Online Hotel Reservation System Project

CS 631– Database Management and System Design

Project Report

GROUP 9:

NEEL PATEL
(nap73@njit.edu)

JASMINKUMAR PATEL
(jp878@njit.edu)

PROFESSOR:

Ananya Dass

INDEX

- Aim and Objective
- Revisions
- Problems Encountered
- Application Implementation
- Programming Concept
- Database Creation
- Table Population
- SQL Queries of Table Creation
- SQL Queries of Table Population (Partial)
- Other Important Queries for STATISTICS of best room,best service, best breakfast, top 5 customers
- Users Guide to Run Program
- Application Flow of User
- Application Flow of Admin
- Imp Queries

Aim and Objective

The main aim of this project is to construe a Relational schema to a real-world database structure, that requires populating the database, and creating a web application for chain of Hotels Hulton.

Revisions

1. Adding a new attribute named c_id in table of credit_card, it is a foreign key to table customer c_id. Based on this action, every customer can store a lots of credit cards in our database, they can choose any card to pay their reservation.

2. Creating a new table named admin, to store the admin account information. The information stored is admin username and password.

Problems Encountered

- PORT ERROR
  Default Tomcat port is 8086, which remained in use. Making some necessary changes the connection was established.

- DEVELOPING HOTEL WEBSITE
  Designing and implementing an interface for the website. Adding different features and ease from the point of view of a customer took more research time than expected. Studiny and using MVC model, the problem was resolved.

- INTERFACING WITH THE SERVICES
  There were problems letting a room reservation include multiple breakfast/service types and can have multiple orders for each breakfast type, and the total number of orders for different breakfast types for a room reservation should not exceed the number of persons that can stay in that room. Referring a book and couple of examples helped in solving problems.

Application Implementation

Tools that we use:

1. JDK 1.8.0

2. Eclipse IDE for Java EE Developers Version: Neon.1a Release (4.6.1)

3. Apache-tomcat-7.0.73

4. MySQL WorkBench 8

5. Safari Browser

Programming Concept

Entire website is created as MVC(Model–view–controller) model. MVC is a software architectural pattern for implementing user interfaces on computers. It divides a given application into three interconnected parts in order to separate internal representations of information from the ways that information is presented to and accepted from the user. The MVC design pattern decouples these major components allowing for efficient code reuse and parallel development.

Different layers of the source code, i.e. Dao, Servlet, Mapper, Beans , JSP

(1) Dao layer:

Communication between the entire program and the database. It has interfaces which contains of methods used to interact with the database in order to fetch required data for application interaction.

(2) Beans layer:

Each bean contain variables of the database tables and their setter & getter methods. They are populated during setting the value or getting the value during database interactions. This helps in easy redirections and access of required variables.

(3) Servlet layer:

This layer is equal to the control layer of MVC, it is responsible for controlling the flow of web application. The function of servlet layer is processing user input from jsp and processing it for and processing the data, before returning it back to jsp. It is responsible for calling core methods for the data processing.

(4) Mapper layer:

It act as the glue layer that connects the dao layer and the servlet layer. It has implementation for all the methods used to store-fetch-update data from the database. It implements the dao interface and has query processing content.

(5) JSP layer:

It act as the presentation layer of the application. It contains JSP elements that provides user interfaces and view of the application. The page can use by the users we put them in webcontent folder. In jsp communicate with server in two ways. One is in the form way while other is using href of a label.

Database Concept

The database used for this web application is MYSQL. We used MYSQL WorkBench 8.0 as an interface of MYSQL. All is commands used for creation & interaction of the database are mentioned in upcoming section of this report.

Creating a MYSQL Connection :

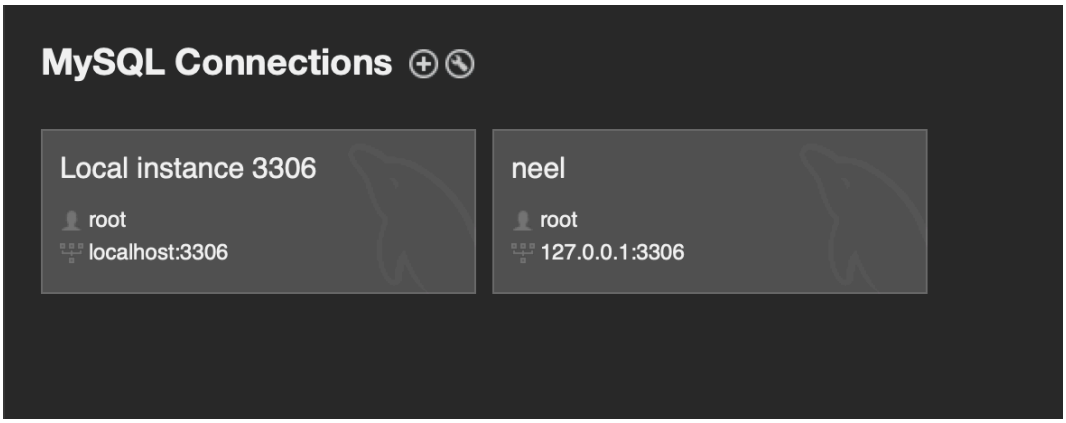
1.Open **MySQL Workbench**
2. Click **New Connection** towards the bottom left of MySQL Workbench

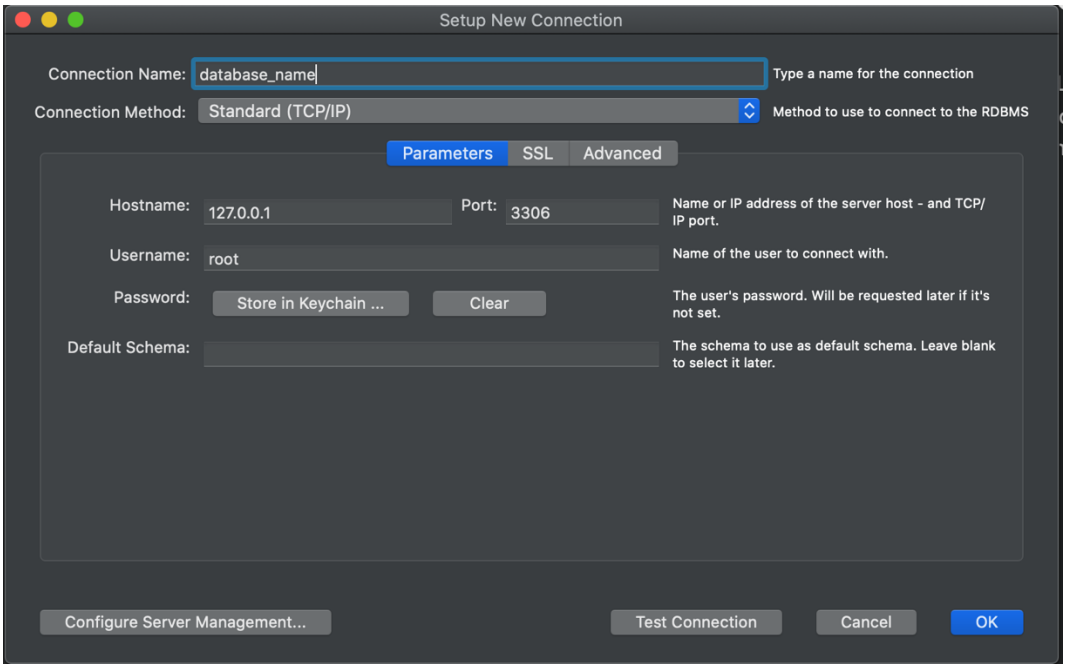
3. In the "*Set up a New Connection Dialogue*" box, Type your Database connection credentials. The credentials will be like the following:

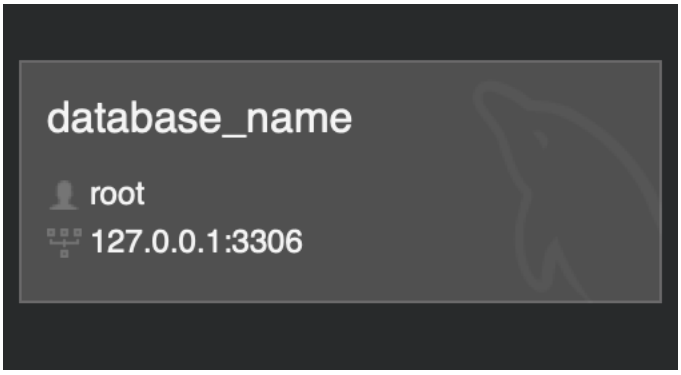
Connection Name: You can name this whatever you like.
Connection Method: Standard (TCP/IP).
Hostname: You can use your domain our your cPanel IP address.
Port: 3306
Username: Your cPanel username or the user you created for the database.
Password: cPanel password or the password for the database user that was created.
Default Schema: This can be left blank.

Click **Test Connection**.

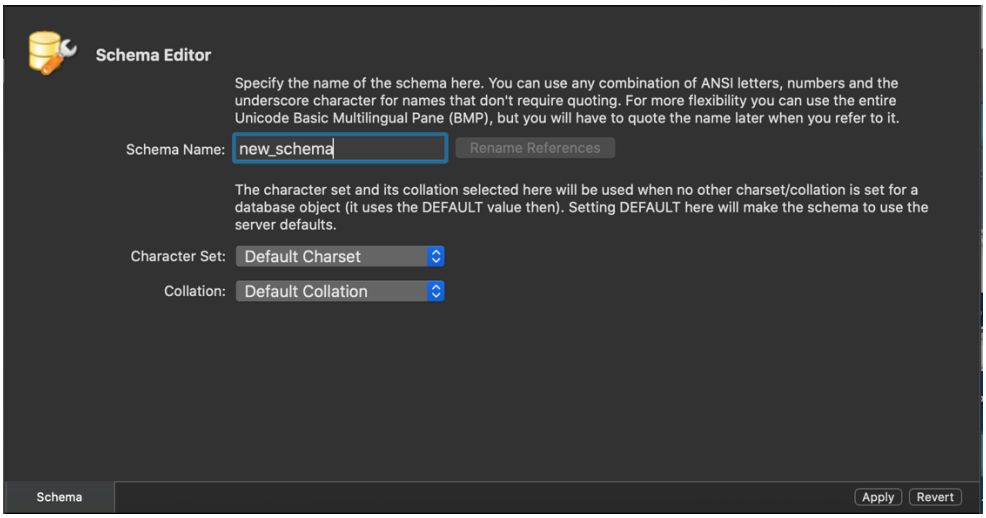
4.Type your password and click the "*Save Password in Vault*" check box.
Click **Ok**.

