Systems Analysis and Design

Instructor : Huang, Chuen-Min

Teamwork2 ver.1

Group 5

ID	Name
B10400075	Jim
B10423018	Sam
B10423051	Tady
B10423019	Javar
B10323022	Awy
B10523029	Timothy
B10523049	Ian
B10523035	Charly
B10523002	Ivan

Date 2018/05/29

Contents Page

1.Please explain the Law of Demeter (LoD) by using of your project.

(1) to itself

```
3  public class OTP {
4      private String OTPtoken;
5
6      public void GenerateAndSendOTP(){
7          int random;
8
9          random=(int)(Math.random()*900000)+100000;
10         String newotp = Integer.toString(random);
11         System.out.printf("Send OTPtoken : %s\n",newotp);
12         updateOTP(newotp);
13     }
14         public void updateOTP(String NewOTP) {
15         OTPtoken = NewOTP;
16     }
17         public boolean verifyOTPnumber(String rOTP) {
18             if (rOTP.equals(OTPtoken)) {
19                 return true;
20             }
21             else {
22                 return false;
23             }
24         }
25 }
```

(2) to objects contained in attributes of itself or a superclass

```
Scanner scan = new Scanner(System.in);
private OTP otp;
private User user;
private Room room;
private Date in;
private Date out;
private DBmgr dbmgr = new DBmgr();
```

```
public void submitSendOTP(){
    otp = new OTP();
    otp.GenerateAndSendOTP(user);
}
```

(3) to an object that is passed as a parameter to the method

```
public void GenerateAndSendOTP(User user){
    int random;

    random=(int)(Math.random()*900000)+100000;
    String newotp = Integer.toString(random);

    System.out.printf("sending OTP: '%s'to'%s'\n ",newotp,user.getphonenumber());

    updateOTP(newotp);
}
```

(4) to an object that is created by the method

```
public void submitSendOTP(){
    otp = new OTP();
    otp.GenerateAndSendOTP(user);
}
```

2.There are six (or seven) types of interaction coupling, each falling on different parts of a good-to-bad continuum. Choose three pieces of your project to describe what types of the coupling they belong to.

(1)Data:

The calling method(submitOTPtoken) passes a variable(rotp) to the called method(verifyOTPnumber).

```
public boolean submitOTPtoken(String rotp) {
    return this.otp.verifyOTPnumber(rotp);
}
```

```
public void updateOTP(String NewOTP) {
    OTPtoken = NewOTP;
}
public boolean verifyOTPnumber(String rotp) {
    if (rotp.equals(OTPtoken)) {
        return true;
    }
    else {
        return false;
    }
}
```

(2) Stamp:

```
public void submitSendOTP(){
    otp = new OTP();
    otp.GenerateAndSendOTP(user);
}
```

