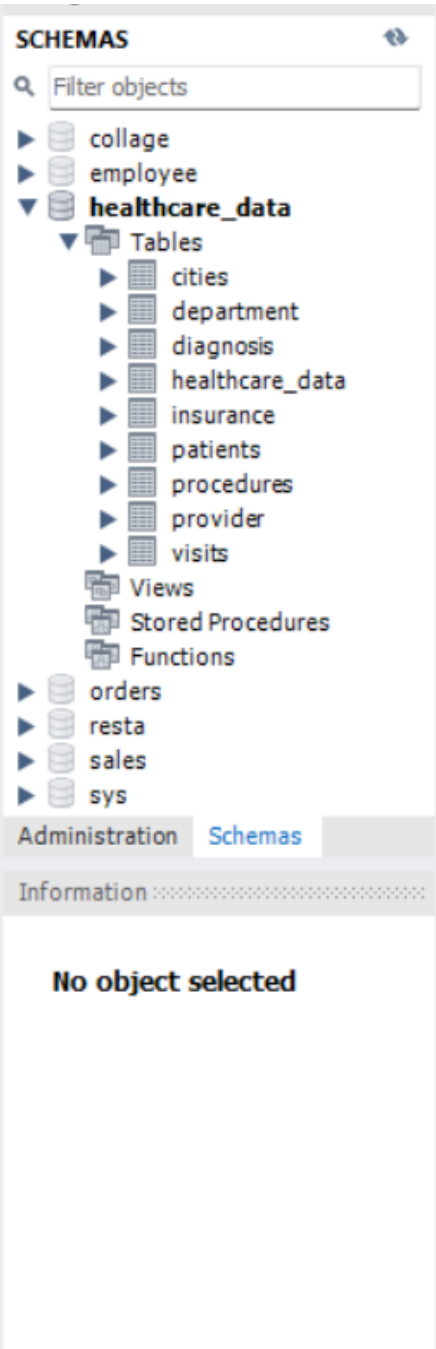
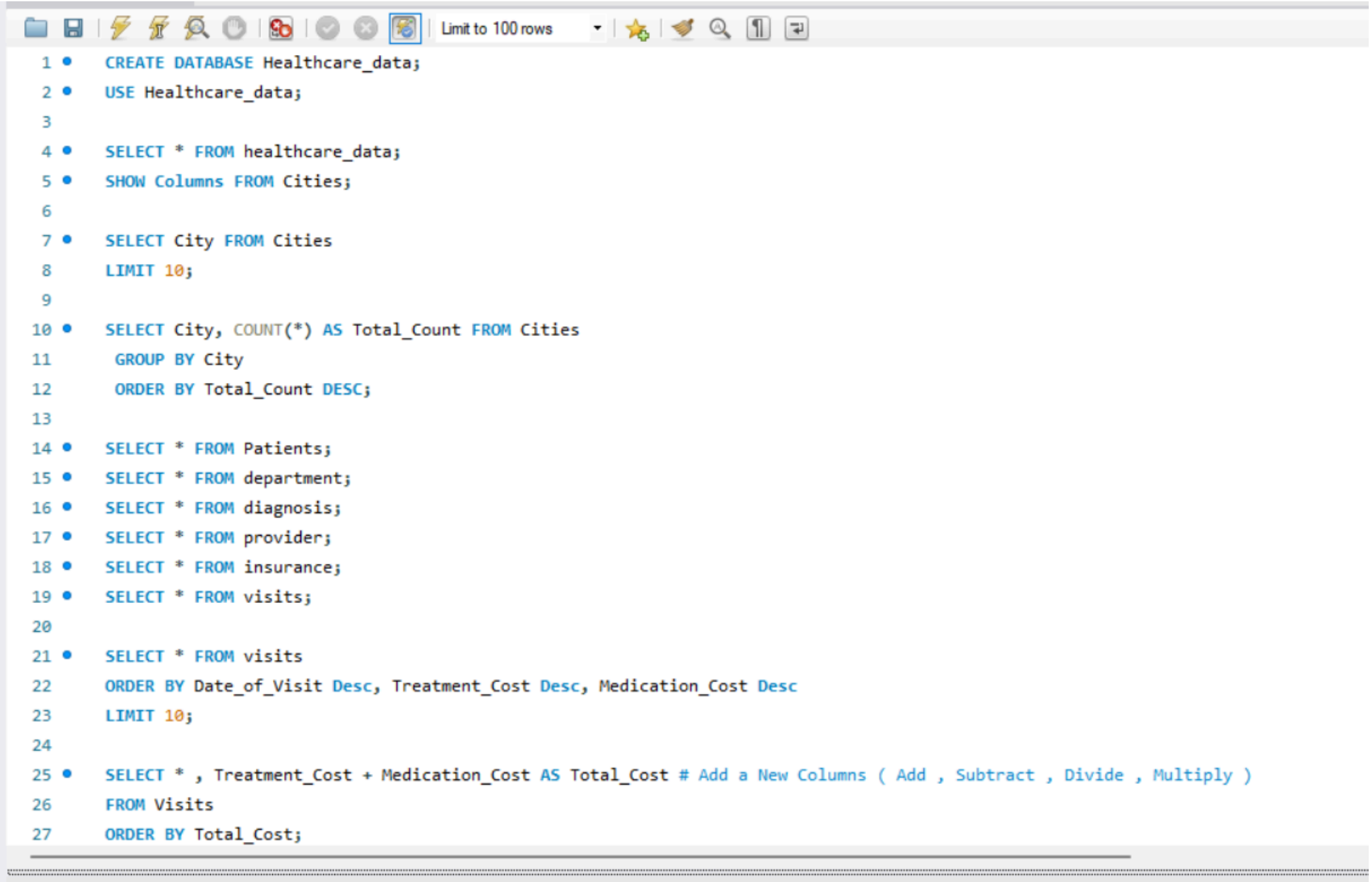
