PART-A

Write Queries to.

Use SET Operators

1. Find the train numbers for which reservation have not yet been made.
2. Find the train names that donot have a first AC class coach.
3. Print all the PNR nos available in the database.
4. Find passenger names who have booked to 'Pune'.

Use Nested Query(in Operators)

1. Find the train names that stop in 'Katpadi'.
2. Find the train names that are superfast and the service tax is zero.
3. Find the Passenger name who have booked for the train that starts from 'chennai'.
4. Find the trains names that have all the AC coaches and the base fare is less than 3000 for each case.

Use Join Query

1. Find the train names that stop in 'Katpadi'.
2. Find the train names that are superfast and the service tax is zero.
3. Find the Passenger name (and train name) who have booked for the train that starts from 'chennai'.
4. Display the trains names , each type of class and the total fare for each type of class.
5. Display all the train details and the ticket details(if booked any).
6. Create a sequence to provide values for the PNR no.
7. Write a query for full outer join using any of the tables above.

PART-B

Write Queries to.

Use Coorelated (and nested) Query

1. Find the train names for which ten tickets have been reserved.
2. Find the trains that have more than ten substations.
3. Find the passengers who do not pass through 'Mettupalam'.
4. Find passengers who have booked for super fast trains.

Complex queries(use groupby/groupby having/join/nested)

1. Take the start station code and end station code and display the train details.
2. List the train names and the number of sub stations it has.
3. List the stations where all types of trains stop.
4. List the trains names that has atleast four bookings.
5. Create a table cancellation history(Insert values from ticket and passenger table).
6. Create a table for all the train numbers and class available in train_ticket_fare with total seats.
7. Find the station name that has highest number of trains stopping at.