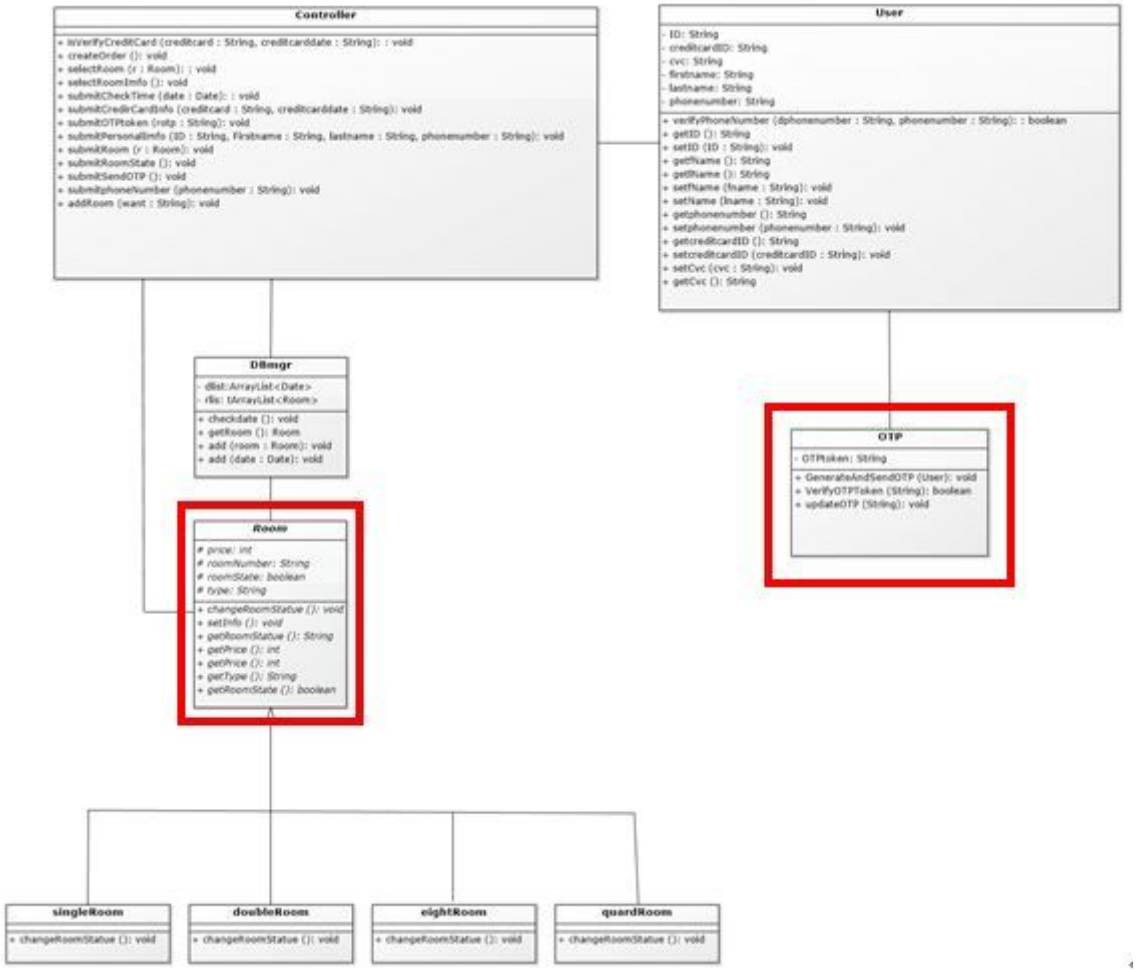
```
public void GenerateAndSendOTP(User user){
    int random;

    random=(int)(Math.random()*900000)+100000;
    String newotp = Integer.toString(random);

    System.out.printf("sending OTP: '%s'to'%s'\n ",newotp,user.getphonenumber());

    updateOTP(newotp);
}
```

(3) No Direct Coupling: OTP do not relate to Room, they do not call one another.



3.There are seven types of method cohesion, choose three pieces of your project to describe what types of the cohesion they belong to.

(1) Functional:

VerifyOTPToken verify the OTP token.

```
    public boolean VerifyOTPToken(String rotp) {
        if (rotp.equals(OTPtoken)) {
            return true;
        }
        else {
            return false;
        }
    }
}
```

(2) Temporal:

```
    public void submitphoneNumber(String phonenumber) {
        while (!(user.verifyPhoneNumber(phonenumber,user.getphonenumber()))){
        System.out.printf("  Doesn't match! \n");
        System.out.printf("Renter your phonenumber:");
        phonenumber = scan.nextLine();}
        System.out.printf("  Success!\n");
    }
```

(3) Sequential:

User.getphonenumber ‘s output is User.getphonenumber ‘s input.

```
    public void submitphoneNumber(String phonenumber) {
        while (!(user.verifyPhoneNumber(phonenumber,user.getphonenumber()))){
        System.out.printf("  Doesn't match! \n");
        System.out.printf("Renter your phonenumber:");
        phonenumber = scan.nextLine();}
        System.out.printf("  Success!\n");
    }

    public void submitCreditCardInfo(String creditcard,String creditcarddate) {
        user.setcreditcardID(creditcard);
        user.setCvc(creditcarddate);
    }
```

4.Connascence generalized the ideas of cohesion and coupling, use three pieces of your project to describe what types of the connascence they belong to.