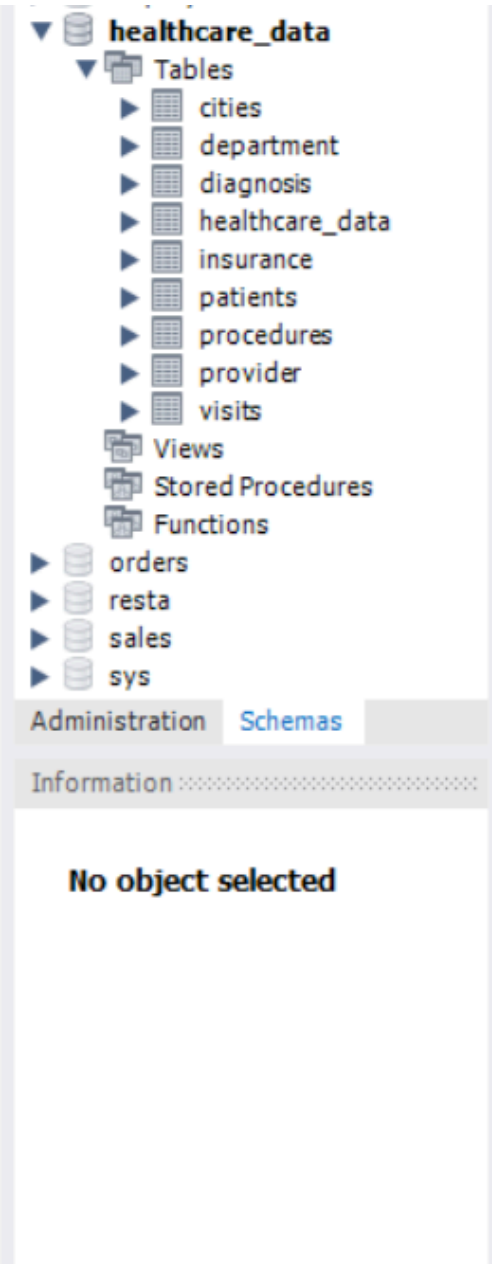
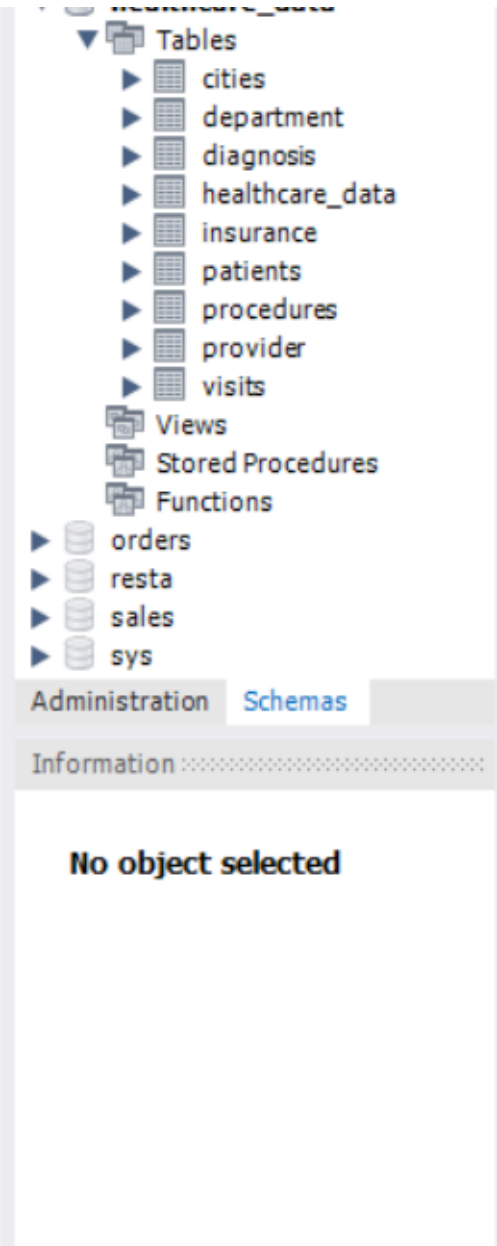

SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected

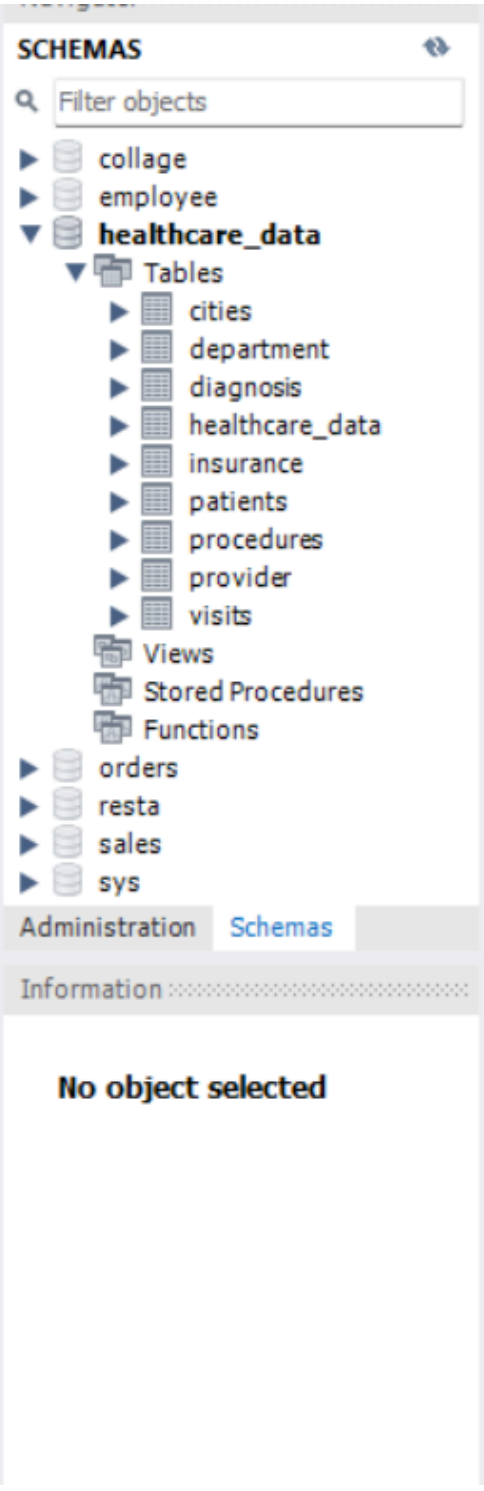

Limit to 100 rows
1 CREATE DATABASE Healthcare_data;
2 USE Healthcare_data;
3
4 SELECT * FROM healthcare_data;
5 SHOW Columns FROM Cities;
6
7 SELECT City FROM Cities
8 LIMIT 10;
9
10 SELECT City, COUNT(*) AS Total_Count FROM Cities
11 GROUP BY City
12 ORDER BY Total_Count DESC;
13
14 SELECT * FROM Patients;
15 SELECT * FROM department;
16 SELECT * FROM diagnosis;
17 SELECT * FROM provider;
18 SELECT * FROM insurance;
19 SELECT * FROM visits;
20
21 SELECT * FROM visits
22 ORDER BY Date_of_Visit Desc, Treatment_Cost Desc, Medication_Cost Desc
23 LIMIT 10;
24
25 SELECT * , Treatment_Cost + Medication_Cost AS Total_Cost # Add a New Columns (Add , Subtract , Divide , Multiply)
26 FROM Visits
27 ORDER BY Total_Cost;