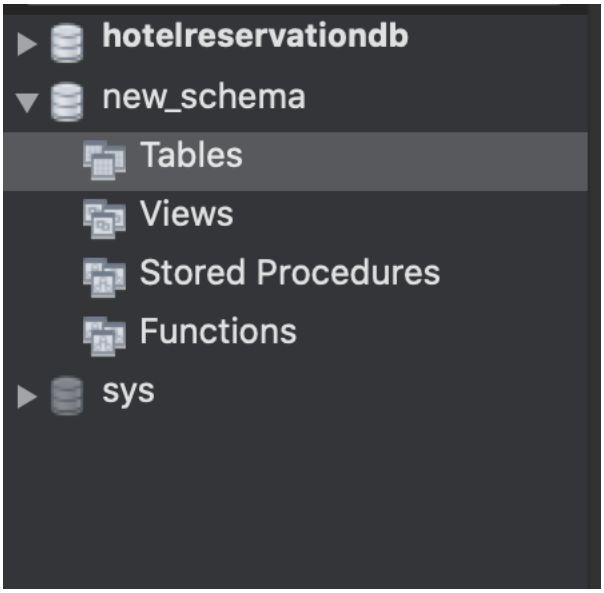

MySQL Connections
Local instance 3306
root
localhost:3306
neel
root
127.0.0.1:3306

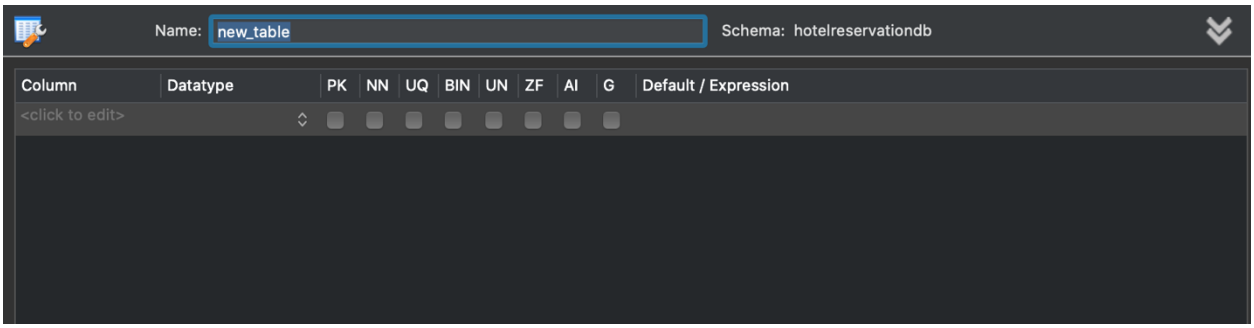

Setup New Connection
Connection Name:
database_name
Type a name for the connection
Connection Method:
Standard (TCP/IP)
Method to use to connect to the RDBMS
Parameters
SSL
Advanced
Hostname:
127.0.0.1
Port:
3306
Name or IP address of the server host - and TCP/IP port.
Username:
root
Name of the user to connect with.
Password:
Store in Keychain ...
Clear
The user's password. Will be requested later if it's not set.
Default Schema:
The schema to use as default schema. Leave blank to select it later.
Configure Server Management...
Test Connection
Cancel
OK

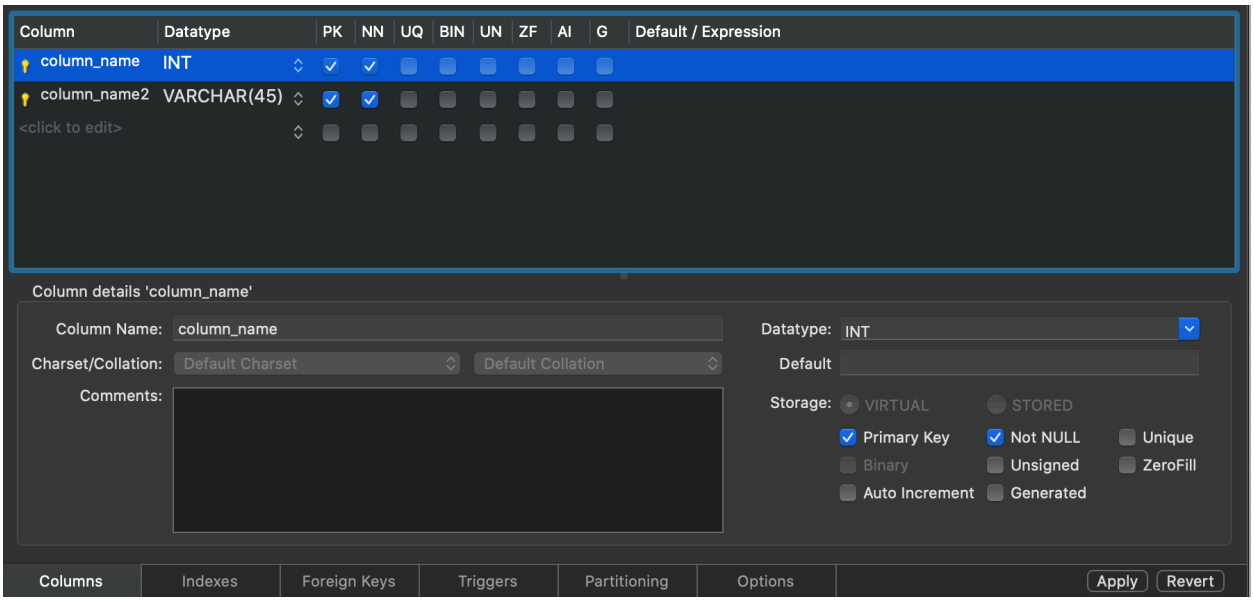

database_name
root
127.0.0.1:3306

Creating a new Database :









➔ CREATE SCHEMA `new_schema` ;

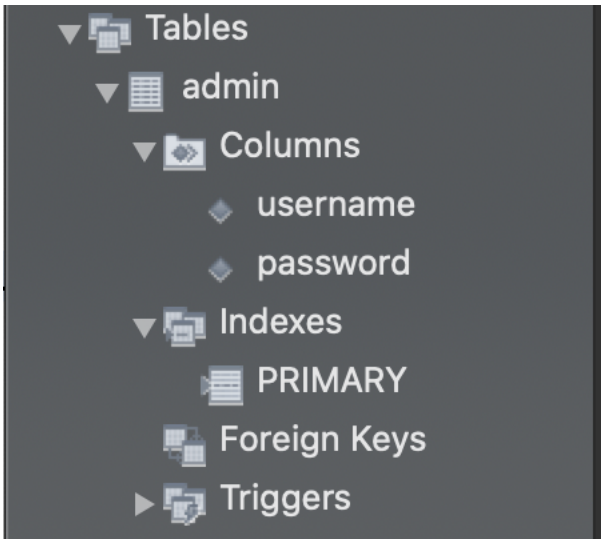
➔ CREATE TABLE `hotelreservationdb`.`new_table` (

`column_name` INT NOT NULL,

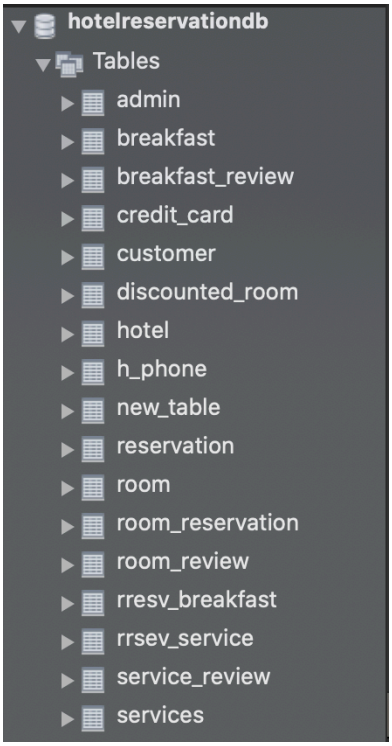
`column_name2` VARCHAR(45) NOT NULL,

PRIMARY KEY (`column_name2`, `column_name`));

Based our relation schema, insert the attribute names, primary keys, foreign keys.



List of Tables used in Web Application



Queries

CREATION OF TABLES :

1.BREAKFAST

```
CREATE TABLE `breakfast` (
 `btype` varchar(100) NOT NULL,
 `hotelid` int(11) NOT NULL,
 `bdes` varchar(250) DEFAULT NULL,
 `bprice` int(11) DEFAULT NULL,
 PRIMARY KEY (`btype`,`hotelid`),
 KEY `hotelid_idx` (`hotelid`),
 CONSTRAINT `hotelid` FOREIGN KEY (`hotelid`) REFERENCES `hotel` (`hotelld`)
```

2.BREAKFAST_REVIEW

```
CREATE TABLE `breakfast_review` (
 `brid` int(11) NOT NULL,
 `text` varchar(250) DEFAULT NULL,
 `cust_id` int(11) DEFAULT NULL,
 `brhotelid` int(11) DEFAULT NULL,
 `b_type` varchar(45) DEFAULT NULL,
 PRIMARY KEY (`brid`),
 KEY `brhotelid_idx` (`brhotelid`),
 KEY `b_type_idx` (`b_type`),
 KEY `cust_id_idx` (`cust_id`),
 CONSTRAINT `b_type` FOREIGN KEY (`b_type`) REFERENCES `breakfast` (`btype`),
 CONSTRAINT `brhotelid` FOREIGN KEY (`brhotelid`) REFERENCES `breakfast` (`hotelid`),
 CONSTRAINT `cust_id` FOREIGN KEY (`cust_id`) REFERENCES `customer` (`c_id`)
```

3.CREDIT_CARD

```
CREATE TABLE `credit_card` (
 `cnumber` varchar(16) NOT NULL,
 `name` varchar(45) DEFAULT NULL,
 `expdate` date DEFAULT NULL,
 `code` int(11) DEFAULT NULL,
 `ctype` varchar(45) DEFAULT NULL,
 `baddress` varchar(150) DEFAULT NULL,
 PRIMARY KEY (`cnumber`)
```

4.CUSTOMER

```
CREATE TABLE `customer` (
  `c_id` int(11) NOT NULL AUTO_INCREMENT,
  `phoneno` int(10) DEFAULT NULL,
  `name` varchar(45) DEFAULT NULL,
  `address` varchar(100) DEFAULT NULL,
  `email` varchar(45) DEFAULT NULL,
  `username` varchar(45) DEFAULT NULL,
  `password` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`c_id`)
```

5.DISCOUNTED_ROOM

```
CREATE TABLE `discounted_room` (
  `ohotelid` int(11) DEFAULT NULL,
  `room_num` int(11) DEFAULT NULL,
  `sdate` date DEFAULT NULL,
  `edate` date DEFAULT NULL,
  `discount` int(11) DEFAULT NULL,
  KEY `ohotelid_idx` (`ohotelid`),
  KEY `room_no_idx` (`room_num`),
  CONSTRAINT `ohotelid` FOREIGN KEY (`ohotelid`) REFERENCES `room` (`rhotelid`),
  CONSTRAINT `room_num` FOREIGN KEY (`room_num`) REFERENCES `room` (`roomnum`)
```