(1)Type:OTPtoken

"VerifyOTPtoken" use the attribute "OTPtoken" which is declared as "String", change its type then it might cause error when comparing with "rotp".

```
package RoomReservationSystem;
public class OTP {
    private String OTPtoken;

    public void GenerateAndSendOTP(User user){
        int random;

        random=(int)(Math.random()*900000)+100000;
        String newotp = Integer.toString(random);

        System.out.printf("sending OTP: '%s'to'%s'\n ",newotp,user.getphonenumber());

        updateOTP(newotp);
    }
    public void updateOTP(String NewOTP) {
        OTPtoken = NewOTP;
    }
    public boolean VerifyOTPToken(String rotp) {
        if (rotp.equals(OTPtoken)) {
            return true;
        }
        else {
            return false;
        }
    }
}
```

(2)Name: roomNumber

If the name of the attribute "roomNumber" changes, the method "getID" will have to change

```
1  package RoomReservationSystem;
2  import java.util.*;
3
4  public class User {
5      private String ID;
6      private String firstname;
7      private String lastname;
8      private String phonenumber;
9      private String creditcardID;
10     private String cvc;
11
12     public User(String ID,String Firstname,String Lastname,String phonenumber) {
13         this.ID = ID;
14         this.firstname = Firstname;
15         this.lastname = Lastname;
16         this.phonenumber = phonenumber;
17         }
18
```

```
    public String getID() {
        return ID;
    }

    public void setID(String ID) {
        this.ID = ID;
    }

    public String getfName() {
        return firstname;
    }

    public String getlName() {
        return lastname;
    }
```

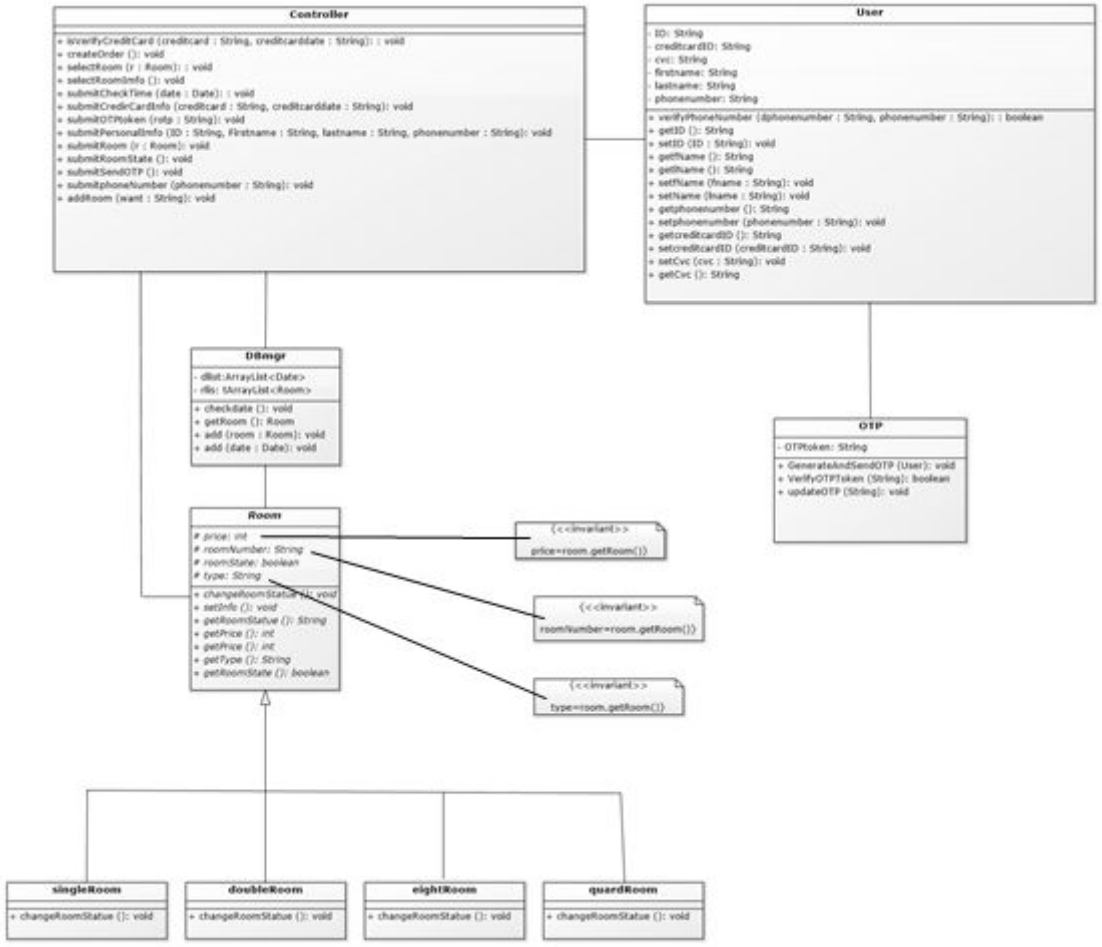
(3)Position:

If “otp.GenerateAndSendOTP” change to there, the result will be wrong.

```
    public void submitOTPtoken(String rotp) {
        while (!(this.otp.VerifyOTPToken(rotp))) {
            System.out.printf("  Error\n");
            otp.GenerateAndSendOTP(user);
            System.out.printf("Reenter OTP:");
            rotp = scan.nextLine();
        }
    }
```

5.Use one class from your project that can create a set of invariants and add them to the CRC card or the class diagram.

Room class has three invariant price,roomNumber,and type



6.Use a method of a class from your project that can create a contract and describe its algorithm specification. Specify the pre- or post- condition and use both Structured English and an activity diagram to specify the algorithm.

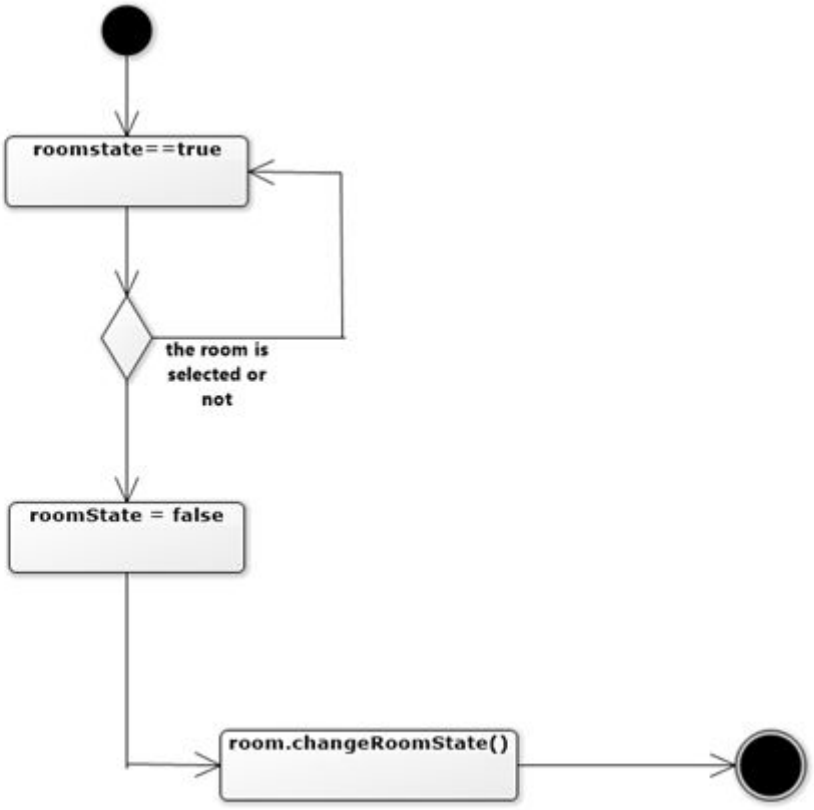
Pre and post condition

Method Name: changeRoomState	Class Name: SingleRoom	ID:3
Client(consumers):Controller		
Associated of Use case: Reserve Stay		
Description of Responsibilities: The method is mainly change Room's state		
Arguments Received: None.		
Type of Value Returned: void		
Pre-Condition: Customer choose room Controller.sumbitRoom()		
Post-Condition Room's state be changed		

Structured English

```
if(roomState == true)
                roomState = false;
        else
                roomState = true;
room.changeRoomState()
```

Activity Diagram



7.Please evaluate any piece of your project in terms of cohesion, coupling, and connascence perspective.

