Login:
SELECT Password FROM User INNER JOIN Customer ON
User.Username=Customer.Username WHERE User.Username = '$username';

SELECT Password FROM User INNER JOIN Manager ON
User.Username=Manager.Username WHERE User.Username = '$username';

Registration:

INSERT INTO User(Username, Password) VALUES('$username', '$hashedPassword');

INSERT INTO Customer(Username, Email, IsStudent)
VALUES('$username','$emailAddress','0');

Student Discount:

UPDATE Customer SET IsStudent=1 WHERE Username='$username';

Train Schedule:

SELECT TrainNumber, ArrivalTime, DepartureTime, Stop.Name, Location FROM Stop
INNER JOIN Station ON Stop.Name = Station.Name WHERE
TrainNumber='$trainNumber' ORDER BY ArrivalTime;

Search Train:
SELECT Name, Location FROM Station;

Select Departure
SELECT Stop.TrainNumber, Stop.DepartureTime, S.ArrivalTime,
TIMEDIFF(S.ArrivalTime, Stop.DepartureTime) AS TimeDifference, FirstClassPrice,
SecondClassPrice FROM Stop CROSS JOIN Stop AS S INNER JOIN TrainRoute ON
Stop.TrainNumber=TrainRoute.TrainNumber WHERE
S.TrainNumber=Stop.TrainNumber AND S.ArrivalTime>Stop.DepartureTime AND
Stop.Name = '$departsFrom' AND S.Name='$arrivesAt';

Travel Extras & Passenger Info:
SELECT MaxNumOfBaggage, NumOfFreeBaggage FROM SystemInfo;

Make Reservation:

SELECT Stop.DepartureTime, S.ArrivalTime, TIMEDIFF(S.ArrivalTime, Stop.DepartureTime) AS TimeDifference, (SELECT Location FROM Station WHERE Name=DepartsFrom) AS DepartureLocation, (SELECT Location FROM Station WHERE Name=ArrivesAt) AS ArrivalLocation, FirstClassPrice, SecondClassPrice FROM Stop CROSS JOIN Stop AS S INNER JOIN TrainRoute ON Stop.TrainNumber=TrainRoute.TrainNumber WHERE S.TrainNumber=Stop.TrainNumber AND S.ArrivalTime>Stop.DepartureTime AND Stop.Name = '$departsFrom' AND S.Name='$arrivesAt' AND TrainRoute.TrainNumber='$trainNumber';

SELECT CardNumber FROM Customer JOIN PaymentInfo ON Customer.Username=PaymentInfo.Username WHERE Customer.Username='$username';

SELECT IsStudent FROM Customer WHERE Username='$username';

SELECT NumOfFreeBaggage, StudentDiscount FROM SystemInfo;

INSERT INTO Reservation (TotalCost, IsCancelled, Username, CardNumber) VALUES('$totalCost', '0', '$username', '$cardNumber');

INSERT INTO Reserves
        VALUES((SELECT MAX(ReservationID) FROM Reservation WHERE Username='$username'), '$trainNumber', '$class', '$departureDate',
        '$passengerName', '$numOfBaggages', '$departsFrom', '$arrivesAt');

Get Cards:
SELECT CardNumber FROM PaymentInfo WHERE Username='$username';

Add Card:
INSERT INTO PaymentInfo
VALUES('$CardNumber', '$CVV', '$expDate', '$nameOnCard', '$username');

Delete Card:

UPDATE Reservation SET CardNumber=NULL WHERE CardNumber='$cardNumber' AND CardNumber NOT IN (SELECT CardNumber FROM (SELECT CardNumber FROM Reservation JOIN Reserves ON Reservation.ReservationID=Reserves.ReservationID WHERE CURDATE()<=DepartureDate AND IsCancelled='0') AS Cn);

