

1. Create a folder on the desktop of your computer and name it by your ID. Save your all works in that folder. 1.Create a 3NF database which will have following Data with a Data File and a Log File by writing SQL script as follows: Database name: RailwayDB, Datafile name : RailwayDB_Data_1 , LogFile Name: RailwayDB_Log_1, Location: Default Database file location, Size(Datafile : 25 MB , LogFile: 2 MB), Maximum Size(Datafile : 100 MB , LogFile: 25 MB), File Growth (Datafile : 5% , LogFile: 1%).

Passenger Name	Phone no	Identification type	Identification number	Train Name	Journey name	Classes	Booking date	Journey date	PNR	no of tkt	Tkt Price	VAT	Service charge	Amount total
Shefine Ahmed	01712345678	NID	11223344	Chilahati Express (805)	9:30 am Dhaka to Nillphamari	AC_S	3/1/2024	4/1/2024	12345AD1BD7	1	1076			
Harry Potter	01712345679	NID	11223355	Nillsagor Express (765)	11:00 AM Dhaka to Natore	Snigdha	4/1/2024	5/1/2024	12345AD2BD1	1	610			
Rayan Gosling	01712345671	NID	11223366	Ekota Express (705)	12:00 AM Dhaka to Natore	C_Chair	4/1/2024	6/1/2024	12345AD3BD2	2	320			
Joye Tribbiany	01712345672	NID	11223377	Simanto Express (748)	3:00 PM Dhaka to Khulna	S_Chair	5/1/2024	6/1/2024	12345AD4BD3	1	290			
Ross Gallar	01712345673	NID	11223388	Nillsagor Express (766)	10:00 AM Dhaka to Natore	Shovon	5/1/2024	7/1/2024	12345AD5BD4	1	300			

2. Insert all the records using script .
3. Create a join query for Those passengers who have booked more than one seat and on a particular date.
4. Create a join query for those Passengers whose class are all except AC_S and Snigdha and journey date are later than 2024-05-01.

5. Create a join query for those Passengers who travel from 2024-04-01 to 2024-06-01.
6. Create a query to Stations that starts with Nil.
7. Create a query to find out Passengers whose name has one of the following characters:a,e,i,t
8. Create a query to find out Passengers whose name starts with R and next letter is one of the A through J.
9. Create a query to find out Passengers whose name starts with R and next letter is not one of the A through J.
10. Give an example of top clause
11. Give an example of OFFSET and FETCH Clause.
12. Give an example of Group Query.
13. Give an example of a ROLLUP operator (A summary query that includes a summary row for each GROUPING LEVEL).
14. Give an example of a CUBE operator (A summary query that includes a summary row for each set of groups).
15. Give an example of GROUPING SETS operator (A summary query with a composite grouping).
16. Give an example of OVER Clause (A query that calculates a cumulative total and moving average).
17. Give an example of SUBQUERY (Select statement that uses a subquery in where clause).
18. INSERT Record into table using script.
19. Write A query that returns passengers who have paid a larger amount than the largest amount paid by passenger 702 by using all keyword
20. Write A query that returns passengers who have paid a smaller amount than the largest amount paid by passenger 702 by using any keyword
21. Write A query that returns passengers who have paid a smaller amount than the largest amount paid by passenger 702 by using SOME keyword
22. Write A query that uses a correlated subquery to return passengers whose amount paid is higher than the passengers average amount paid
23. INSERT A PASSENGER RECORD.
24. DELETE A PASSENGER RECORD.
25. Give an Example of EXISTS Operator
26. Give an Example of CAST function.
27. Give an Example of CONVERT function.
28. Give an Example of CTE (Common Table Expression)
29. Give an Example of UPDATE Statement.
30. Give an Example of DELETE Row.
31. Give an Example of ERROR HANDLING with TRY....CATCH statement.
32. Give an Example of MERGE.
33. Give an Example of CURSOR
34. Create a script that calls the stored procedure.
35. Create a script that calls the stored procedure.
36. Create a script that invokes the function.
37. Create a SELECT statement that invokes the function
38. Create a script that calls the procedure.

39. Create a TRANSACTION with two save points.
40. Add a column
41. Drop a column
42. Drop table
43. Create a cluster index
44. Create an UPDATEABLE VIEW.
45. A create VIEW Statement that creates a READ-ONLY VIEW (Non updateable).
46. Create an ENCRYPTED VIEW.
47. A Create PROCEDURE statement that includes output and optional parameter
48. Create a RETURN statement for a stored procedure/A STORED PROCEDURE that returns a value.
49. Create A statement that creates a SCALAR-VALUED FUNCTION.
50. Create A statement that creates a SIMPLE TABLE-VALUED function.
51. Create A statement that creates a multi valued table function.
52. Create An AFTER TRIGGER that archives deleted data.
53. Give an Example of Validate data and raise error using THROW statement.
54. Give an EXAMPLE OF MERGE.