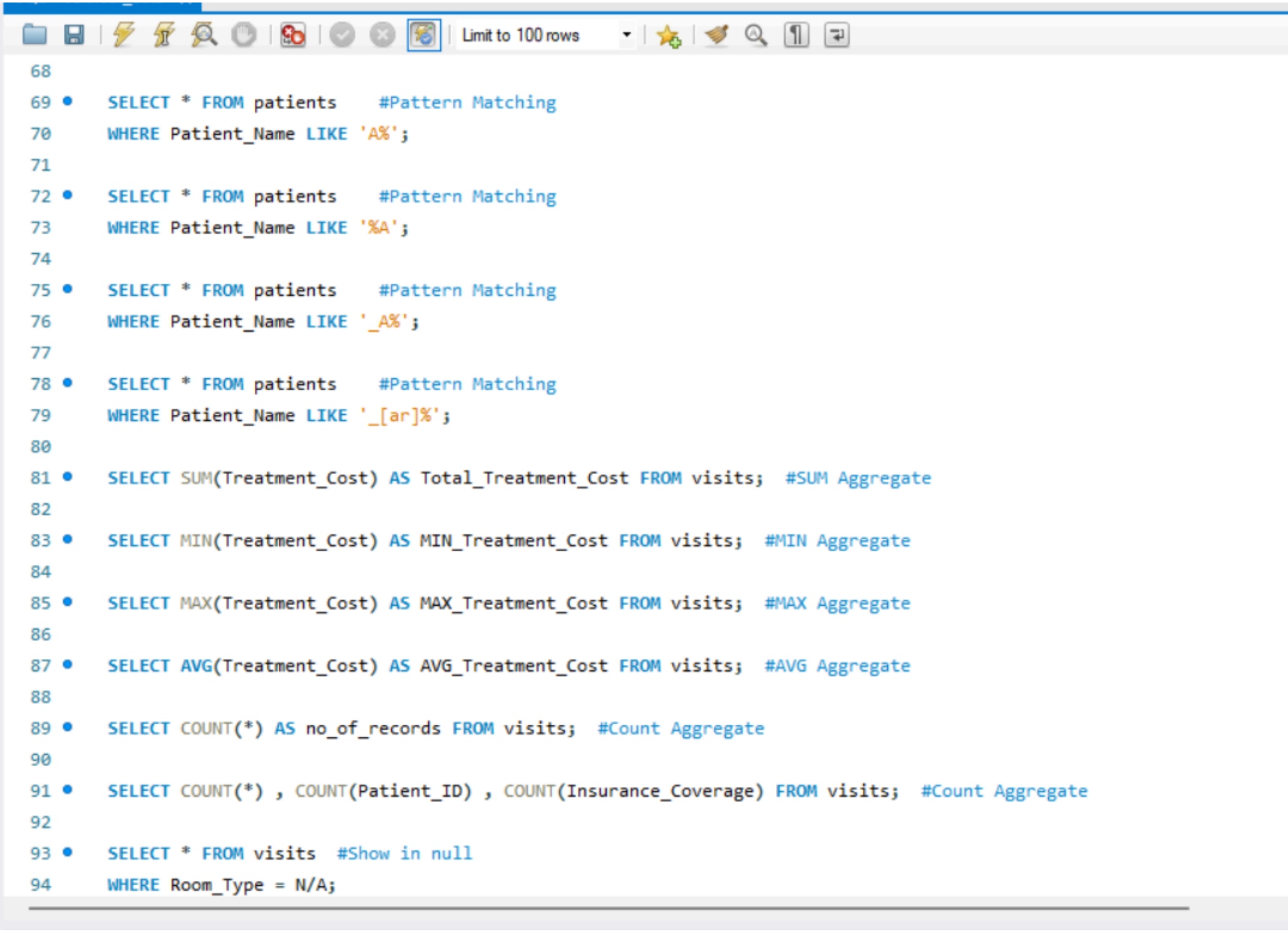


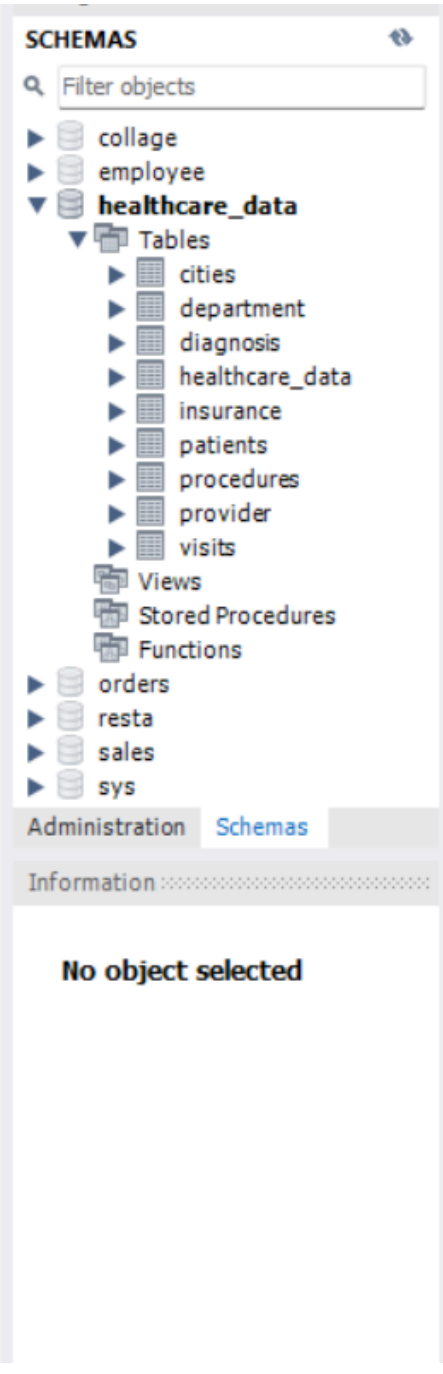

```sql
WHERE State = 'England';

SELECT * FROM visits
WHERE Patient_Satisfaction_Score>5
ORDER BY Patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Date_of_Visit = '01-01-2024';

SELECT * FROM visits
WHERE Payment_Status = 'pending' and  Room_Type = 'Private Room'; #all the condition has to satisfy

SELECT * FROM visits
WHERE Payment_Status = 'pending' or  Room_Type = 'Private Room';  #anyone of the condition has satisfy

SELECT * FROM visits
WHERE (Payment_Status = 'pending' or  Room_Type = 'Private Room') and patient_Satisfaction_Score>5
ORDER BY Patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score>=4 and patient_satisfaction_Score<=8  #Between Condition
ORDER BY patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score BETWEEN 3 and 5   #Between Condition 3,4,5
```

