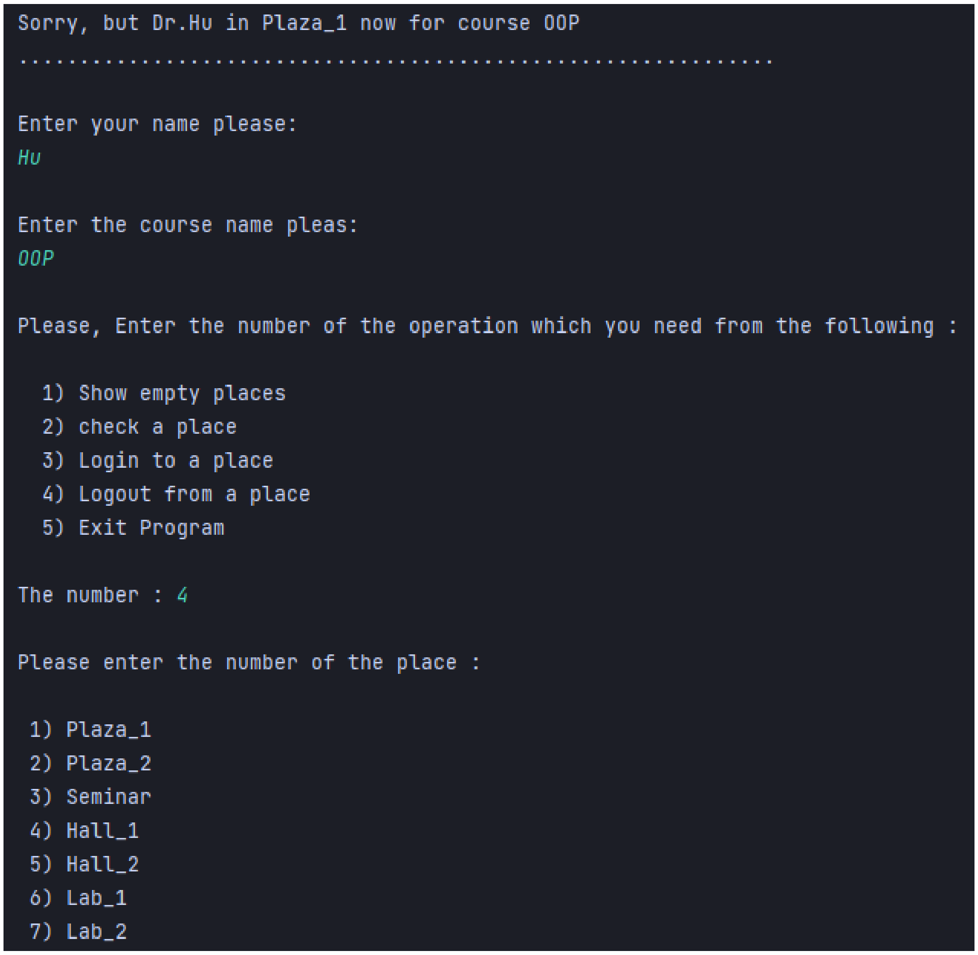
```
Please enter the number of the place :

 1) Plaza_1
 2) Plaza_2
 3) Seminar
 4) Hall_1
 5) Hall_2
 6) Lab_1
 7) Lab_2
 8) Lab_3
 9) Lab_4
 10) Lab_5
 11) Lab_6

The number : 1

Plaza_1 is empty for now
..............................................................