We evaluate class “Course_List” in terms of cohesion , coupling , and connascence.

```java
package RoomReservationSystem;
public class OTP {
    private String OTPtoken;


    public void GenerateAndSendOTP(User user){
        int random;
        String newotp = null;


        random=(int)(Math.random()*900000)+100000;
        newotp = Integer.toString(random);

        System.out.printf("sending OTP: '%s'to'%s'\n ",newotp,user.getphonenumber());

        updateOTP(newotp);
    }
    public void updateOTP(String NewOTP) {
        OTPtoken = NewOTP;
    }
    public boolean VerifyOTPToken(String rotp) {
        if (rotp.equals(OTPtoken)) {
            return true;
        }
        else {
            return false;
        }
    }
}
```

Coupling:Data	The calling method passes a variable to the called method Example:Controller’s method submitOTPtoken calls OTP’s method VerifyOTPToken and passes rotp.
Coupling: Stamp	The method “GenerateAndSendOTP” only uses part of argument (user.getphonenumber).
Method Cohesion: Functional	updateOTP update the OTP token.
Class Cohesion: Ideal	“OTP” has none of the mixed cohesion.
Connascence: Type	OTPtoken is an attribute which type is String.
Connascence: Name	updateOTP method set NewOTP to attribute named OTPtoken.
Connascence: Position	In method GenerateAndSendOTP,if “updateOTP(newotp)” change to “String newOTP = null”,it will be wrong.

8.Assume that you are going to adopt RDBMs to your project, please describe the referential integrity.

Room Table

Room Table			
Room Number	Type	Price	State
101	Single	2000	Reserved
102	Single	2000	Reserved
103	Single	2000	Unreserved
201	Double	2500	Unreserved
202	Double	2500	Reserved
203	Double	2500	Unreserved
401	Quad	3000	Reserved
402	Quad	3000	Unreserved
403	Quad	3000	Unreserved
801	Eight	4000	Reserved
802	Eight	4000	Reserved
803	Eight	4000	Unreserved

Customer Table

Customer Table					
CustomerID	First name	Last name	Phone Number	Creditcard Number	CVC
I226756199	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874
A124027532	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144
Q124162583	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123
K126174825	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456
J124157852	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552
M226741015	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852
B124230187	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798
M129865742	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465
P129864533	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854

Reserve order Table

Reserve order Table					
Reserve Number	Room Number	CustomerID	Reserve Date	Checkin Date	Checkout Date
005	101	I226756199	2018/5/26	2018/5/30	2018/5/25
006	102	A124027532	2018/5/28	2018/6/5	2018/5/25
001	202	Q124162583	2018/5/26	2018/5/27	2018/5/24
002	401	K126174825	2018/5/30	2018/5/31	2018/5/24
007	401	J124157852	2018/6/1	2018/6/2	2018/5/25
003	801	M226741015	2018/6/2	2018/6/5	2018/5/24
008	801	B124230187	2018/5/30	2018/6/1	2018/5/25
004	802	M129865742	2018/5/28	2018/5/31	2018/5/24
009	801	P129864533	2018/6/2	2018/6/4	2018/5/25

Referential integrity in our project is in order table. OrderTable is reference other two table. If RoomNumber in Room Table has been changed, RoomNumber in Reserve order Table must been changed too. If CustomerID in Customer Table has been changed, CustomerID in Reserve order Table also been changed. Suppose if only change one table's data and each one doesn't been changed, it will cause reference error. The data in this table can't reference to another data in other table. Then the data can't be display correctly and completely.

9.Using the steps of normalization, create a model that represents the file of your project in third normal form. Please make necessary assumptions to explain why the tables are related.

0NF: One single row has many info in it. Like example, Room Number 401 has two reservation informations in it.

Room Number	Type	Price	State	Reserve Number	Reserve Date	Checkin Date	Checkout Date	CustomerID	First name	Last name	Phone Number	Creditcard Number	CVC
101	Single	2000	Reserved	005	2018/5/26	2018/5/30	2018/5/25	I226756199	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874
102	Single	2000	Reserved	006	2018/5/28	2018/6/5	2018/5/25	A124027532	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144
103	Single	2000	Unreserved										
201	Double	2500	Unreserved										
202	Double	2500	Reserved	001	2018/5/26	2018/5/27	2018/5/24	Q124162583	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123
203	Double	2500	Unreserved										
401	Quad	3000	Reserved	002	2018/5/30	2018/5/31	2018/5/24	K126174825	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456
				007	2018/6/1	2018/6/2	2018/5/25	J124157852	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552
402	Quad	3000	Unreserved										
403	Quad	3000	Unreserved										
801	Eight	4000	Reserved	003	2018/6/2	2018/6/5	2018/5/24	M226741015	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852
				008	2018/5/30	2018/6/1	2018/5/25	B124230187	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798
802	Eight	4000	Reserved	004	2018/5/28	2018/5/31	2018/5/24	M129865742	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465
				009	2018/6/2	2018/6/4	2018/5/25	P129864533	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854
803	Eight	4000	Unreserved										

1NF:Each table cell should contain a single value.

