

## SQL QUERIES

/\*LOGIN\*/

/\*Find if a given username and password combination is in the database. \*/

/\*Inputs to the database are given by quotations and a + symbol \*/

```
SELECT * FROM User WHERE Username = '+username' AND  
Password='+password+';
```

/\*Determine if that user is a manager or customer based on the username.\*/

```
SELECT * FROM Customer WHERE Username='+username+';
```

```
SELECT * FROM Manager WHERE Username='+username+';
```

/\*NEW USER REGISTRATION

Check if Username is taken.\*/

```
SELECT COUNT(Username) FROM User WHERE  
Username='+username+';
```

/\*Insert new user information into database\*/

```
INSERT INTO User(Username, Password) VALUES  
('+username+', '+password+');
```

```
INSERT INTO Customer(Username, EmailAddress)
VALUES('+username+', '+EmailAddress+');
```

```
/*Add School Info
```

Update the boolean flag to show that the user is a student if the email entered ends in '.edu'\*/

```
UPDATE Customer SET IsStudent = 1 WHERE
Username='+username+';
```

```
/*View Train Schedule:
```

Get information for all of the routes that a user-inputted train number takes.\*/

```
SELECT Takes.TrainNumber AS `Train Number`,
Route.ArrivalTime AS `Arrival Time`, Route.DepartureTime AS
`Departure Time`, Route.StationName AS Station
FROM Route INNER JOIN Takes
ON Takes.RouteNum = Route.RouteNum WHERE
TrainNumber='+train_number+';
```

```
/*Customers can SearchTrain/Select Departure:*/
```

```
SELECT Arrival_Station_Routes.TrainNumber AS Train,
concat(Departure_Station_Routes.DepartureTime, '-',
Arrival_Station_Routes.ArrivalTime, ' ',
```

```

Hour(TIMEDIFF(Arrival_Station_Routes.ArrivalTime,
Departure_Station_Routes.DepartureTime)), 'hr', ' ',
Minute(TIMEDIFF(Arrival_Station_Routes.ArrivalTime,
Departure_Station_Routes.DepartureTime)), 'min')
AS Time, Arrival_Station_Routes.1stClassPrice,
Arrival_Station_Routes.2ndClassPrice
FROM
(SELECT Train.TrainNumber, Route.ArrivalTime,
Route.DepartureTime, Train.1stClassPrice, Train.2ndClassPrice
FROM Route INNER JOIN Takes ON Route.RouteNum =
Takes.RouteNum
INNER JOIN Train ON Takes.TrainNumber = Train.TrainNumber
AND Route.StationName = '+Arrival Station+')
Arrival_Station_Routes,
(SELECT Train.TrainNumber, Route.ArrivalTime,
Route.DepartureTime, Train.1stClassPrice, Train.2ndClassPrice
FROM Route INNER JOIN Takes INNER JOIN Train
ON Route.RouteNum = Takes.RouteNum AND
Takes.TrainNumber = Train.TrainNumber AND
Route.StationName = '+Departure Station+')
Departure_Station_Routes
WHERE Arrival_Station_Routes.TrainNumber =
Departure_Station_Routes.TrainNumber;

/*Users can Select how many Bags they will take & other
Passenger Info:*/

```

```
SELECT MaxBags, FreeBags FROM System;
```

```
/*Creating the Reservation
```

```
If First Ticket*/
```

```
INSERT INTO Reservation (Username) VALUES ('+username+');
```

```
INSERT INTO Ticket VALUES ('+TicketID+', '+ReservationID+',  
' + NumBags+', '+DepartureDate+',  
' + PassengerName+', '+ArrivalRouteNum+', '+DepartureRouteNu  
m+', '+Class+', '+Price+');
```

```
/*If not the first ticket for the reservation */
```

```
SELECT Arrival_Station_Routes.TrainNumber AS Train,
```

```
concat(Departure_Station_Routes.DepartureTime, '-',
```

```
Arrival_Station_Routes.ArrivalTime, ' ',
```

```
Hour(TIMEDIFF(Arrival_Station_Routes.ArrivalTime,
```

```
Departure_Station_Routes.DepartureTime)), 'hr', ' ',
```

```
Minute(TIMEDIFF(Arrival_Station_Routes.ArrivalTime,
```

```
Departure_Station_Routes.DepartureTime)), 'min')
```

```
AS Time, Arrival_Station_Routes.1stClassPrice,
```

```
Arrival_Station_Routes.2ndClassPrice
```

```
FROM
```

```
(SELECT Train.TrainNumber, Route.ArrivalTime,
```

```
Route.DepartureTime, Train.1stClassPrice, Train.2ndClassPrice
```

```
FROM Route INNER JOIN Takes ON Route.RouteNum =
```

```
Takes.RouteNum INNER JOIN Train ON Takes.TrainNumber =
```

```
Train.TrainNumber
```

```

AND Route.StationName ='Chicago(CHI)')
Arrival_Station_Routes, (SELECT Train.TrainNumber,
Route.ArrivalTime, Route.DepartureTime, Train.1stClassPrice,
Train.2ndClassPrice

FROM Route INNER JOIN Takes INNER JOIN Train ON
Route.RouteNum = Takes.RouteNum AND Takes.TrainNumber
= Train.TrainNumber AND Route.StationName = 'Atlanta(ATL)')
Departure_Station_Routes

WHERE Arrival_Station_Routes.TrainNumber =
Departure_Station_Routes.TrainNumber;