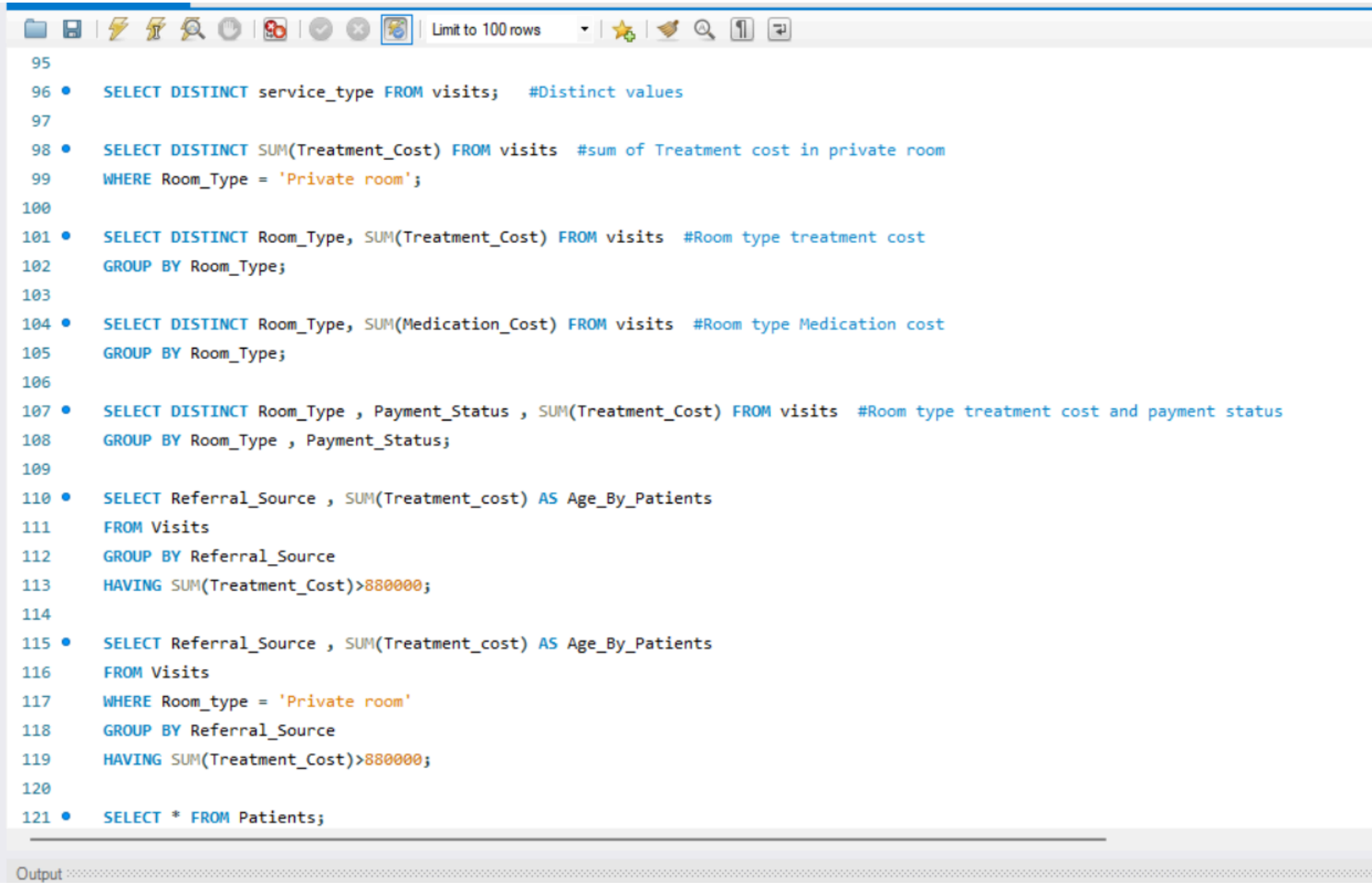


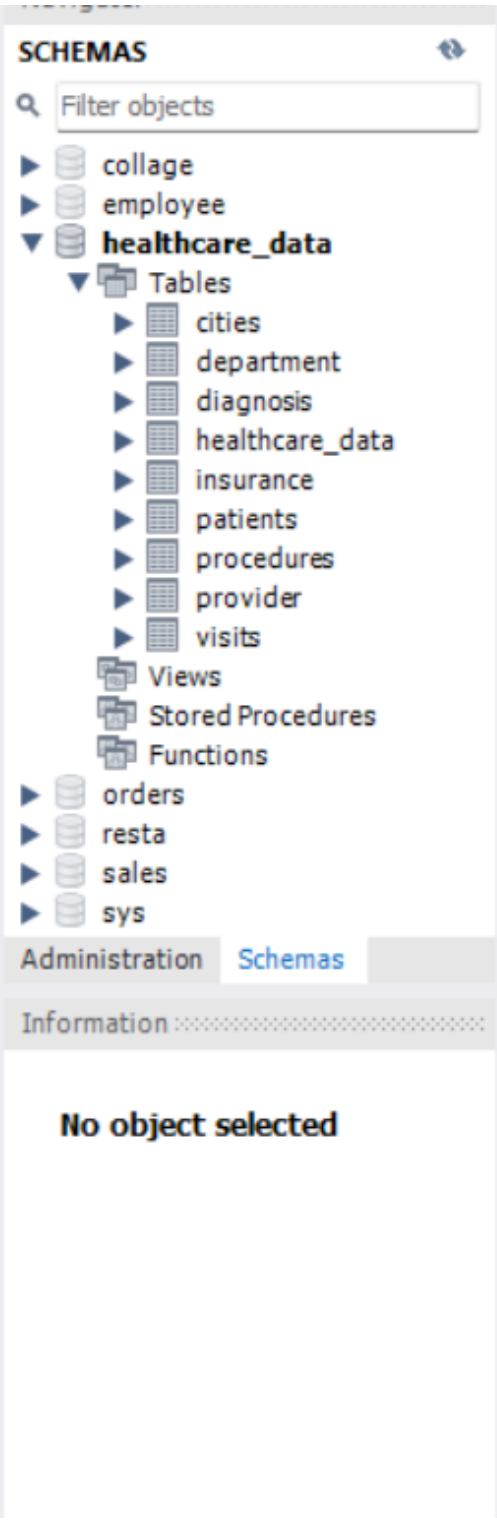

```sql
ORDER BY Patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score>=4 and patient_satisfaction_Score<=8  #Between Condition
ORDER BY patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score BETWEEN 3 and 5   #Between Condition 3,4,5
ORDER BY patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Discharge_Date BETWEEN '09-11-2024' and '01-01-2025'
ORDER BY Discharge_Date;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score IN (3,5)  #IN Condintion (3,5)
ORDER BY patient_Satisfaction_Score;

SELECT * FROM visits
WHERE Patient_Satisfaction_Score NOT IN (3,5)  #NOT IN Condintion (1,2,4,6,7,8,9)
ORDER BY patient_Satisfaction_Score;

SELECT * FROM patients    #Pattern Matching
WHERE Patient_Name LIKE 'A%';
```