Please, Enter the number of the operation which you need from the following :

 1) Login in this place as a Dr
 2) Make another operation with student mode
 3) Make another operation with Dr mode
 4) Exit program

The number :
```



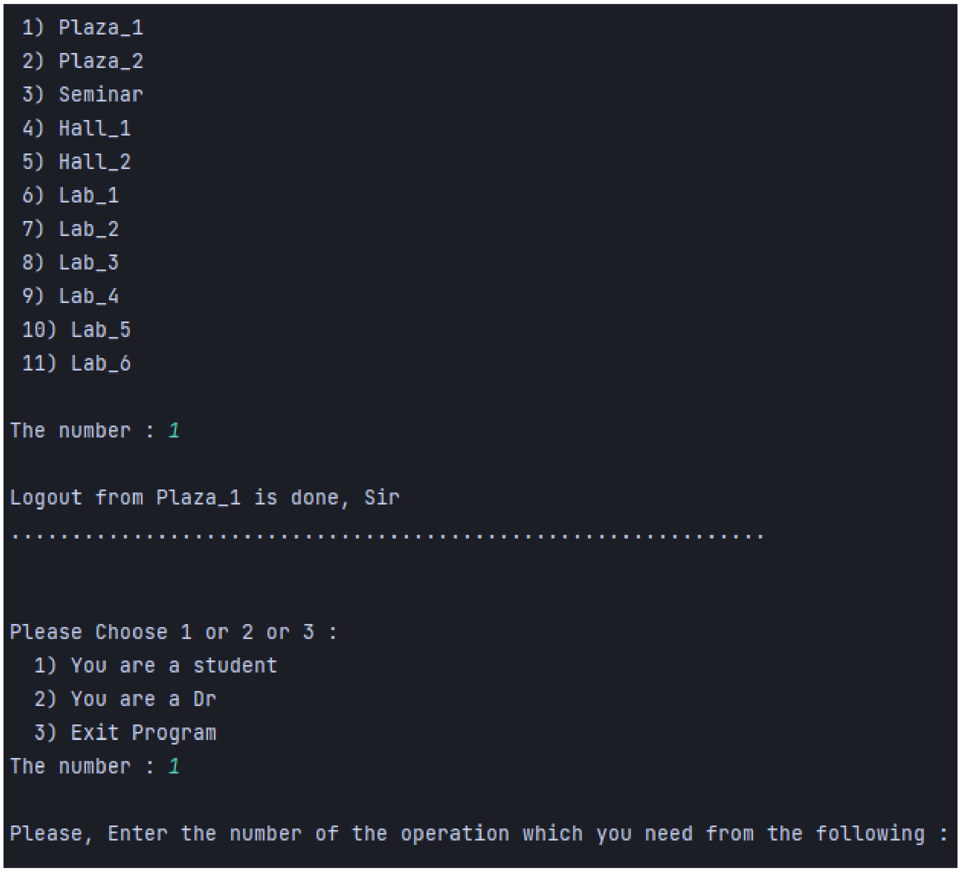
```
The number :
1

Enter your name please:
Hu

Enter the course name pleas:
OOP

Login to Plaza_1 is done, Sir
................................................................



Please Choose 1 or 2 or 3 :
  1) You are a student
  2) You are a Dr
  3) Exit Program
The number : 2

Enter your name please:
Gh

Enter the course name pleas:
OOp

Please, Enter the number of the operation which you need from the following :
```



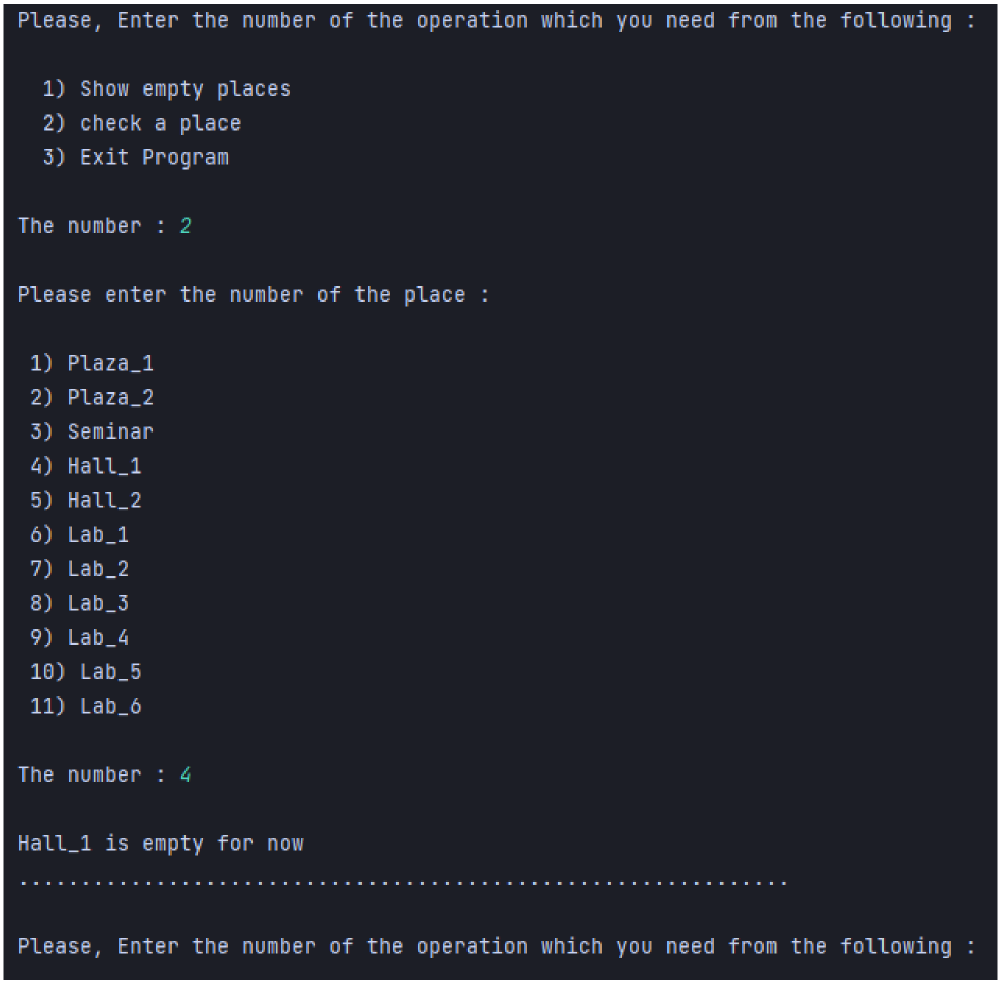
```
Please, Enter the number of the operation which you need from the following :

  1) Show empty places
  2) check a place
  3) Login to a place
  4) Logout from a place
  5) Exit Program

The number : 3

Please enter the number of the place :