```

/\* Finding the Total Cost of a Reservation \*/

```

SELECT
IF(P1.IsStudent=1,P1.BagPrice*'+SDiscount+',P1.BagPrice)
AS StudentBagPrice FROM

(SELECT Ticket.ReservationID,
User.Username, Customer.IsStudent, Ticket.DepartureDate,
IF('+MaxBags+' < Ticket.NumBags + '+Freebags+', Ticket.Price
+ 30 * ('+MaxBags+'-(Ticket.NumBags+'+FreeBags+')),
Ticket.Price)

AS BagPrice, Ticket.Price, Ticket.NumBags FROM

Ticket INNER JOIN Reservation ON Reservation.ReservationID
= Ticket.ReservationID INNER JOIN User

ON Reservation.Username = User.Username INNER JOIN
Customer ON Customer.Username = User.Username) AS P1

WHERE P1.Username= '+Username+' AND P1.ReservationID =
'+ReservationID+';

```

**/\*Removing a Ticket from a Reservation \*/**

**DELETE FROM Ticket WHERE Ticket.TicketID='+ticketID+';**

**/\*Confirmation Screen Where Users are given Reservation ID\*/**

**INSERT INTO Ticket VALUES ('+TicketID+', '+ReservationID+',  
' +NumBags+', '+DepartureDate+',  
' +PassengerName+', '+ArrivalRouteNum+', '+DepartureRouteNum+', '+Class+', '+Price+');**

**/\* Update Reservation 1:**

**Selecting Information for Update\*/**

**SELECT R1.TrainNumber AS `Train`,  
concat(DATE\_FORMAT(R1.DepartureDate, '%b %d'), '  
' ,R2.DepartureTime, ' - ', R1.ArrivalTime, ' ',  
Hour(TIMEDIFF(R1.ArrivalTime,R2.DepartureTime)), 'hr', Minute  
(TIMEDIFF(R1.ArrivalTime,R2.DepartureTime)), 'min')  
AS Time, R2.StationName AS `Departs From`, R1.StationName AS  
`Arrives At`, R1.Class AS `Class`, R1.Price AS `Price`,  
R1.NumBags AS `#of Bags`, R1.PassengerName AS `Passenger  
Name`**