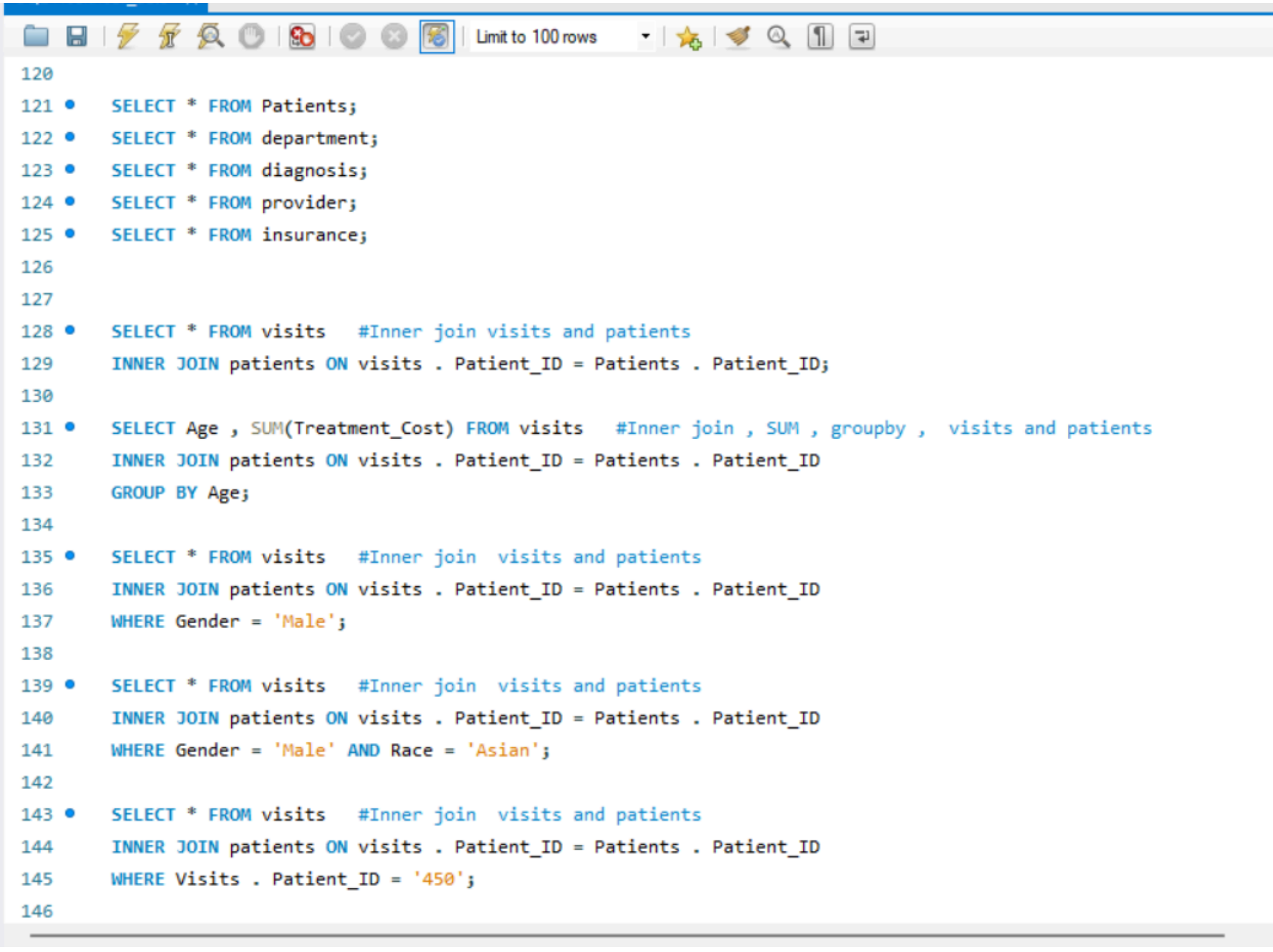

SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected


Limit to 100 rows
68
69 SELECT * FROM patients #Pattern Matching
70 WHERE Patient_Name LIKE 'A%';
71
72 SELECT * FROM patients #Pattern Matching
73 WHERE Patient_Name LIKE '%A';
74
75 SELECT * FROM patients #Pattern Matching
76 WHERE Patient_Name LIKE '_A%';
77
78 SELECT * FROM patients #Pattern Matching
79 WHERE Patient_Name LIKE '_[ar]%';
80
81 SELECT SUM(Treatment_Cost) AS Total_Treatment_Cost FROM visits; #SUM Aggregate
82
83 SELECT MIN(Treatment_Cost) AS MIN_Treatment_Cost FROM visits; #MIN Aggregate
84
85 SELECT MAX(Treatment_Cost) AS MAX_Treatment_Cost FROM visits; #MAX Aggregate
86
87 SELECT AVG(Treatment_Cost) AS AVG_Treatment_Cost FROM visits; #AVG Aggregate
88
89 SELECT COUNT(*) AS no_of_records FROM visits; #Count Aggregate
90
91 SELECT COUNT(*) , COUNT(Patient_ID) , COUNT(Insurance_Coverage) FROM visits; #Count Aggregate
92
93 SELECT * FROM visits #Show in null
94 WHERE Room_Type = N/A;


SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected


Limit to 100 rows
95
96 SELECT DISTINCT service_type FROM visits; #Distinct values
97
98 SELECT DISTINCT SUM(Treatment_Cost) FROM visits #sum of Treatment cost in private room
99 WHERE Room_Type = 'Private room';
100
101 SELECT DISTINCT Room_Type, SUM(Treatment_Cost) FROM visits #Room type treatment cost
102 GROUP BY Room_Type;
103
104 SELECT DISTINCT Room_Type, SUM(Medication_Cost) FROM visits #Room type Medication cost
105 GROUP BY Room_Type;
106