Room Number	Type	Price	State	Reserve Number	Reserve Date	Checkin Date	Checkout Date	CustomerID	First name	Last name	Phone Number	Creditcard Number	CVC
101	Single	2000	Reserved	005	2018/5/26	2018/5/30	2018/5/25	I226756199	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874
102	Single	2000	Reserved	006	2018/5/28	2018/6/5	2018/5/25	A124027532	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144
103	Single	2000	Unreserved										
201	Double	2500	Unreserved										
202	Double	2500	Reserved	001	2018/5/26	2018/5/27	2018/5/24	Q124162583	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123
203	Double	2500	Unreserved										
401	Quad	3000	Reserved	002	2018/5/30	2018/5/31	2018/5/24	K126174825	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456
401	Quad	3000	Reserved	007	2018/6/1	2018/6/2	2018/5/25	J124157852	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552
402	Quad	3000	Unreserved										
403	Quad	3000	Unreserved										
801	Eight	4000	Reserved	003	2018/6/2	2018/6/5	2018/5/24	M226741015	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852
801	Eight	4000	Reserved	008	2018/5/30	2018/6/1	2018/5/25	B124230187	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798
802	Eight	4000	Reserved	004	2018/5/28	2018/5/31	2018/5/24	M129865742	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465
801	Eight	4000	Reserved	009	2018/6/2	2018/6/4	2018/5/25	P129864533	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854
803	Eight	4000	Unreserved										

2NF:We have divided our 1NF table into two tables- Room table & Reserve Order Table. Room table contains room information and its state. Reserve Order Table contains information on orders and customer information (CustomerID & First Name & Last Name & Phone Number & Credit Card ID & CVC) are partially dependent on reserve order table. We have the primary key called Room Number in Room table. In Reserve Order Table, Room Number is the foreign key.

Room Table

Room Number	Type	Price	State
101	Single	2000	Reserved
102	Single	2000	Reserved
103	Single	2000	Unreserved
201	Double	2500	Unreserved
202	Double	2500	Reserved
203	Double	2500	Unreserved
401	Quad	3000	Reserved
402	Quad	3000	Unreserved
403	Quad	3000	Unreserved
801	Eight	4000	Reserved
802	Eight	4000	Reserved
803	Eight	4000	Unreserved

Reserve order Table

Reserve Number	Room Number	CustomerID	First name	Last name	Phone Number	Creditcard Number	CVC	Reserve Date	Checkin Date	Checkout Date
005	101	I226756199	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874	2018/5/26	2018/5/30	2018/5/25
006	102	A124027532	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144	2018/5/28	2018/6/5	2018/5/25
001	202	Q124162583	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123	2018/5/26	2018/5/27	2018/5/24
002	401	K126174825	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456	2018/5/30	2018/5/31	2018/5/24
007	401	J124157852	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552	2018/6/1	2018/6/2	2018/5/25
003	801	M226741015	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852	2018/6/2	2018/6/5	2018/5/24
008	801	B124230187	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798	2018/5/30	2018/6/1	2018/5/25
004	802	M129865742	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465	2018/5/28	2018/5/31	2018/5/24
009	801	P129864533	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854	2018/6/2	2018/6/4	2018/5/25

3NF: To move our 2NF table into 3NF, we need to divide our table to solve partially dependent.