**FROM (SELECT \* FROM Ticket INNER JOIN Route ON  
Ticket.ArrivalRouteNum=Route.RouteNum NATURAL JOIN  
Takes) R1, (SELECT \* FROM Ticket INNER JOIN Route ON  
Ticket.DepartureRouteNum=Route.RouteNum) R2**

```
WHERE R1.ReservationID = '1' AND R2.ReservationID = '1' AND  
R1.TicketID = R2.TicketID;
```

```
/*Updating the Reservation to a new Date*/
```

```
UPDATE Ticket SET DepartureDate = '+NewDepartureDate+'  
WHERE TicketID='+ticketID+';
```

```
UPDATE Reservation SET NumUpdates=NumUpdates+1  
WHERE ReservationID= '1';
```

```
/*Calculating the Total Cost of an Updated Reservation*/
```

```
SELECT IF(P2.DepartureDate !='+NewDepartDate+',  
P2.StudentBagPrice + (P2.NumUpdates* 50), StudentBagPrice)  
AS `Total Cost`
```

```
FROM
```

```
(SELECT P1.ReservationID, P1.NumUpdates, P1.IsStudent,  
P1.Price,  
P1.Username,IF(P1.IsStudent=1,P1.BagPrice*'+SDiscount+',P1.  
BagPrice)
```

```
AS StudentBagPrice, P1.BagPrice, P1.DepartureDate FROM
```

```
(SELECT Ticket.ReservationID,  
User.Username, Customer.IsStudent, Ticket.DepartureDate,  
Reservation.NumUpdates,
```

```
IF(4 < Ticket.NumBags + '+Freebags+', Ticket.Price + 30 *  
('+MaxBags+'-(Ticket.NumBags+'+FreeBags+')), Ticket.Price)  
AS BagPrice, Ticket.Price, Ticket.NumBags
```

```
FROM Ticket INNER JOIN Reservation ON  
Reservation.ReservationID = Ticket.ReservationID INNER JOIN  
User ON Reservation.Username = User.Username INNER JOIN  
Customer ON Customer.Username = User.Username)
```

```
AS P1) AS P2
```

```
WHERE P2.Username= '+Username+' AND P2.ReservationID =  
'+ReservationID+';
```

```
/*Calculating the Total Cost of a Cancelled Reservation*/
```

```
SELECT P3.TotalCost AS `Total Cost`, CURDATE() AS `Cancel  
Date`,
```

```
IF(P3.IsCancelled = 0 AND DATE_ADD(CURDATE(), INTERVAL  
7 DAY) > P3.DepartureDate, P3.TotalCost*0.8,
```

```
IF(P3.IsCancelled =0 AND DATE_ADD(CURDATE(), INTERVAL  
7 DAY) < P3.DepartureDate AND P3.DepartureDate >  
DATE_ADD(CURDATE(), INTERVAL 1 DAY), P3.TotalCost*0.5,
```

```
IF(P3.IsCancelled = 0 AND P3.TotalCost<0, 0,
```

```
IF(P3.IsCancelled = 0 AND P3.TotalCost>0, P3.TotalCost - 50,
```

```
IF( P3.IsCancelled!=0 OR DATE_ADD(CURDATE(), INTERVAL 1  
DAY) <=P3.DepartureDate,P3.TotalCost,P3.TotalCost)))) AS  
`Amount Refunded` FROM
```

```
(SELECT IF(P2.DepartureDate !='DepartureDate' ,  
P2.StudentBagPrice + 50, StudentBagPrice)
```

```
AS TotalCost, P2.Username, P2.ReservationID,  
P2.DepartureDate, P2.IsCancelled FROM
```



```

(SELECT P1.ReservationID, P1.IsStudent, P1.Price,
P1.IsCancelled, P1.Username,
IF(P1.IsStudent=1,P1.BagPrice*0.8,P1.BagPrice) AS
StudentBagPrice, P1.BagPrice, P1.DepartureDate FROM
(SELECT Ticket.ReservationID,
User.Username, Customer.IsStudent,
Reservation.IsCancelled, Ticket.DepartureDate,
IF(4 < Ticket.NumBags+'FreeBags', Ticket.Price + 30*(4-
(Ticket.NumBags+'FreeBags')), Ticket.Price) AS
BagPrice, Ticket.Price, Ticket.NumBags
FROM Ticket INNER JOIN Reservation ON
Reservation.ReservationID = Ticket.ReservationID INNER JOIN
User ON Reservation.Username = User.Username INNER JOIN
Customer ON Customer.Username = User.Username)
AS P1) AS P2) AS P3
WHERE P3.Username= 'username' AND P3.ReservationID = '1';

```

/\*Cancels the Reservation\*/

```

INSERT INTO Reservation(IsCancelled) VALUES(1) WHERE
ReservationID = '+ReservationID+';

```

/\*Customers can View Reviews\*/

```

SELECT Rating, Comment FROM Review
WHERE TrainNumber = '1000 Express'
ORDER BY FIELD(Rating, 'Very Good', 'Good', 'Neutral', 'Bad' ,
'Very Bad');