107 SELECT DISTINCT Room_Type , Payment_Status , SUM(Treatment_Cost) FROM visits #Room type treatment cost and payment status
108 GROUP BY Room_Type , Payment_Status;
109
110 SELECT Referral_Source , SUM(Treatment_cost) AS Age_By_Patients
111 FROM Visits
112 GROUP BY Referral_Source
113 HAVING SUM(Treatment_Cost)>880000;
114
115 SELECT Referral_Source , SUM(Treatment_cost) AS Age_By_Patients
116 FROM Visits
117 WHERE Room_type = 'Private room'
118 GROUP BY Referral_Source
119 HAVING SUM(Treatment_Cost)>880000;
120
121 SELECT * FROM Patients;
Output


SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected


Limit to 100 rows
120
121 SELECT * FROM Patients;
122 SELECT * FROM department;
123 SELECT * FROM diagnosis;
124 SELECT * FROM provider;
125 SELECT * FROM insurance;
126
127
128 SELECT * FROM visits #Inner join visits and patients
129 INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID;
130
131 SELECT Age , SUM(Treatment_Cost) FROM visits #Inner join , SUM , groupby , visits and patients
132 INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID
133 GROUP BY Age;
134
135 SELECT * FROM visits #Inner join visits and patients
136 INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID
137 WHERE Gender = 'Male';
138
139 SELECT * FROM visits #Inner join visits and patients
140 INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID
141 WHERE Gender = 'Male' AND Race = 'Asian';
142
143 SELECT * FROM visits #Inner join visits and patients
144 INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID
145 WHERE Visits . Patient_ID = '450';
146