DELETE FROM PaymentInfo WHERE CardNumber NOT IN (SELECT CardNumber FROM (SELECT CardNumber FROM Reservation JOIN Reserves ON Reservation.ReservationID=Reserves.ReservationID WHERE CURDATE()<=DepartureDate AND IsCancelled='0') AS Cn) AND CardNumber='$cardNumber';

Confirmation:
SELECT MAX(ReservationID) AS Rid FROM Reservation WHERE Username='$username';

Update Reservation:
SELECT ChangeFee FROM SystemInfo;

SELECT TotalCost FROM Reservation WHERE ReservationID='$reservationID';

UPDATE Reserves SET DepartureDate='$departureDate' WHERE ReservationID='$reservationID' AND TrainNumber='$trainNumber';

UPDATE Reservation SET TotalCost='$newCost' WHERE ReservationID='$reservationID';

Cancel Reservation:
UPDATE Reservation SET TotalCost='$newCost', IsCancelled='1' WHERE ReservationID='$reservationID';

Search Reservation (Used in Update Reservation and Cancel Reservation):
SELECT Stop.TrainNumber, DepartureDate, Stop.DepartureTime, S.ArrivalTime, TIMEDIFF(S.ArrivalTime, Stop.DepartureTime) AS TimeDifference, (SELECT Location FROM Station WHERE Name=DepartsFrom) AS DepartureLocation, DepartsFrom, (SELECT Location FROM Station WHERE Name=ArrivesAt) AS ArrivalLocation, ArrivesAt, Class, FirstClassPrice, SecondClassPrice, NumberOfBaggages, PassengerName FROM Stop CROSS JOIN Stop AS S INNER JOIN TrainRoute ON Stop.TrainNumber=TrainRoute.TrainNumber JOIN Reserves ON Reserves.TrainNumber=TrainRoute.TrainNumber JOIN Reservation ON Reserves.ReservationID=Reservation.ReservationID WHERE S.TrainNumber=Stop.TrainNumber AND S.ArrivalTime>Stop.DepartureTime AND Stop.Name=Reserves.DepartsFrom AND S.Name=Reserves.ArrivesAt AND Reserves.ReservationID='$reservationID' AND Username='$username' AND IsCancelled='0' AND CURDATE()<DepartureDate;

SELECT TotalCost FROM Reservation WHERE Reservation ID = '$reservationID' AND Username = '$username';

View Review:
SELECT Comment, Rating FROM Review WHERE TrainNumber='$trainNumber' ORDER BY ReviewNumber DESC;

Give Review:
 SELECT TrainNumber FROM TrainRoute WHERE TrainNumber='$trainNumber';

//For review without comment
INSERT INTO Review (Rating, Username, TrainNumber) VALUES('$rating', '$username', '$trainNumber');

//For review with comment
INSERT INTO Review (Comment, Rating, Username, TrainNumber) VALUES('$comment', '$rating', '$username', '$trainNumber');

Revenue Report:

SELECT SUM(TotalCost) AS Revenue FROM Reservation WHERE ReservationID IN (SELECT Reserves.ReservationID FROM Reservation JOIN Reserves ON Reservation.ReservationID=Reserves.ReservationID WHERE EXTRACT(MONTH FROM DepartureDate)='$monthNumber' AND EXTRACT(YEAR FROM DepartureDate)='$year' AND Reserves.ReservationID NOT IN (SELECT ReservationID FROM Reserves WHERE DepartureDate<'$year-$monthNumber-01') GROUP BY Reserves.ReservationID);

Popular Train Report:

SELECT TrainNumber, COUNT(Reserves.ReservationID) AS Total FROM Reserves JOIN Reservation ON Reserves.ReservationID=Reservation.ReservationID WHERE EXTRACT(MONTH FROM DepartureDate)='$monthNumber' AND EXTRACT(YEAR FROM DepartureDate)='$year' AND IsCancelled='0' GROUP BY TrainNumber ORDER BY COUNT(Reserves.ReservationID) DESC LIMIT 3;