```

**/\*Customers can give Reviews\*/**

**INSERT INTO Review VALUES(NULL,'RachelThorne','Good', 'My comment','1000 Express');**

**/\*Manager can View the Report of the Revenue each Train Makes in the past 3 months\*/**

**SELECT DATE\_FORMAT(P4.DepartureDate,'%M') AS  
`Month`,SUM(P4.UpdatedCost) AS `Revenue`  
FROM (SELECT P3.TotalCost AS `Total Cost`, CURDATE() AS  
`Cancel Date`, P3.DepartureDate,  
(IF(P3.IsCancelled = 0 AND DATE\_ADD(CURDATE(),  
INTERVAL 7 DAY) > P3.DepartureDate, P3.TotalCost\*0.8,  
IF(P3.IsCancelled =0 AND DATE\_ADD(CURDATE(), INTERVAL  
7 DAY) < P3.DepartureDate AND P3.DepartureDate >  
DATE\_ADD(CURDATE(), INTERVAL 1 DAY), P3.TotalCost\*0.5,  
IF(P3.IsCancelled = 0 AND P3.TotalCost<0, 0,  
IF(P3.IsCancelled = 0 AND P3.TotalCost>0, P3.TotalCost - 50,  
IF( P3.IsCancelled!=0 OR DATE\_ADD(CURDATE(), INTERVAL 1  
DAY)  
<=P3.DepartureDate,P3.TotalCost,P3.TotalCost)))))+P3.TotalC  
ost) AS UpdatedCost FROM  
(SELECT IF(P2.DepartureDate !=2016-06-02 ,  
P2.StudentBagPrice + 50, StudentBagPrice) AS TotalCost,  
P2.Username, P2.ReservationID, P2.DepartureDate,  
P2.IsCancelled FROM**

```

(SELECT P1.ReservationID, P1.IsStudent, P1.Price,
P1.IsCancelled, P1.Username,
IF(P1.IsStudent=1,P1.BagPrice*0.8,P1.BagPrice) AS
StudentBagPrice, P1.BagPrice, P1.DepartureDate FROM

(SELECT Ticket.ReservationID,
User.Username, Customer.IsStudent,
Reservation.IsCancelled, Ticket.DepartureDate,
IF(4 < Ticket.NumBags+2, Ticket.Price + 30*(4-
(Ticket.NumBags+2)), Ticket.Price) AS
BagPrice, Ticket.Price, Ticket.NumBags FROM Ticket INNER
JOIN Reservation ON Reservation.ReservationID =
Ticket.ReservationID INNER JOIN User ON
Reservation.Username = User.Username INNER JOIN Customer
ON Customer.Username = User.Username)
AS P1) AS P2) AS P3

WHERE DATE_FORMAT(P3.DepartureDate,'%M') = 'February'
OR DATE_FORMAT(P3.DepartureDate,'%M')='March' OR
DATE_FORMAT(P3.DepartureDate,'%M')= 'April') AS P4
GROUP BY DATE_FORMAT(P4.DepartureDate,'%M')

ORDER BY FIELD(DATE_FORMAT(P4.DepartureDate,'%M'),
'February','March','April'), COUNT(UpdatedCost) DESC;

/*View Popular Train Routes Reports from the past 3 months*/
Select R1.Month, Train.TrainNumber, Count(*) AS
Count FROM (SELECT Ticket.ReservationID,
Ticket.ArrivalRouteNum,

```

DATE\_FORMAT(Ticket.DepartureDate,'%M') AS Month FROM  
Ticket

WHERE DATE\_FORMAT(Ticket.DepartureDate,'%M') =  
'February' OR

DATE\_FORMAT(Ticket.DepartureDate,'%M')='March' OR  
DATE\_FORMAT(Ticket.DepartureDate,'%M')= 'April')

AS R1 INNER JOIN Route ON Route.RouteNum =  
R1.ArrivalRouteNum INNER JOIN Takes ON Takes.RouteNum =  
Route.RouteNum INNER JOIN Train ON Train.TrainNumber =  
Takes.TrainNumber GROUP BY R1.Month,Train.TrainNumber  
ORDER BY FIELD(Month, 'February','March','April'), COUNT(\*)  
DESC;

/\*Gets the Arrival Route Number and Depature Route Number  
for Tickets\*/

Select Depart.RouteNum, Arrive.RouteNum FROM

(SELECT Train.TrainNumber, Route.RouteNum,  
Route.StationName FROM Train INNER JOIN Takes ON  
Takes.TrainNumber = Train.TrainNumber INNER JOIN Route  
ON Route.RouteNum = Takes.RouteNum AND  
Route.StationName = 'Atlanta(ATL)' AND Train.TrainNumber =  
'1000 Regional' ) AS Depart,

(SELECT Train.TrainNumber, Route.RouteNum,  
Route.StationName FROM Train INNER JOIN Takes ON  
Takes.TrainNumber = Train.TrainNumber INNER JOIN Route  
ON Route.RouteNum = Takes.RouteNum AND  
Route.StationName = 'Chicago(CHI)' AND Train.TrainNumber =  
'1000 Regional') AS Arrive

```
WHERE Arrive.TrainNumber = Depart.TrainNumber;
```