6.H_PHONE

```
CREATE TABLE `h_phone` (
  `phone_no` int(10) NOT NULL,
  `photelid` int(11) NOT NULL,
  PRIMARY KEY (`phone_no`,`photelid`),
  KEY `photeid_idx` (`photelid`),
  CONSTRAINT `photeid` FOREIGN KEY (`photelid`) REFERENCES `hotel` (`hotelId`)
```

7.HOTEL

```
CREATE TABLE `hotel` (
  `hotelId` int(11) NOT NULL,
  `street` varchar(250) DEFAULT NULL,
  `country` varchar(50) DEFAULT NULL,
  `state` varchar(50) DEFAULT NULL,
  `zip` varchar(45) DEFAULT NULL,
  `name` varchar(250) NOT NULL,
  `country_id` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`hotelId`)
```

8.RESERVATION

```
CREATE TABLE `reservation` (
  `invoice_no` int(11) NOT NULL,
  `rdate` date DEFAULT NULL,
  `tamount` int(11) DEFAULT NULL,
  `cid` int(11) DEFAULT NULL,
  `c_no` varchar(16) DEFAULT NULL,
  PRIMARY KEY (`invoice_no`),
  KEY `c_no_idx` (`c_no`),
  KEY `cid_idx` (`cid`),
  CONSTRAINT `c_no` FOREIGN KEY (`c_no`) REFERENCES `credit_card` (`cnumber`),
  CONSTRAINT `cid` FOREIGN KEY (`cid`) REFERENCES `customer` (`c_id`)
```

9.ROOM

```
CREATE TABLE `room` (
  `roomnum` int(11) NOT NULL,
  `rhotelid` int(11) NOT NULL,
  `price` int(11) DEFAULT NULL,
  `floor` int(11) DEFAULT NULL,
  `rtype` varchar(45) DEFAULT NULL,
  `description` varchar(45) DEFAULT NULL,
  `capacity` int(11) DEFAULT NULL,
  PRIMARY KEY (`roomnum`,`rhotelid`),
  KEY `rhotelid_idx` (`rhotelid`),
  CONSTRAINT `rhotelid` FOREIGN KEY (`rhotelid`) REFERENCES `hotel` (`hotelId`)
```

10.ROOM_RESERVATION

```
CREATE TABLE `room_reservation` (
  `Invoiceno` int(11) DEFAULT NULL,
  `hotelid` int(11) NOT NULL,
  `roomno` int(11) NOT NULL,
  `checkindate` date NOT NULL,
  `checkoutdate` date DEFAULT NULL,
  PRIMARY KEY (`hotelid`,`roomno`,`checkindate`),
  KEY `room_invoice_idx` (`Invoiceno`),
  KEY `room_roomnum_idx` (`roomno`),
  KEY `checkindate` (`checkindate`),
  CONSTRAINT `room_hid` FOREIGN KEY (`hotelid`) REFERENCES `room` (`rhotelid`),
  CONSTRAINT `room_invoice` FOREIGN KEY (`Invoiceno`) REFERENCES `reservation` (`invoice_no`),
  CONSTRAINT `room_roomnum` FOREIGN KEY (`roomno`) REFERENCES `room` (`roomnum`)
```

11.ROOM_REVIEW

```
CREATE TABLE `room_review` (
  `RID` int(11) NOT NULL AUTO_INCREMENT,
  `Rating` int(11) DEFAULT NULL,
  `text` varchar(45) DEFAULT NULL,
  `cid` varchar(45) DEFAULT NULL,
  `hotelid` int(11) DEFAULT NULL,
```

```
  `roomno` int(11) DEFAULT NULL,
  PRIMARY KEY (`RID`),
  KEY `rreview_hid_idx` (`hotelid`),
  KEY `rreview_roomno_idx` (`roomno`),
  CONSTRAINT `rreview_hid` FOREIGN KEY (`hotelid`) REFERENCES `room` (`rhotelid`),
  CONSTRAINT `rreview_roomno` FOREIGN KEY (`roomno`) REFERENCES `room` (`roomnum`)
```

12.RRSEV_SERVICE

```
CREATE TABLE `rrsev_service` (
  `s_rhotelid` int(11) NOT NULL,
  `s_rroomno` int(11) NOT NULL,
  `s_rcheckindate` date NOT NULL,
  `s_stype` varchar(45) DEFAULT NULL,
  KEY `s_rhid_idx` (`s_rhotelid`),
  KEY `r_srhid_idx` (`s_rhotelid`),
  KEY `r_sstype_idx` (`s_stype`),
  KEY `r_scheckin_idx` (`s_rcheckindate`),
  KEY `r_sroomno_idx` (`s_rroomno`),
  CONSTRAINT `r_scheckin` FOREIGN KEY (`s_rcheckindate`) REFERENCES `room_reservation`
(`checkindate`),
  CONSTRAINT `r_srhid` FOREIGN KEY (`s_rhotelid`) REFERENCES `services` (`shotelid`),
  CONSTRAINT `r_sroomno` FOREIGN KEY (`s_rroomno`) REFERENCES `room_reservation` (`roomno`),
  CONSTRAINT `r_sshid` FOREIGN KEY (`s_rhotelid`) REFERENCES `room_reservation` (`hotelid`),
  CONSTRAINT `r_sstype` FOREIGN KEY (`s_stype`) REFERENCES `services` (`stype`)
```

13.SERVICE_REVIEW

```
CREATE TABLE `service_review` (
  `SRID` int(11) NOT NULL,
  `Hotelid` int(11) DEFAULT NULL,
  `Stype` varchar(45) DEFAULT NULL,
  `cid` int(11) DEFAULT NULL,
  `rating` varchar(45) DEFAULT NULL,
  `text` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`SRID`),
  KEY `s_rhid_idx` (`Hotelid`),
  KEY `s_rstype_idx` (`Stype`),
  CONSTRAINT `s_rcid` FOREIGN KEY (`SRID`) REFERENCES `customer` (`c_id`),
  CONSTRAINT `s_rhid` FOREIGN KEY (`Hotelid`) REFERENCES `services` (`shotelid`),
  CONSTRAINT `s_rstype` FOREIGN KEY (`Stype`) REFERENCES `services` (`stype`)
```

14.SERVICES

```
CREATE TABLE `services` (
  `stype` varchar(50) NOT NULL,
  `shotelid` int(11) NOT NULL,
  `sprice` int(11) DEFAULT NULL,
  PRIMARY KEY (`stype`,`shotelid`),
  KEY `hotelid_idx` (`shotelid`),
  CONSTRAINT `shotelid` FOREIGN KEY (`shotelid`) REFERENCES `hotel` (`hotelld`)
```

Table Population

Partial SQL insert queries is used to populate tables in database.

```
-- Records of breakfast
-- ----------------------------
INSERT INTO `breakfast` VALUES ('0', 'American', '150', 'American'); INSERT INTO `breakfast` VALUES ('0',
'continental', '100', 'continental'); INSERT INTO `breakfast` VALUES ('0', 'English', '200', 'English');
INSERT INTO `breakfast` VALUES ('0', 'French', '230', 'French');
INSERT INTO `breakfast` VALUES ('0', 'Italian', '250', 'Italian');
INSERT INTO `breakfast` VALUES ('1', 'American', '150', 'American');

-- ----------------------------

-- Records of breakfast_review
-- ----------------------------
INSERT INTO `breakfast_review` VALUES ('3001', '1', 'American', '0', 'breakfast review', '7', '2017- 04-29
09:29:17');
INSERT INTO `breakfast_review` VALUES ('3002', '2', 'continental', '1', 'breakfast review', '10', '2017- 04-29
09:29:17');
INSERT INTO `breakfast_review` VALUES ('3003', '3', 'continental', '2', 'breakfast review', '7', '2017- 04-29
09:29:17');
INSERT INTO `breakfast_review` VALUES ('3004', '4', 'continental', '3', 'breakfast review', '9', '2017- 04-29
09:29:17');

-- ----------------------------



-- Records of credit_card
-- ----------------------------
INSERT INTO `credit_card` VALUES ('1017258780368178', 'Amex', 'address410', '535', '2030-04-03 00:00:00',
'c410', '410');
INSERT INTO `credit_card` VALUES ('1043210066621620', 'Discover', 'address450', '539', '2030-04- 03 00:00:00',
'c450', '450');
INSERT INTO `credit_card` VALUES ('1052634640785415', 'Discover', 'address389', '692', '2030-04- 03 00:00:00',
'c389', '389');
```

INSERT INTO `credit_card` VALUES ('1066565136632600', 'Master', 'address129', '456', '2030-04- 03 00:00:00', 'c129', '129');

-- ----------

-- Records of customer
-- ----------
INSERT INTO `customer` VALUES ('0', '2347820c@gmail.com', 'p0');
INSERT INTO `customer` VALUES ('1', '2347821c@gmail.com', 'p1');
INSERT INTO `customer` VALUES ('2', '2347822c@gmail.com', 'p2');
INSERT INTO `customer` VALUES ('3', '2347823c@gmail.com', 'p3');

-- ----------

-- Records of discount_room
-- ----------
INSERT INTO `discount_room` VALUES ('0', INSERT INTO `discount_room` VALUES ('0', INSERT INTO `discount_room` VALUES ('0', INSERT INTO `discount_room` VALUES ('0', INSERT INTO `discount_room` VALUES ('0',

-- ----------

'customer0', 'street0', '15626485430', 'customer1', 'street1', '15626485431', 'customer2', 'street2', '15626485432', 'customer3', 'street3', '15626485433',

-- Records of hotel
-- ----------
INSERT INTO `hotel` VALUES ('0', 'street0', 'country0', 'state0', '07029', 'hotel0'); INSERT INTO `hotel` VALUES ('1', 'street1', 'country1', 'state1', '07030', 'hotel1'); INSERT INTO `hotel` VALUES ('2', 'street2', 'country2', 'state2', '07031', 'hotel2'); INSERT INTO `hotel` VALUES ('3', 'street3', 'country3', 'state3', '07032', 'hotel3');

-- ----------

-- Records of manager
-- ----------
INSERT INTO `manager` VALUES ('admin', 'admin', 'Hulton Hotel Chain');

'1', '0.98', '2017-04-20', '2017-06-13'); '2', '0.99', '2017-08-08', '2017-10-16'); '3', '0.9', '2017-02-05', '2017-04-10'); '4', '0.8', '2017-10-24', '2017-12-06');

'5', '0.8400000000000001', '2017-04-14', '2017-06-23');

-- ----------

-- Records of phone
-- ----------
INSERT INTO `phone` VALUES ('0', '13911124555'); INSERT INTO `phone` VALUES ('0', '13917793333'); INSERT INTO `phone` VALUES ('0', '13927698906'); INSERT INTO `phone` VALUES ('1', '13956801052'); INSERT INTO `phone` VALUES ('1', '13957156215'); INSERT INTO `phone` VALUES ('1', '13959076272');