Room Table

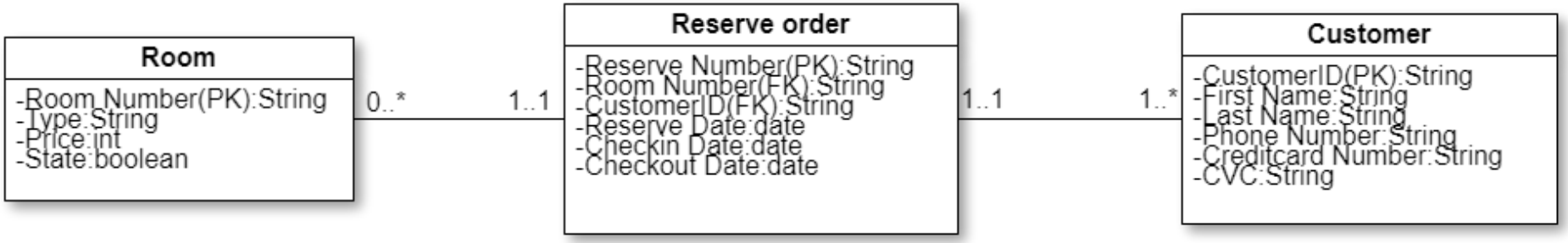
Room Number	Type	Price	State
101	Single	2000	Reserved
102	Single	2000	Reserved
103	Single	2000	Unreserved
201	Double	2500	Unreserved
202	Double	2500	Reserved
203	Double	2500	Unreserved
401	Quad	3000	Reserved
402	Quad	3000	Unreserved
403	Quad	3000	Unreserved
801	Eight	4000	Reserved
802	Eight	4000	Reserved
803	Eight	4000	Unreserved

Customer Table

CustomerID	First name	Last name	Phone Number	Creditcard Number	CVC
I226756199	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874
A124027532	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144
Q124162583	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123
K126174825	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456
J124157852	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552
M226741015	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852
B124230187	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798
M129865742	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465
P129864533	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854

Reserve order Table

Reserve Number	Room Number	CustomerID	Reserve Date	Checkin Date	Checkout Date
005	101	I226756199	2018/5/26	2018/5/30	2018/5/25
006	102	A124027532	2018/5/28	2018/6/5	2018/5/25
001	202	Q124162583	2018/5/26	2018/5/27	2018/5/24
002	401	K126174825	2018/5/30	2018/5/31	2018/5/24
007	401	J124157852	2018/6/1	2018/6/2	2018/5/25
003	801	M226741015	2018/6/2	2018/6/5	2018/5/24
008	801	B124230187	2018/5/30	2018/6/1	2018/5/25
004	802	M129865742	2018/5/28	2018/5/31	2018/5/24
009	801	P129864533	2018/6/2	2018/6/4	2018/5/25

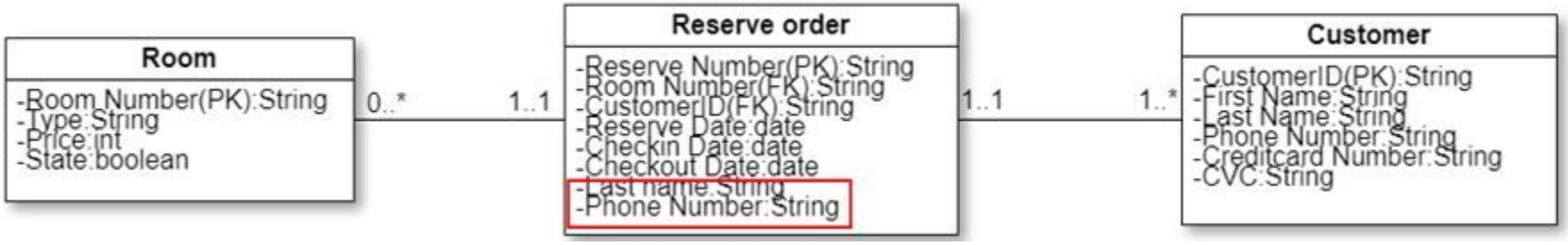

Room
-Room Number(PK):String
-Type:String
-Price:int
-State:boolean
0..*
1..1
Reserve order
-Reserve Number(PK):String
-Room Number(FK):String
-CustomerID(FK):String
-Reserve Date:date
-Checkin Date:date
-Checkout Date:date
1..1
1..*
Customer
-CustomerID(PK):String
-First Name:String
-Last Name:String
-Phone Number:String
-Creditcard Number:String
-CVC:String

10.Describe how you would denormalize the model that you created in question 9. Draw the new class diagram based on your suggested changes.

