**Final Project SQL Queries**

Team 6

Chris Altonji, caltonji3, caltonji@gatech.edu

Alison McGarey, amcgarey3,  amcgarey3@gatech.edu

Omar Mujahid, omujahid3,  omar\_mjhd@gatech.edu

Kelcy Newton, knewton6,  knewton6@gatech.edu

All team members are in CS 4400-B

**Key:**

**Query Title**

QUERY

**VARIABLE IN QUERY**

**Queries**

**Find a Customer:**

SELECT \* FROM CUSTOMER WHERE Username = "**USERNAME**”;

**Find a Manager:**

SELECT \* FROM MANAGEMENT WHERE Username = "**USERNAME**”;

**Create a new Customer:**

INSERT INTO CUSTOMER ( Username, Password, Email ) VALUES ("**USERNAME**", "**PASSWORD**", "**EMAIL**");

**Add Payment Information:**

INSERT INTO PAYMENT\_INFORMATION ( Name, Exp\_date, CVV, Card\_no, Username ) VALUES ("**NAME**","**EXP\_DATE**"," **CVV**","**CARD\_NO** ",”**USERNAME**");

**Find Payment Information:**

SELECT \* FROM PAYMENT\_INFORMATION WHERE Username = “**USERNAME**";

**Delete Payment Information:**

DELETE FROM PAYMENT\_INFORMATION WHERE Username = "**USERNAME**" AND Card\_no = " **CARD\_NO** ";

**Find Reservation:**

SELECT \* FROM RESERVATION WHERE Reservation\_ID = "**RESERVATION\_ID**" AND username = “**USERNAME** ";

**Create Reservation:**

**(Due to the nature of this query, we decided to include the whole function that generates the query)**

exports.createReservation = function(startDate, endDate, totalCost, isCancelled, cardNo, username, roomArray) { //roomArray = location, room\_No, Extra\_bed

var query = "INSERT INTO RESERVATION ( Start\_date, End\_date, Total\_cost, Is\_cancelled, Card\_no, Username ) VALUES ("

+ mysql.escape(startDate) + "," + mysql.escape(endDate) + "," + mysql.escape(totalCost) + "," + mysql.escape(isCancelled)

+ "," + mysql.escape(cardNo) + "," + mysql.escape(username) + ");";

query = query + "SET @reservation\_id = (SELECT MAX(Reservation\_ID) FROM RESERVATION);";

query = query + "SELECT @reservation\_id;";

query = query + " INSERT INTO HAS\_ROOM (Reservation\_ID, Extra\_bed, Room\_no, location) VALUES (@reservation\_id, ";

for (var i = 0; i < roomArray.length; i++) {

var room = roomArray[i];

query = query + mysql.escape(room.Extra\_bed) + ", " + mysql.escape(room.Room\_no) + ", " + mysql.escape(room.location)+ "); ";

if (i != roomArray.length - 1) {

query = query + " INSERT INTO HAS\_ROOM (Reservation\_ID, Extra\_bed, Room\_no, location) VALUES (@reservation\_id, ";

}

}

query = query + "SELECT @reservation\_id;";

return query;

}

**Cancel Reservation:**

UPDATE RESERVATION SET Is\_cancelled = "**1**", Cancel\_date = CURDATE() WHERE Reservation\_Id = "**RESERVATION\_ID**" AND username = "**USERNAME**";

**Update Reservation:**

UPDATE RESERVATION SET Start\_date = "**START\_DATE**", End\_date = "**END\_DATE**", Total\_cost = "**COST**" WHERE Reservation\_ID = "**Reservation\_ID**" AND username = "**USERNAME**";

**Find Reviews:**

SELECT Rating, Comment FROM HOTEL\_REVIEW WHERE Location = " **LOCATION**" ORDER BY Review\_no DESC;

**Create Review No Comment:**

INSERT INTO HOTEL\_REVIEW ( Rating, Location, Username ) VALUES ("**RATING**","

**LOCATION** ","**USERNAME**");

**Create Review With Comment:**

INSERT INTO HOTEL\_REVIEW ( Comment, Rating, Location, Username ) VALUES ("**COMMENT**","**RATING**",”**LOCATION**","**USERNAME** ");

**Search Rooms:**

**(Due to the nature of how this query is built, we included the entire function for clarity)**

exports.searchRooms = function(roomArray) { //TODO: test this?