-- ----------

-- Records of reservation
-- ----------
INSERT INTO `reservation` VALUES ('2917205693201729119O', '12', '1400655115772624', '2017- 04-29 09:29:17.000000', '510');
INSERT INTO `reservation` VALUES ('29172124932017291466', '9', '7389753215178781', '2017-04- 29 09:29:17.000000', '560');
INSERT INTO `reservation` VALUES ('29172211932017291486', '0', '2534397781775065', '2017-04- 29 09:29:17.000000', '560');
INSERT INTO `reservation` VALUES ('29172234932017291127', '2', '8668629889848350', '2017-04- 29 09:29:17.000000', '621');

-- ----------

-- Records of room
-- ---------- INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES INSERT INTO `room` VALUES

-- ----------

('0', '1', 'standard', '300', '1', '1', 'standard'); ('0', '2', 'suite', '600', '2', '4', 'suite');
('0', '3', 'deluxe', '500', '3', '3', 'deluxe');
('0', '4', 'standard', '300', '4', '1', 'standard'); ('0', '5', 'standard', '300', '0', '1', 'standard'); ('1', '1', 'deluxe', '500', '1', '3', 'deluxe');

('1', '2', 'deluxe', '500', '2', '3', 'deluxe'); ('1', '3', 'suite', '600', '3', '4', 'suite');

-- Records of room_reservation
-- ----------
INSERT INTO `room_reservation` VALUES ('29172211932017291486', '1', '1', '2017-06-02 00:00:00', '2017-06-03 00:00:00');
INSERT INTO `room_reservation` VALUES ('29202655932017291009', '1', '2', '2017-06-06 00:00:00', '2017-06-07 00:00:00');
INSERT INTO `room_reservation` VALUES ('29232302932017291670', '1', '3', '2017-06-10 00:00:00', '2017-06-11 00:00:00');

INSERT INTO `room_reservation` VALUES ('29262213932017291044', '1', '4', '2017-06-14 00:00:00', '2017-06-15 00:00:00');
INSERT INTO `room_reservation` VALUES ('29282068932017291612', '1', '5', '2017-06-18 00:00:00', '2017-06-19 00:00:00');

-- ----------

-- Records of room_review
-- ----------
INSERT INTO `room_review` VALUES ('3001', '1', 'roomreview', '0', '1', '1', '2017-04-29 09:29:17'); INSERT INTO `room_review` VALUES ('3002', '9', 'roomreview', '1', '2', '2', '2017-04-29 09:29:17'); INSERT INTO `room_review` VALUES ('3003', '10', 'roomreview', '2', '3', '3', '2017-04-29 09:29:17'); INSERT INTO `room_review` VALUES ('3004', '10', 'roomreview', '3', '4', '4', '2017-04-29 09:29:17'); INSERT INTO `room_review` VALUES ('3005', '7', 'roomreview', '4', '5', '5', '2017-04-29 09:29:17');

-- ----------

-- Records of rresv_breakfast
-- ----------
INSERT INTO `rresv_breakfast` VALUES ('2', 'American', '6', '2017-06-18 00:00:00', '1'); INSERT INTO `rresv_breakfast` VALUES ('3', 'American', '6', '2017-06-14 00:00:00', '1'); INSERT INTO `rresv_breakfast` VALUES ('4', 'American', '4', '2017-06-02 00:00:00', '1'); INSERT INTO `rresv_breakfast` VALUES ('5', 'American', '1', '2017-06-22 00:00:00', '1'); INSERT INTO `rresv_breakfast` VALUES ('5', 'American', '9', '2017-06-18 00:00:00', '1');

-- ----------

-- Records of rresv_service
-- ----------
INSERT INTO `rresv_service` VALUES ('1', 'parking', '1', '2017-06-02 00:00:00');
INSERT INTO `rresv_service` VALUES ('1', 'airport-drop-off', '2', '2017-06-06 00:00:00'); INSERT INTO `rresv_service` VALUES ('1', 'airport-drop-off', '3', '2017-06-10 00:00:00'); INSERT INTO `rresv_service` VALUES ('1', 'airport-drop-off', '4', '2017-06-14 00:00:00'); INSERT INTO `rresv_service` VALUES ('1', 'airport-pick-up', '5', '2017-06-18 00:00:00'); INSERT INTO `rresv_service` VALUES ('1', 'airport-pick-up', '6', '2017-06-22 00:00:00');

-- ----------

-- Records of service
-- ----------
INSERT INTO `service` VALUES ('0', 'airport-drop-off', '21'); INSERT INTO `service` VALUES ('0', 'airport-pick-up', '21'); INSERT INTO `service` VALUES ('0', 'laundry', '13');
INSERT INTO `service` VALUES ('0', 'parking', '10');
INSERT INTO `service` VALUES ('1', 'airport-drop-off', '21'); INSERT INTO `service` VALUES ('1', 'airport-pick-up', '21'); INSERT INTO `service` VALUES ('1', 'laundry', '13');
INSERT INTO `service` VALUES ('1', 'parking', '10');

-- ----------

-- Records of service_review
-- ----------
INSERT INTO `service_review` VALUES ('3001', '1', 'airport-pick-up', '0', '4', 'service review', '2017- 04-29 09:29:17');
INSERT INTO `service_review` VALUES ('3002', '2', 'parking', '1', '5', 'service review', '2017-04-29 09:29:17');
INSERT INTO `service_review` VALUES ('3003', '3', 'airport-drop-off', '2', '3', 'service review', '2017- 04-29 09:29:17');
INSERT INTO `service_review` VALUES ('3004', '4', 'laundry', '3', '5', 'service review', '2017-04-29 09:29:17');
INSERT INTO `service_review` VALUES ('3005', '5', 'airport-pick-up', '4', '2', 'service review', '2017- 04-29 09:29:17');
INSERT INTO `service_review` VALUES ('3006', '6', 'parking', '5', '9', 'service review', '2017-04-29 09:29:17');

Other Important Queries

1. BEST-BREAKFAST

public ResultSet queryBestBreakfast(Date start,Date end) {

String sql = "**select b_type, avg(rating) from breakfast_review where (b_type,rating) in (select b_type,rating from breakfast_review where rdate BETWEEN ? and ?) GROUP BY b_type ORDER BY avg(rating) DESC**";

return execSQL(sql,start,end);

}

2. BEST-ROOM

public ResultSet queryBestRoom(Date start,Date end,Integer hotelId) {

String sql = "**select hotel_id,rtype,avg(rating) from (room_review natural join room) where (rtype,rating,hotel_id) in (select rtype,rating,hotel_id from (room_review NATURAL JOIN room) where hotel_id=? and rdate BETWEEN ? and ?) GROUP BY rtype ORDER BY avg(rating) DESC**";

return execSQL(sql,hotelId,start,end);

}

3.FIVE-BEST CUSTOMERS

public ResultSet queryFiveBestCustomer(Date start,Date end) {

String sql = "**select c_Id,name,sum(amount),reservation.r_date from(customer NATURAL JOIN reservation) group by c_Id having reservation.r_date BETWEEN ? and ? order by sum(amount) desc** ";

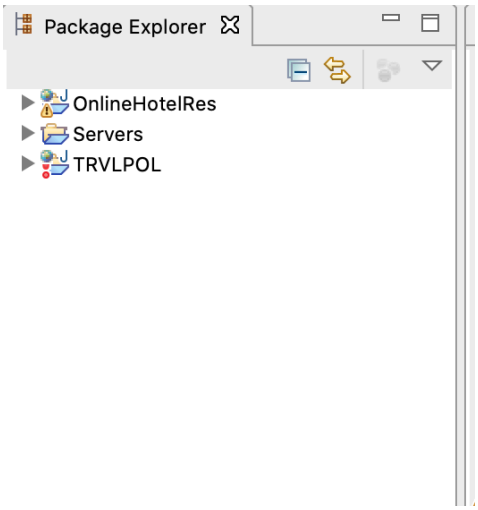
return execSQL(sql,start,end);