 1) Plaza_1
 2) Plaza_2
 3) Seminar
 4) Hall_1
 5) Hall_2
 6) Lab_1
 7) Lab_2
 8) Lab_3
 9) Lab_4
 10) Lab_5
 11) Lab_6

The number : 1

Sorry, but Dr.Hu in Plaza_1 now for course OOP
```



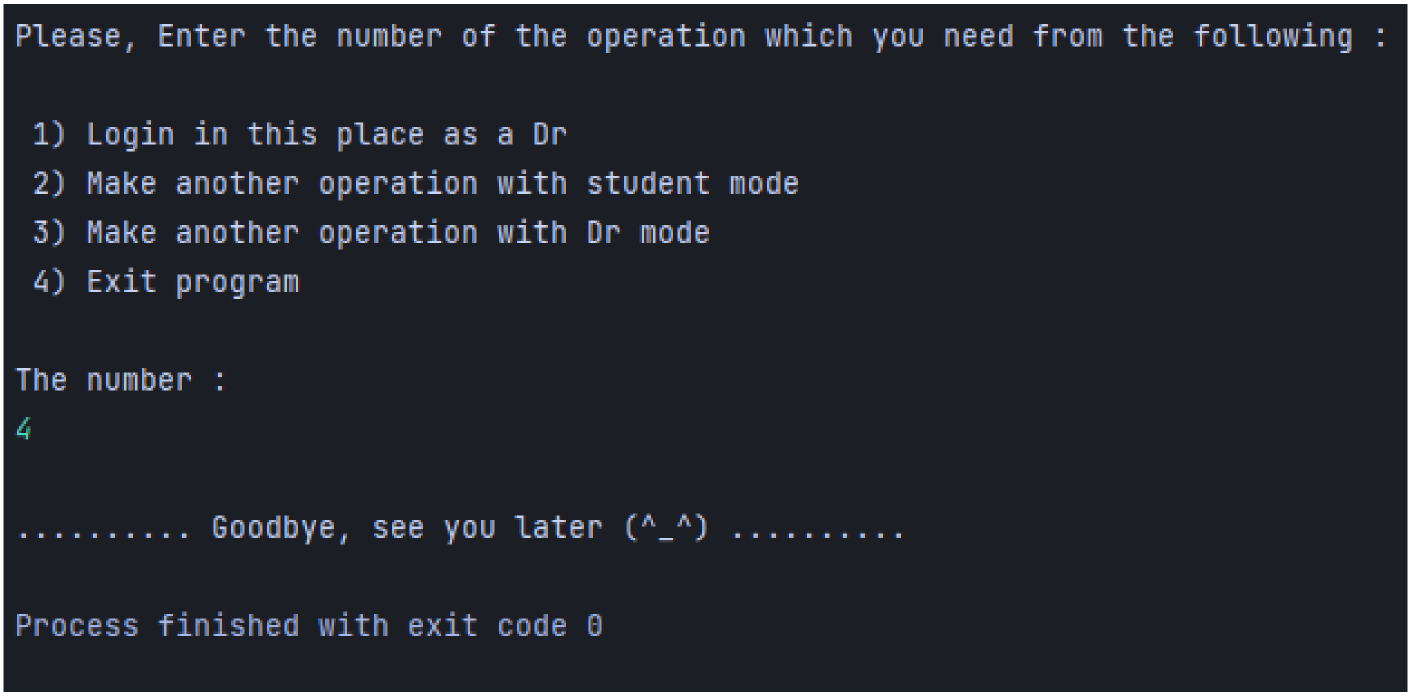
```
Sorry, but Dr.Hu in Plaza_1 now for course OOP
..............................................................

Enter your name please:
Hu

Enter the course name pleas:
OOP

Please, Enter the number of the operation which you need from the following :

  1) Show empty places
  2) check a place
  3) Login to a place
  4) Logout from a place
  5) Exit Program

The number : 4

Please enter the number of the place :

 1) Plaza_1
 2) Plaza_2
 3) Seminar
 4) Hall_1
 5) Hall_2
 6) Lab_1
 7) Lab_2
```



```
 1) Plaza_1
 2) Plaza_2
 3) Seminar
 4) Hall_1
 5) Hall_2
 6) Lab_1
 7) Lab_2
 8) Lab_3
 9) Lab_4
 10) Lab_5
 11) Lab_6

The number : 1

Logout from Plaza_1 is done, Sir
.............................................................



Please Choose 1 or 2 or 3 :
  1) You are a student
  2) You are a Dr
  3) Exit Program
The number : 1

Please, Enter the number of the operation which you need from the following :
```



```
Please, Enter the number of the operation which you need from the following :

  1) Show empty places
  2) check a place
  3) Exit Program

The number : 2

Please enter the number of the place :

 1) Plaza_1
 2) Plaza_2
 3) Seminar
 4) Hall_1
 5) Hall_2
 6) Lab_1
 7) Lab_2
 8) Lab_3
 9) Lab_4
 10) Lab_5
 11) Lab_6

The number : 4

Hall_1 is empty for now
..............................................................

Please, Enter the number of the operation which you need from the following :
```



```
Please, Enter the number of the operation which you need from the following :

 1) Login in this place as a Dr
 2) Make another operation with student mode
 3) Make another operation with Dr mode
 4) Exit program

The number :
4


.......... Goodbye, see you later (^_^) ..........

Process finished with exit code 0
```