var query = "SELECT \* FROM ROOM WHERE (";

for (var i = 0; i < roomArray.length; i++) {

var room = roomArray[i];

query = query + " location = " + mysql.escape(room.location) + " AND Room\_no = " + mysql.escape(room.Room\_no) + ")";

if (i != roomArray.length - 1) {

query = query + " OR (";

} else {

query = query + ";";

}

}

return query;

}

**search Available Rooms:**

SELECT \* FROM ROOM WHERE LOCATION = "**LOCATION**" AND NOT EXISTS (SELECT Room\_no FROM HAS\_ROOM NATURAL JOIN RESERVATION WHERE ROOM.Room\_no = HAS\_ROOM.Room\_no AND ROOM.location = HAS\_ROOM.location AND RESERVATION.Is\_cancelled = "**0**" AND (( "**START\_DATE**" <= End\_date AND "**END\_DATE**" >= End\_date)));

**Search Available Update:**

SELECT \* FROM ROOM NATURAL JOIN HAS\_ROOM WHERE ROOM\_NO = "**ROOM\_NO**" AND LOCATION = "**LOCATION** " AND Reservation\_ID= "**RESERVATION\_ID**" AND NOT EXISTS (SELECT Room\_no, Extra\_bed FROM HAS\_ROOM NATURAL JOIN RESERVATION WHERE ROOM.Room\_no = HAS\_ROOM.Room\_no AND ROOM.location = HAS\_ROOM.location AND RESERVATION.Is\_cancelled= 0 AND RESERVATION.Reservation\_ID != "**RESERVATION\_ID**" AND ((Start\_date <= "**START\_DATE**" AND "**START\_DATE**" <= End\_date) OR (Start\_date <= "**END\_DATE**"AND "**END\_DATE**"<= End\_date) OR "**START\_DATE**" <= Start\_date AND End\_date <= "**END\_DATE**")));

**Search Rooms By ID:**

SELECT Room\_no, Location, No\_people, Cost\_of\_extra\_bed\_per\_day, Room\_category, Cost\_day, Extra\_bed FROM ROOM NATURAL JOIN HAS\_ROOM WHERE Reservation\_Id = "**RESERVATION\_ID**";

**Create Reservation Report:**

SELECT location, COUNT( \* ) AS location\_count FROM RESERVATION NATURAL JOIN HAS\_ROOM WHERE MONTH(Start\_date) = "**MONTH\_NUMBER**" AND Is\_cancelled = "**0**" GROUP BY location;

**Create Popular Room Report:**

SELECT location, Room\_category, MAX(sub.counter) AS total\_rooms FROM (SELECT location, Room\_category, COUNT( \* ) AS counter FROM ROOM NATURAL JOIN RESERVATION NATURAL JOIN HAS\_ROOM WHERE MONTH( Start\_date ) = "**MONTH\_NUMBER**" AND Is\_cancelled = "**0**" GROUP BY location, Room\_category) AS sub GROUP BY location;

**Create Monthly Report:**

SELECT "**MONTH\_NUMBER**" AS Month, "**CITY** " AS City, IFNULL(NOREFUND.cost,"**0**") + IFNULL(PARTIAL.cost, "**0**") + IFNULL(BASE.cost,"**0**") AS Revenue FROM (SELECT SUM( Total\_cost ) AS cost FROM ROOM NATURAL JOIN RESERVATION NATURAL JOIN HAS\_ROOM WHERE MONTH( Start\_date ) = "**MONTH\_NUMBER**" AND Is\_cancelled = "**0**" AND Location ="**CITY**")AS BASE, (SELECT SUM( Total\_cost ) \*"**0.2**" AS cost FROM ROOM NATURAL JOIN RESERVATION NATURAL JOIN HAS\_ROOM WHERE MONTH( Start\_date ) = "**MONTH\_NUMBER**" AND Is\_cancelled = “**1**" AND DATEDIFF( Start\_date, Cancel\_date ) ="**2**" OR "**3**" AND Location ="**CITY**") AS PARTIAL, (SELECT SUM( Total\_cost ) AS cost FROM ROOM NATURAL JOIN RESERVATION NATURAL JOIN HAS\_ROOM WHERE MONTH( Start\_date ) = "**MONTH\_NUMBER**" AND Is\_cancelled = "**1**" AND DATEDIFF( Start\_date, Cancel\_date ) < "**2**" AND Location =" **CITY**") AS NOREFUND;