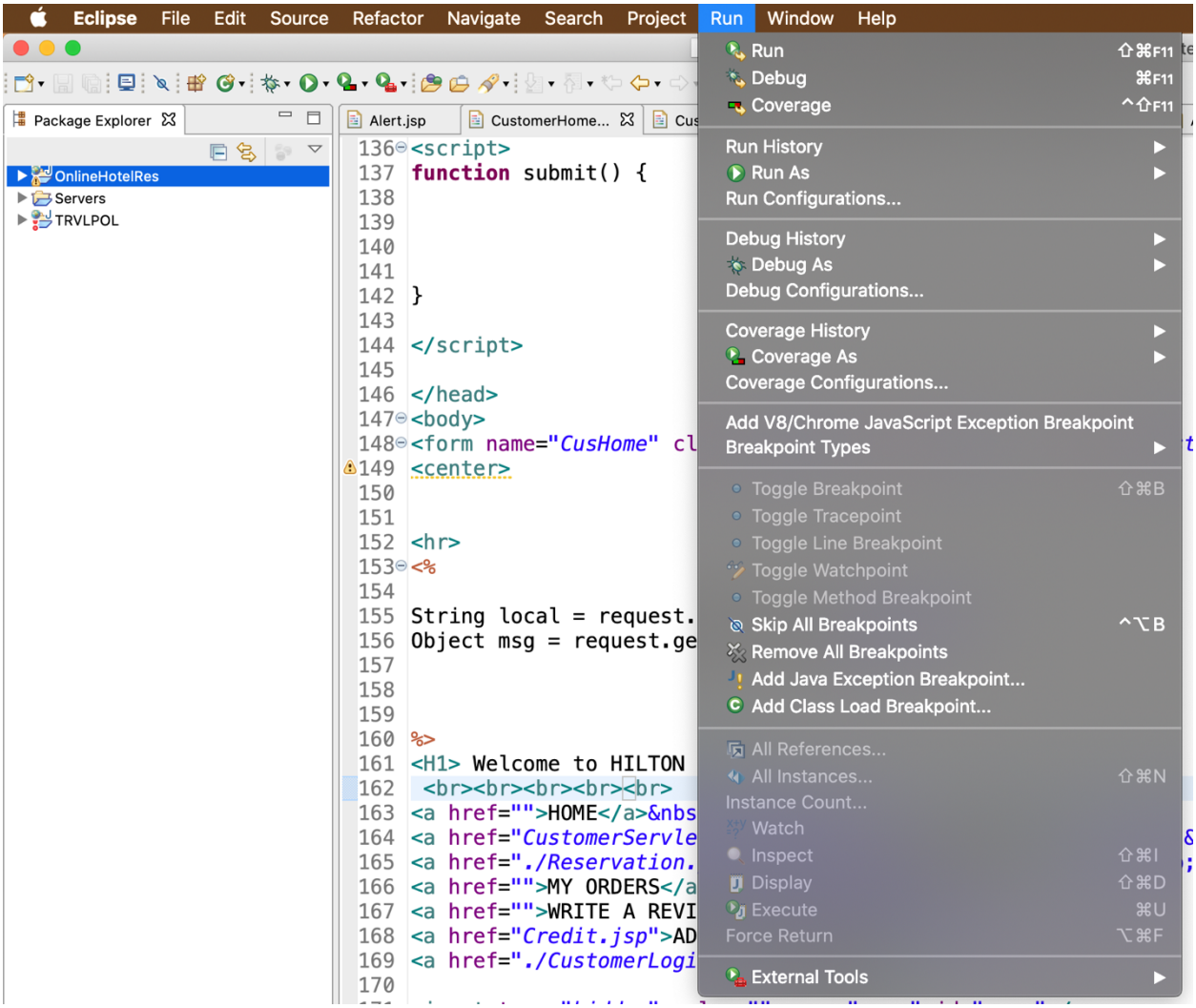
}

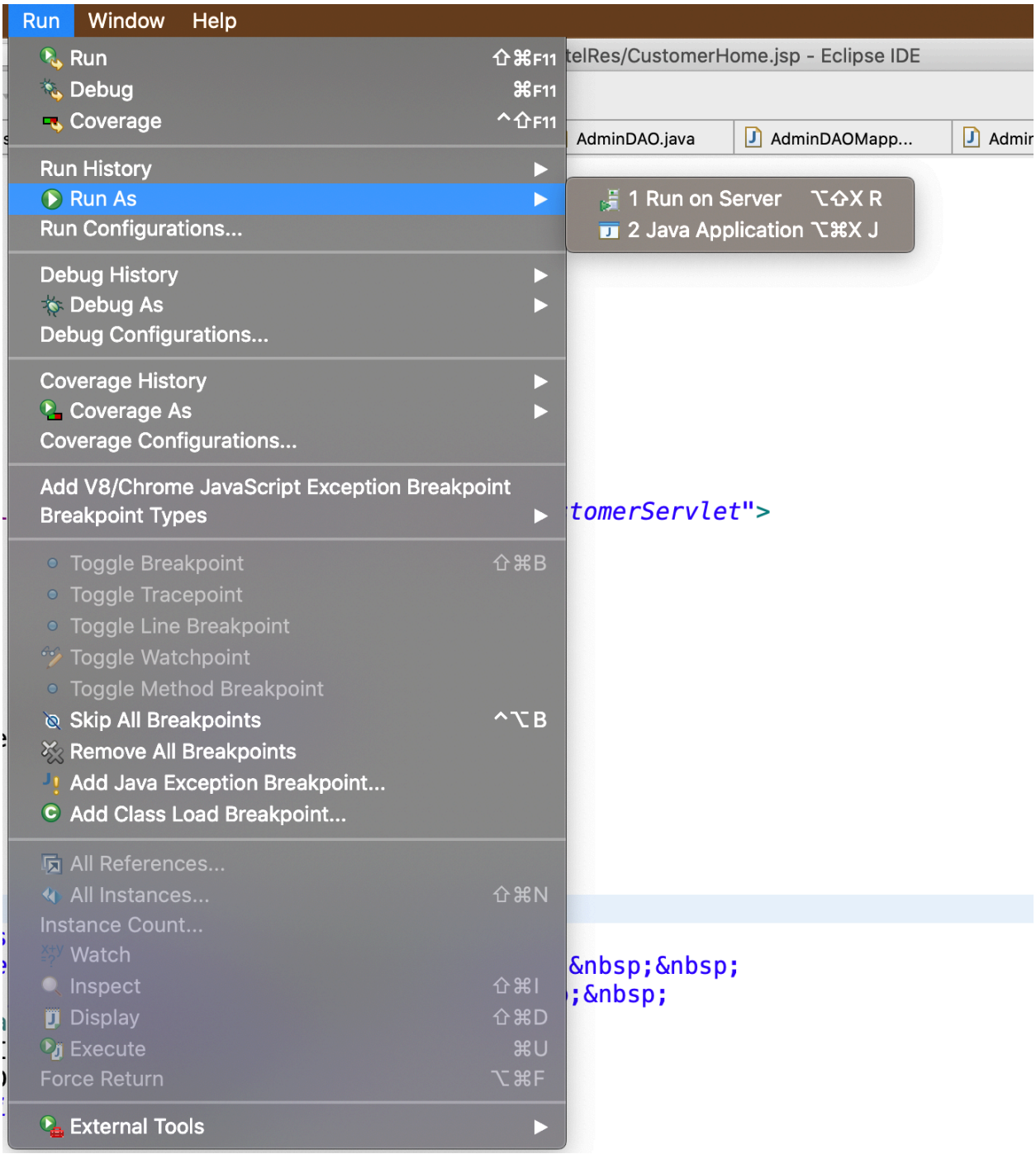
4. BEST SERVICE

```
public ResultSet queryBestService(Date start,Date end) {

        String sql = "select s_type,avg(rating) from service_review where (s_type,rating) in (select s_type,rating from service_review where rdate BETWEEN ? and ?) GROUP BY s_type ORDER BY avg(rating) DESC";

        return execSQL(sql,start,end);

}
```

