**Phase 3 – Group 17**

**Team Members:**

Jatin Nanda; jnanda3@gatech.edu; jnanda3

Christine Shih; cshih37@gatech.edu; cshih37

Esha Singh; esingh7@gatech.edu; esingh7

Raymond Wong; rwong1103@gatech.edu; rwong32

SQL STATEMENTS – PHASE 3

**Login attempt:**

SELECT \*

FROM Manager

WHERE Username = *username*

AND Password = *password*;

**Manager Login Attempt**

SELECT \*

FROM Manager

WHERE Username = *username*

AND Password = *password;*

**Get Manager Username List**

SELECT Username FROM Manager;

**New User Registration**

INSERT INTO Customer (Username, Password, Email) VALUES (*username, password, email*);

**Add Review for Train X**

INSERT INTO Review (Comment, Rating, Username, TrainNumber)

VALUES (*comment, rating, username, trainx*);

**Set As Student**

UPDATE Customer SET isStudent = 1

WHERE Username = *customer*;

**Get Schedule for Train**

SELECT \* FROM Stop

WHERE TrainNumber = *trainId;*

**Get Good Reviews**

SELECT COMMENT FROM Review

WHERE TrainNumber = ‘*trainId’*

AND Rating > 5

AND Comment IS NOT NULL;

**Get Neutral Reviews**

SELECT COMMENT FROM Review

WHERE TrainNumber = ‘*trainId’*

AND Rating = 5

AND Comment IS NOT NULL;

**Get Bad Reviews**

SELECT COMMENT FROM Review

WHERE TrainNumber = ‘*trainId’*

AND Rating < 5

AND Comment IS NOT NULL;

**Get Stations**

SELECT Name, Location FROM Station;

**Find Requested Trains**

SELECT Stop.TrainNumber, StationName, ArrivalTime, DepartureTime, 1stClassPrice, 2ndClassPrice

FROM Stop JOIN TrainRoute on Stop.TrainNumber = TrainRoute.TrainNumber

WHERE (stationName = "Mordor" OR StationName = "Winterfell")

AND (Stop.TrainNumber

IN (

SELECT TrainNumber FROM (

SELECT Stop.TrainNumber, StationName FROM Stop JOIN TrainRoute ON Stop.TrainNumber = TrainRoute.TrainNumber WHERE (

stationName = "Mordor" OR StationName = "Winterfell")

) a Group by TrainNumber having count(\*) = 2 )

)

**Check if Train Exists**

SELECT \* FROM TrainRoute

WHERE TrainNumber = *trainNum;*

**Add Card**

INSERT into PaymentInfo (CardNumber, CVV, ExpDate, NameOnCard, Username)

VALUES (*cardNum, cardCVV, expDate, name, username*);

**Remove Card**

DELETE FROM PaymentInfo

WHERE CardNumber LIKE ‘%*LAST 4 DIGITS*’

AND Username LIKE ‘ *username*’;

**Verify Card Not In Use**

SELECT Reserves.DepartureDate FROM (PaymentInfo INNER JOIN Reservation ON

PaymentInfo.CardNumber = Reservation.CardNumber) INNER JOIN Reserves ON

Reservation.ReservationId = Reserves.ReservationId WHERE PaymentInfo.CardNumber like

'%*last4Digits*'

**Get Cards that belong to user**

SELECT CardNumber FROM PaymentInfo

WHERE Username LIKE *username*;

**Get Card Number (using last 4 digits)**

Select CardNumber FROM PaymentInfo

WHERE CardNumber LIKE ‘%*LAST 4 DIGITS*’

AND Username LIKE ‘username’;

**Get isStudent for user**

SELECT isStudent FROM Customer

WHERE Username = *username*;

**Add Reservation (1 into Reservation, 1 into Reserves)**

INSERT into Reservation (ReservationId, isCancelled, Username, CardNumber, Price)

VALUES ( *reservationId, 0, username, cardNumber, price*);

INSERT into Reserves (ReservationId, TrainNumber, Class, DepartureDate, PassengerName, NumberOfBags, DepartsFrom, ArrivesAt)

VALUES ( *reservationId, trainNumber, class, departureDate, passengerName, numBags, departsFrom, arrivesAt);*

**Get Selected Reservation**

SELECT ReservationId FROM Reservation

WHERE ReservationId = *id* AND Username = *username*

AND isCancelled = 0;

**Get Class Prices**

SELECT *classVal* FROM TrainRoute WHERE TrainNumber = *trainNum;*

**Get Reservation Price**

SELECT Price FROM Reservation WHERE ReservationId = *resId*;

**Update Reservation Price and Time**

UPDATE Reservation SET Price = *price* WHERE ReservationId = *resId*;

UPDATE Reserves SET DepartureDate = *time* WHERE ReservationId = *resId;*

**Get Revenue for Past 3 months**

SELECT SUM(Reservation.Price)

FROM Reservation

INNER JOIN Reserves

ON Reservation.ReservationId = Reserves.ReservationID

WHERE Reserves.DepartureDate like '%-03-%'

**Get 3 Most Popular Routes for a Month**

SELECT Reserves.TrainNumber, COUNT(Reserves.TrainNumber)

AS 'occurance'

FROM Reservation

INNER JOIN Reserves ON Reservation.ReservationID = Reserves.ReservationID

WHERE Reservation.isCancelled = 0 AND Reserves.DepartureDate LIKE '%-01-%'

GROUP BY TrainNumber

ORDER BY occurance DESC

LIMIT 3

**Cancel Reservation**

UPDATE Reservation SET Price = '*price*', isCancelled = 1 WHERE ReservationId = '*reservationId*';