Every time the reservation completed, which means a customer has reserved a room(s), in order to boost the speed of data in and out, we've decided to denormalize, adding the customer's Last Name and Phone Number into Reserve Order Table.

Reserve order Table

Reserve Number	Room Number	CustomerID	Reserve Date	Checkin Date	Checkout Date	Last name	Phone Number
005	101	I226756199	2018/5/26	2018/5/30	2018/5/25	Johansson	(09) 1111-1111
006	102	A124027532	2018/5/28	2018/6/5	2018/5/25	Lawrence	(09) 2222-2222
001	202	Q124162583	2018/5/26	2018/5/27	2018/5/24	Statham	(09) 1234-5678
002	401	K126174825	2018/5/30	2018/5/31	2018/5/24	Downey	(09) 8765-4321
007	401	J124157852	2018/6/1	2018/6/2	2018/5/25	Martin	(09) 7766-5555
003	801	M226741015	2018/6/2	2018/6/5	2018/5/24	Cruise	(09) 2345-6789
008	801	B124230187	2018/5/30	2018/6/1	2018/5/25	Watson	(09) 8755-6421
004	802	M129865742	2018/5/28	2018/5/31	2018/5/24	Walker	(09) 3456-7890
009	801	P129864533	2018/6/2	2018/6/4	2018/5/25	Mars	(09) 9123-4567



11.Examine the model that you created in question 10. Develop the inter-file clustering and index strategies. Describe how your clustering strategy will improve the performance of the database. List possible indices you would recommend and describe the reasons.

We used our customer information & order information as inter-file clustering, because of frequently using. One order must correspond to one customer information.

We choose reservation date as the index of Reserve Order table. Every order must have the time checking in and out, so as the index, it boosts the way we want to find all reservations in a particular date, also it can perform as a calculator to calculate the number of reservations in that date.

Reserve Date Index

Reserve Date	Pointer
2018/5/26	*
2018/5/26	*
2018/5/28	*
2018/5/28	*
2018/5/30	*
2018/5/30	*
2018/6/1	*
2018/6/2	*
2018/6/2	*

Reserve order Table

Reserve Number	Room Number	CustomerID	Reserve Date	Checkin Date	Checkout Date	First name	Last name	Phone Number	Creditcard Number	CVC
005	101	I226756199	2018/5/26	2018/5/30	2018/5/25	Scarlett	Johansson	(09) 1111-1111	7777-8888-9999-0000	874
006	102	A124027532	2018/5/28	2018/6/5	2018/5/25	Jennifer	Lawrence	(09) 2222-2222	9999-8888-7777-6666	144
001	202	Q124162583	2018/5/26	2018/5/27	2018/5/24	Jason	Statham	(09) 1234-5678	1111-2222-3333-4444	123
002	401	K126174825	2018/5/30	2018/5/31	2018/5/24	Robert	Downey	(09) 8765-4321	5555-6666-7777-8888	456
007	401	J124157852	2018/6/1	2018/6/2	2018/5/25	Frank	Martin	(09) 7766-5555	5555-4444-3333-2222	552
003	801	M226741015	2018/6/2	2018/6/5	2018/5/24	Tom	Cruise	(09) 2345-6789	9999-0000-1111-2222	852
008	801	B124230187	2018/5/30	2018/6/1	2018/5/25	Emma	Watson	(09) 8755-6421	8888-0000-7777-1111	798
004	802	M129865742	2018/5/28	2018/5/31	2018/5/24	Paul	Walker	(09) 3456-7890	3333-4444-5555-6666	465
009	801	P129864533	2018/6/2	2018/6/4	2018/5/25	Bruno	Mars	(09) 9123-4567	6666-5555-7777-4444	854

Score Chart

ID	Name	Score	Implementation content
B10400075	Jim	100%	Answer the quations,write document,Presenter
B10423018	Sam	100%	Answer the quations,write document,Presenter
B10423051	Tady	100%	Answer the quations,coder,write document,Presenter
B10423019	Javar	100%	Answer the quations,write document,Presenter
B10323022	Awy	100%	Answer the quations,verify document,Presenter
B10523029	Timothy	100%	Answer the quations,coder,translator,Presenter
B10523049	Ian	70%	Presenter
B10523035	Charly	70%	Presenter
B10523002	Ivan	70%	Show up in meeting