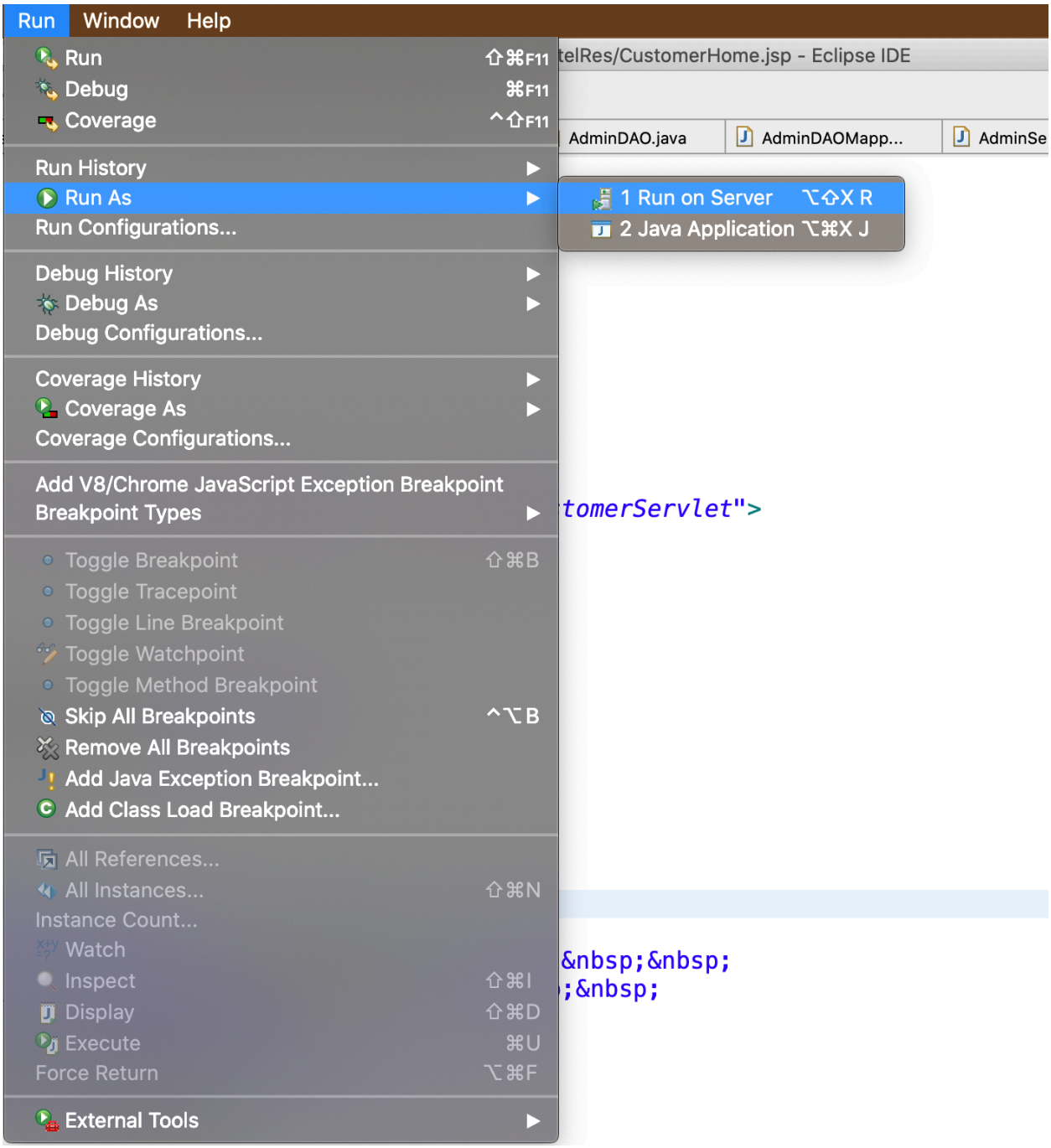
User Guide to Run the Application





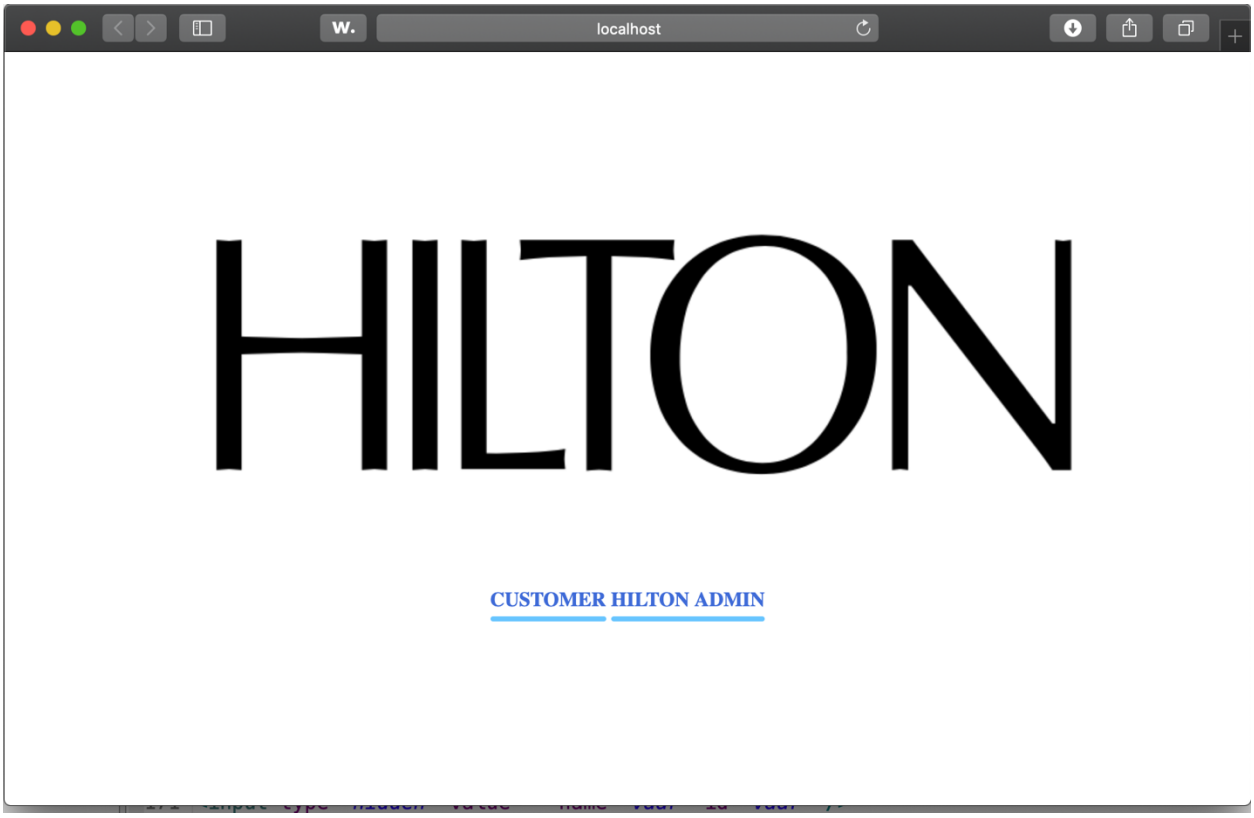

Run
Window
Help
Run ⇧⌘F11
Debug ⌘F11
Coverage ^⇧F11
Run History
Run As
Run Configurations...
Debug History
Debug As
Debug Configurations...
Coverage History
Coverage As
Coverage Configurations...
Add V8/Chrome JavaScript Exception Breakpoint
Breakpoint Types
Toggle Breakpoint ⇧⌘B
Toggle Tracepoint
Toggle Line Breakpoint
Toggle Watchpoint
Toggle Method Breakpoint
Skip All Breakpoints ^⌥B
Remove All Breakpoints
Add Java Exception Breakpoint...
Add Class Load Breakpoint...
All References...
All Instances... ⇧⌘N
Instance Count...
Watch
Inspect ⇧⌘I
Display ⇧⌘D
Execute ⌘U
Force Return ⌥⌘F
External Tools
telRes/CustomerHome.jsp - Eclipse IDE
AdminDAO.java
AdminDAOMapp...
Admir
1 Run on Server ⌥⇧X R
2 Java Application ⌥⌘X J
tomerServlet">

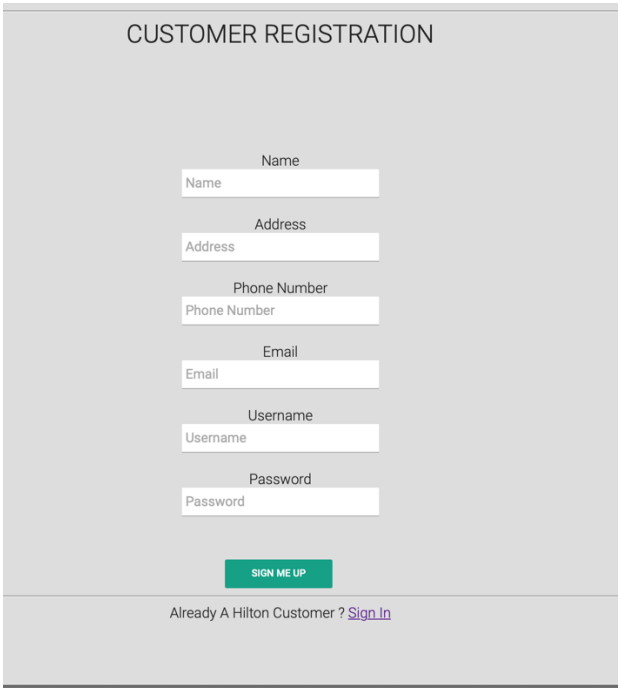
;

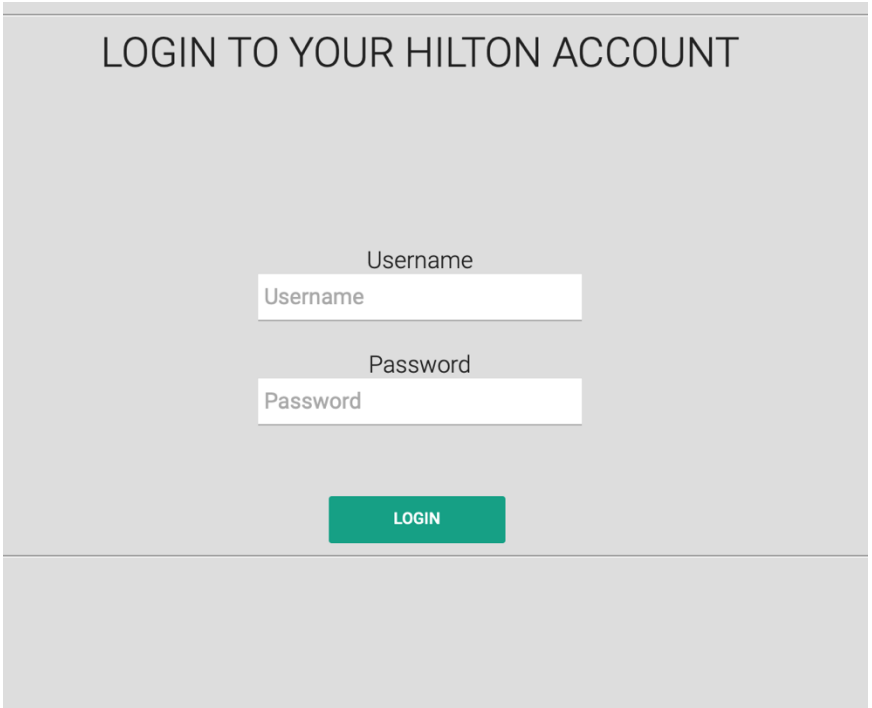

Run
Window
Help
Run ⇧⌘F11
Debug ⌘F11
Coverage ^⇧F11
Run History
Run As
Run Configurations...
Debug History
Debug As
Debug Configurations...
Coverage History
Coverage As
Coverage Configurations...
Add V8/Chrome JavaScript Exception Breakpoint
Breakpoint Types
Toggle Breakpoint ⇧⌘B
Toggle Tracepoint
Toggle Line Breakpoint
Toggle Watchpoint
Toggle Method Breakpoint
Skip All Breakpoints ^⌥B
Remove All Breakpoints
Add Java Exception Breakpoint...
Add Class Load Breakpoint...
All References...
All Instances... ⇧⌘N
Instance Count...
Watch
Inspect ⇧⌘I
Display ⇧⌘D
Execute ⌘U
Force Return ⌥⌘F
External Tools
1 Run on Server ⌥⇧X R
2 Java Application ⌥⌘X J
telRes/CustomerHome.jsp - Eclipse IDE
AdminDAO.java
AdminDAOMapp...
AdminSe
tomerServlet">

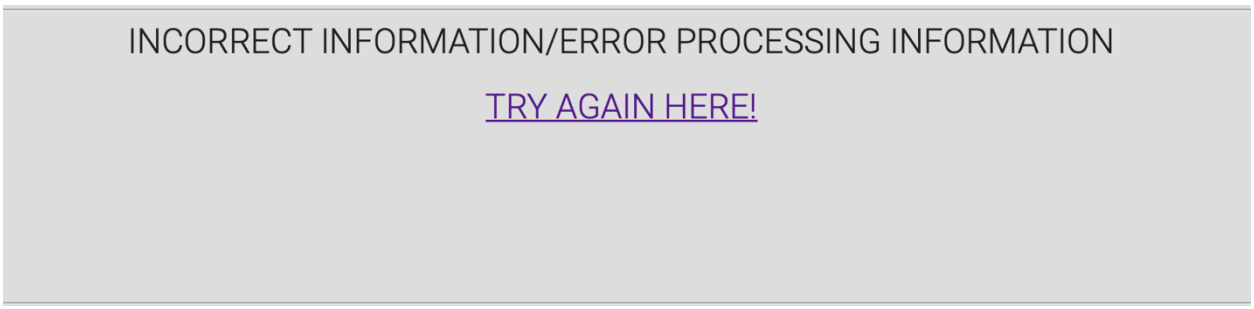
;

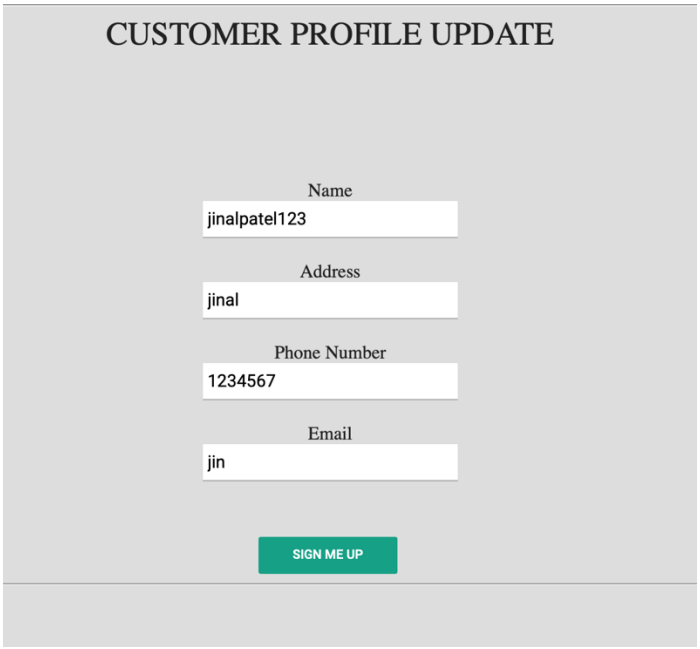
Application Flow of User

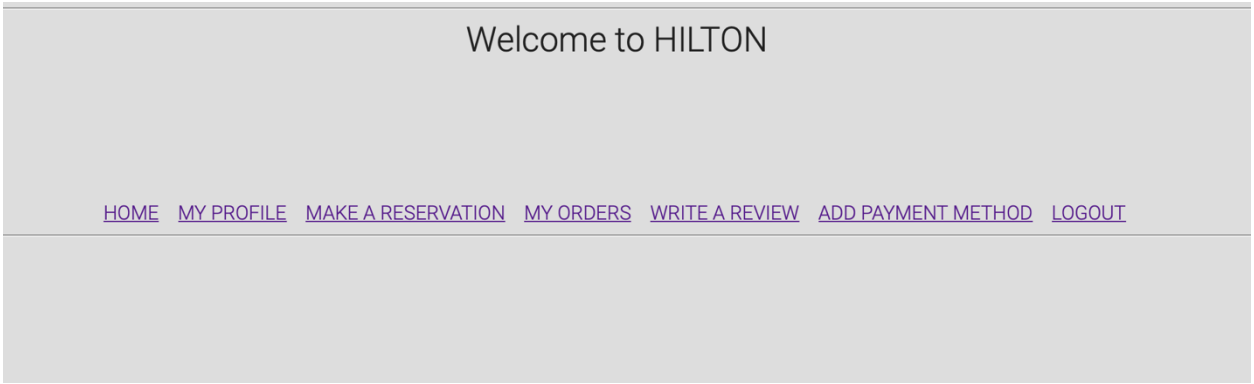


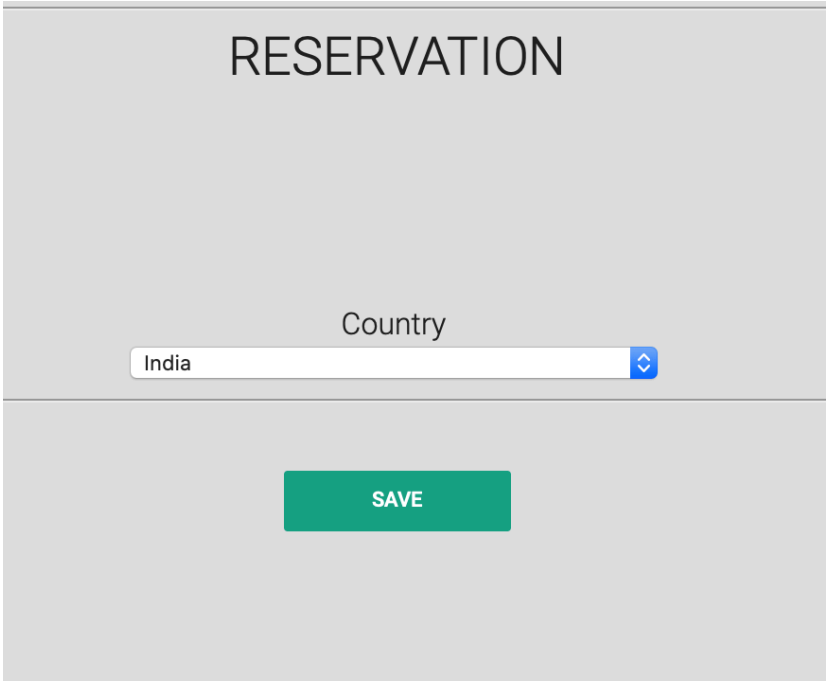



LOGIN TO YOUR HILTON ACCOUNT
Username
Username
Password
Password
LOGIN


INCORRECT INFORMATION/ERROR PROCESSING INFORMATION
TRY AGAIN HERE!


CUSTOMER PROFILE UPDATE
Name
jinalpatel123
Address
jinal
Phone Number
1234567
Email
jin
SIGN ME UP


Welcome to HILTON
HOME
MY PROFILE
MAKE A RESERVATION
MY ORDERS
WRITE A REVIEW
ADD PAYMENT METHOD
LOGOUT


RESERVATION
Country
India
SAVE

Welcome to HILTON

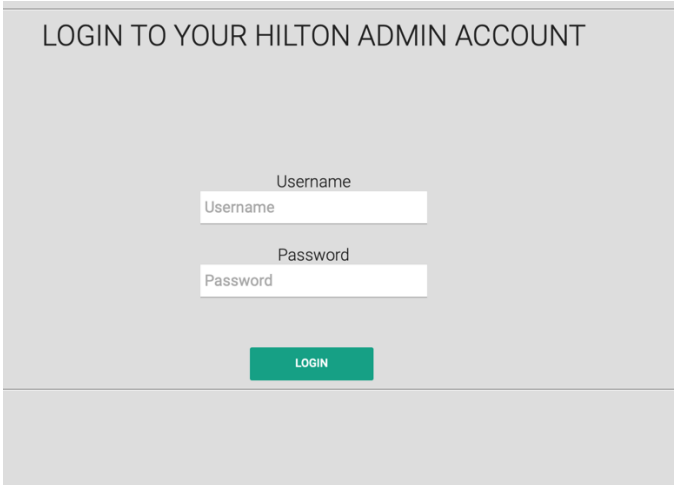
Hulton at Ahemdabad Hulton at Maharashtra

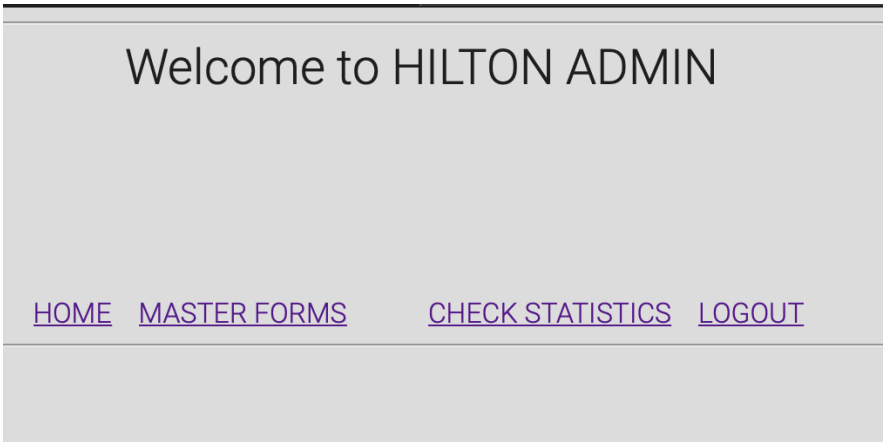
Registration

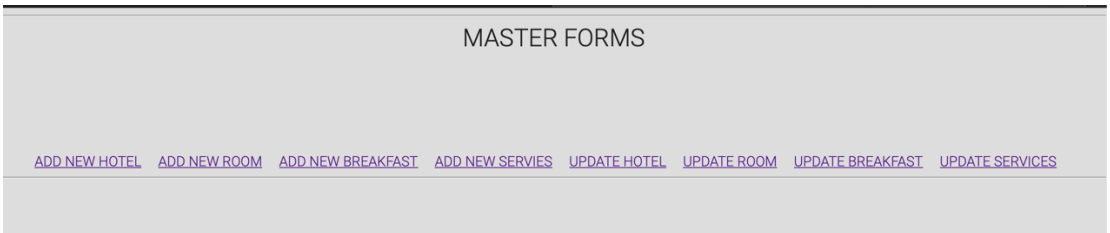
Name:
Address:
PhoneNumber:
Email:
UserName:
Password:

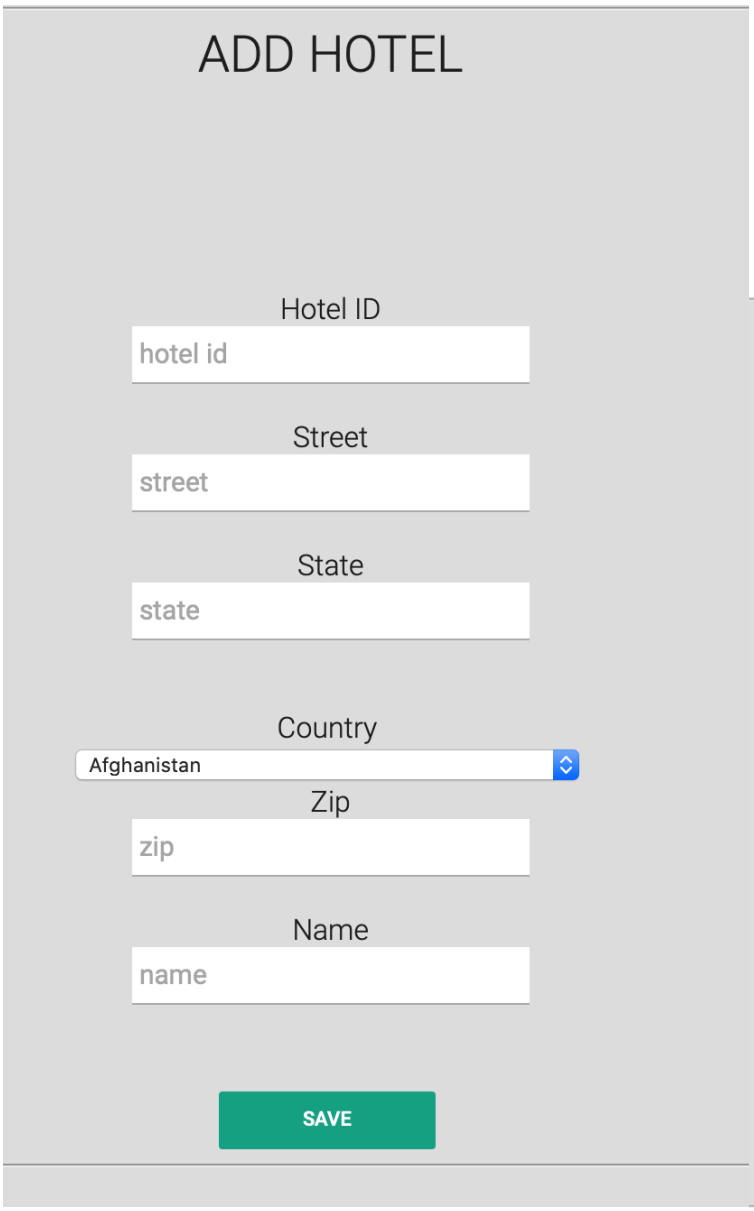
GO TO LOGIN | Register

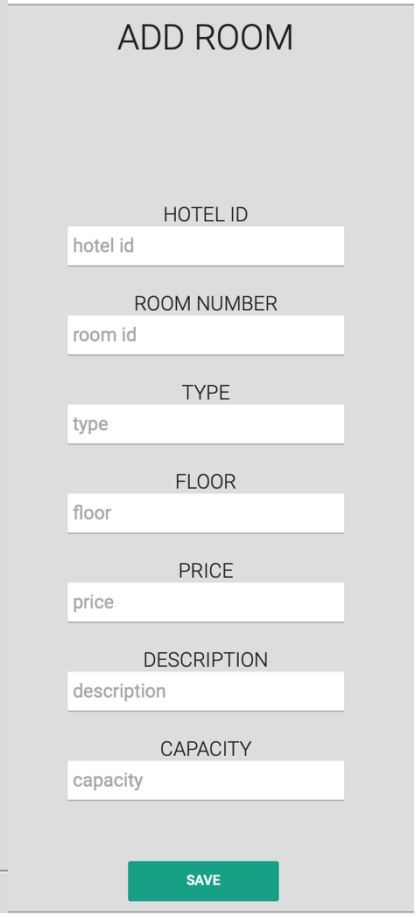
Application Flow of Admin

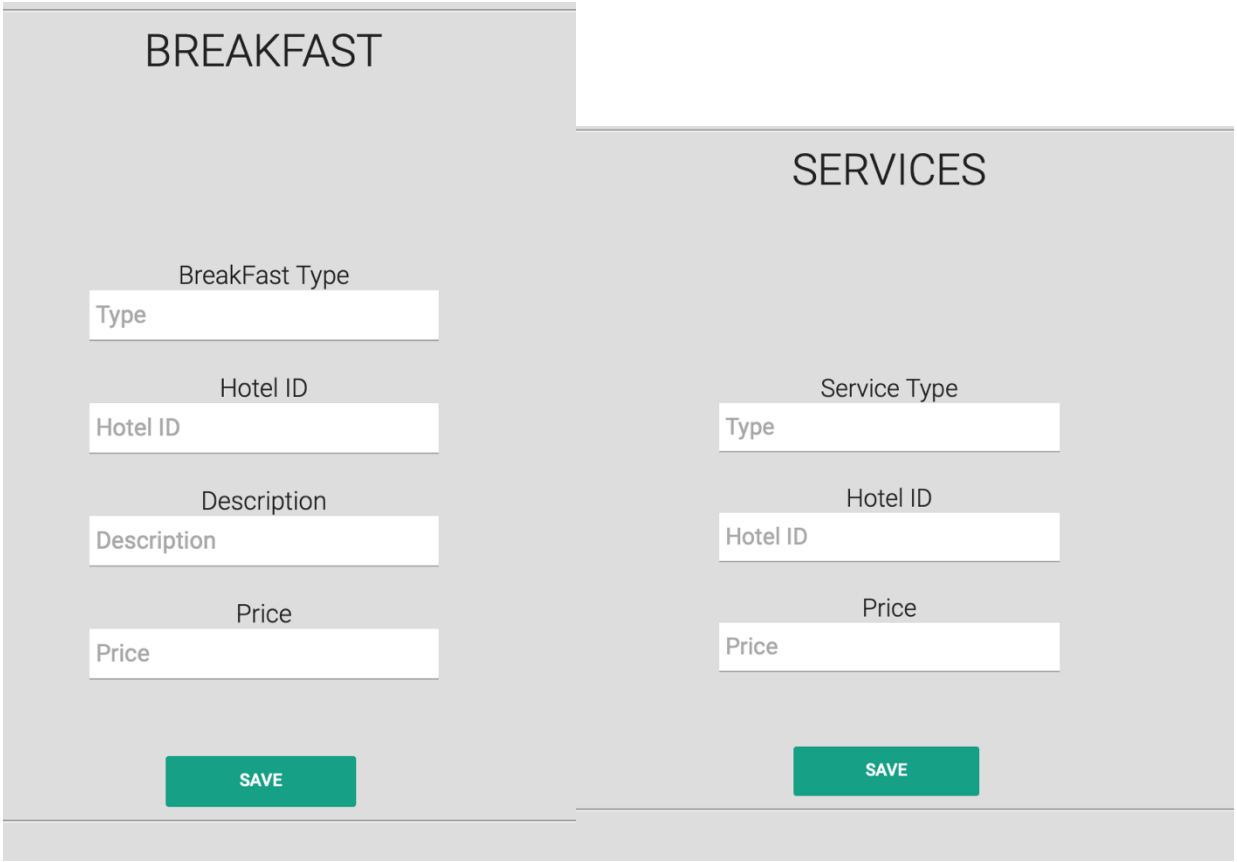






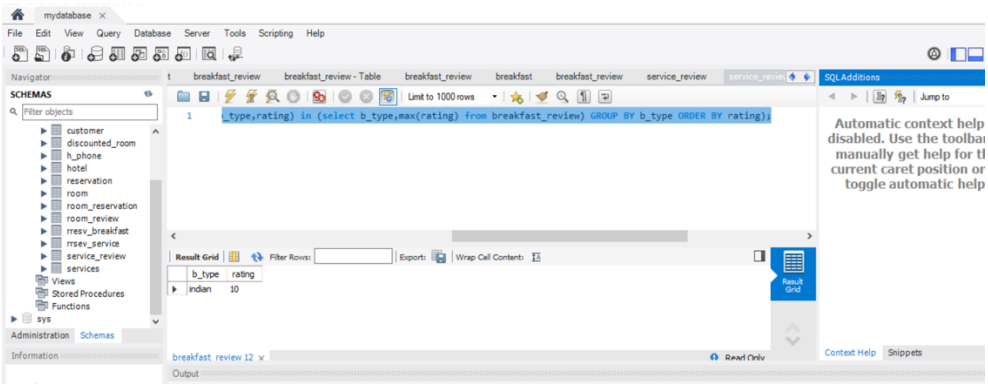

ADD HOTEL
Hotel ID
hotel id
Street
street
State
state
Country
Afghanistan
Zip
zip
Name
name
SAVE


ADD ROOM
HOTEL ID
hotel id
ROOM NUMBER
room id
TYPE
type
FLOOR
floor
PRICE
price
DESCRIPTION
description
CAPACITY
capacity
SAVE

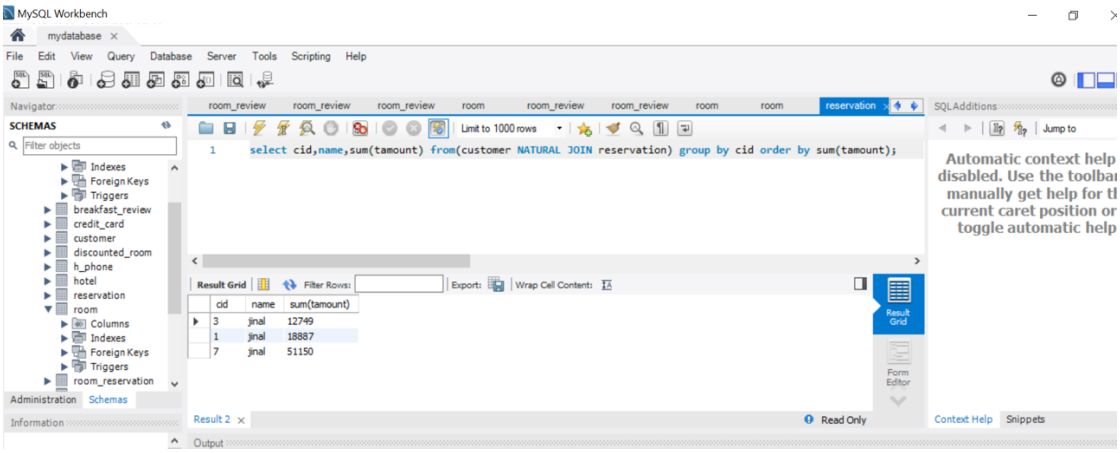

BREAKFAST
BreakFast Type
Type
Hotel ID
Hotel ID
Description
Description
Price
Price
SAVE
SERVICES
Service Type
Type
Hotel ID
Hotel ID
Price
Price
SAVE

Imp Queries :

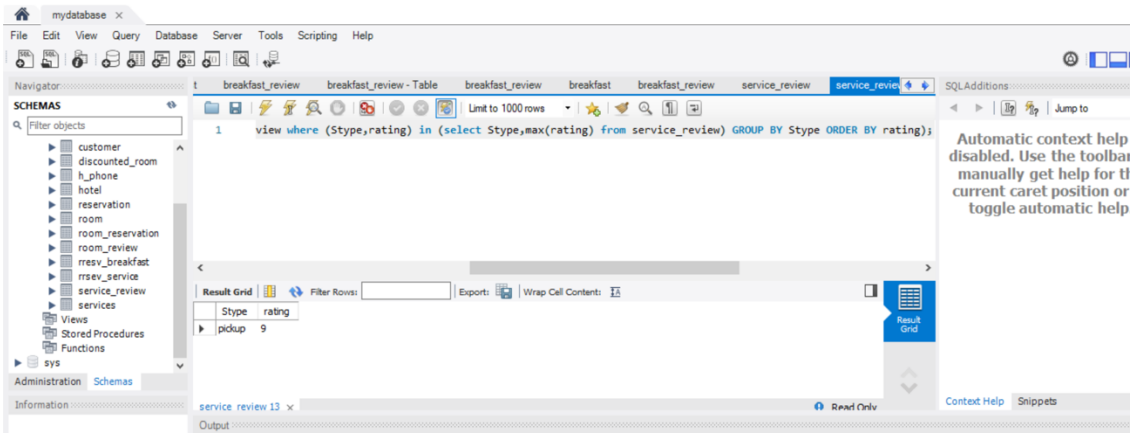
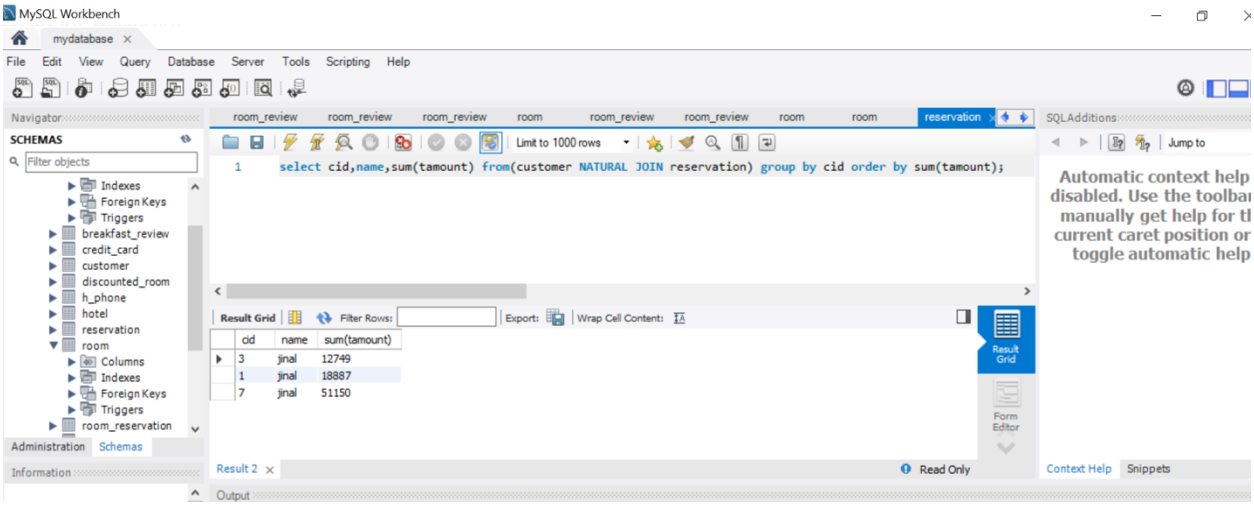
Top Breakfast



Top Customer



Top Service



Top 5 customers