Navigator

SCHEMAS

Filter objects

- collage
- employee
- **healthcare_data**
  - Tables
    - cities
    - department
    - diagnosis
    - healthcare_data
    - insurance
    - patients
    - procedures
    - provider
    - visits
  - Views
  - Stored Procedures
  - Functions
- orders
- resta
- sales
- sys

Administration | Schemas

Information

No object selected

SQL Healthcar_data

Limit to 100 rows

```
146
147 •  SELECT Date_of_Visit , Age FROM visits   #Inner join  visits and patients
148    INNER JOIN patients ON visits . Patient_ID = Patients . Patient_ID
149    WHERE Visits . Patient_ID = '450';
150
151 •  SELECT visits . * , Patients . *  FROM  visits  #INNER join  visits and patients
152    INNER JOIN  patients ON visits . Patient_ID = Patients . Patient_ID
153    WHERE visits . Room_Type IS NOT NULL;
154
155 •  SELECT patients.Age , Visits.Payment_Status  FROM  visits  #INNER join  visits and patients
156    INNER JOIN  patients ON visits . Patient_ID = Patients . Patient_ID;
157
158
159 •  SELECT * FROM  visits  #LIFT join  visits and patients
160    LEFT JOIN  patients ON visits . Patient_ID = Patients . Patient_ID
161    WHERE Patients . Gender = 'male';
162
163 •  SELECT patients. * , Visits.Payment_Status  FROM  visits  #LEFT join  visits and patients
164    LEFT JOIN  patients ON visits . Patient_ID = Patients . Patient_ID;
165
166 •  SELECT patients . Patient_Name , SUM(visits.Treatment_Cost) AS NAME_By_Treartment_Cost FROM  visits  #LEFT join  visits and patients
167    LEFT JOIN  patients ON visits . Patient_ID = Patients . Patient_ID
168    GROUP BY patients . Patient_Name;
169
170 •  SELECT *                                    # add a new Column Profit Bucket
171    ,Case WHEN Treatment_Cost < 50 THEN 'Loss'
172    WHEN Treatment_Cost < 100 THEN 'Low Profit'
```

Navigator

SCHEMAS

Filter objects

- collage
- employee
- **healthcare_data**
  - Tables
    - cities
    - department
    - diagnosis
    - healthcare_data
    - insurance
    - patients
    - procedures
    - provider
    - visits
  - Views
  - Stored Procedures
  - Functions
- orders
- resta
- sales
- sys

Administration | Schemas

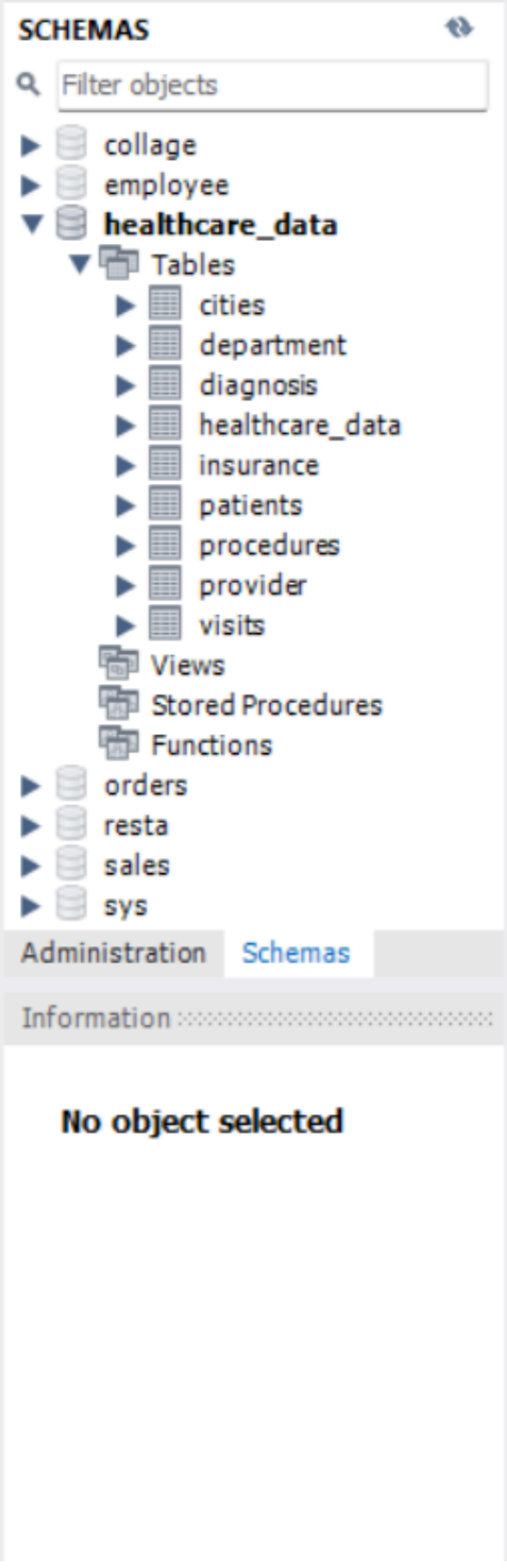
Information

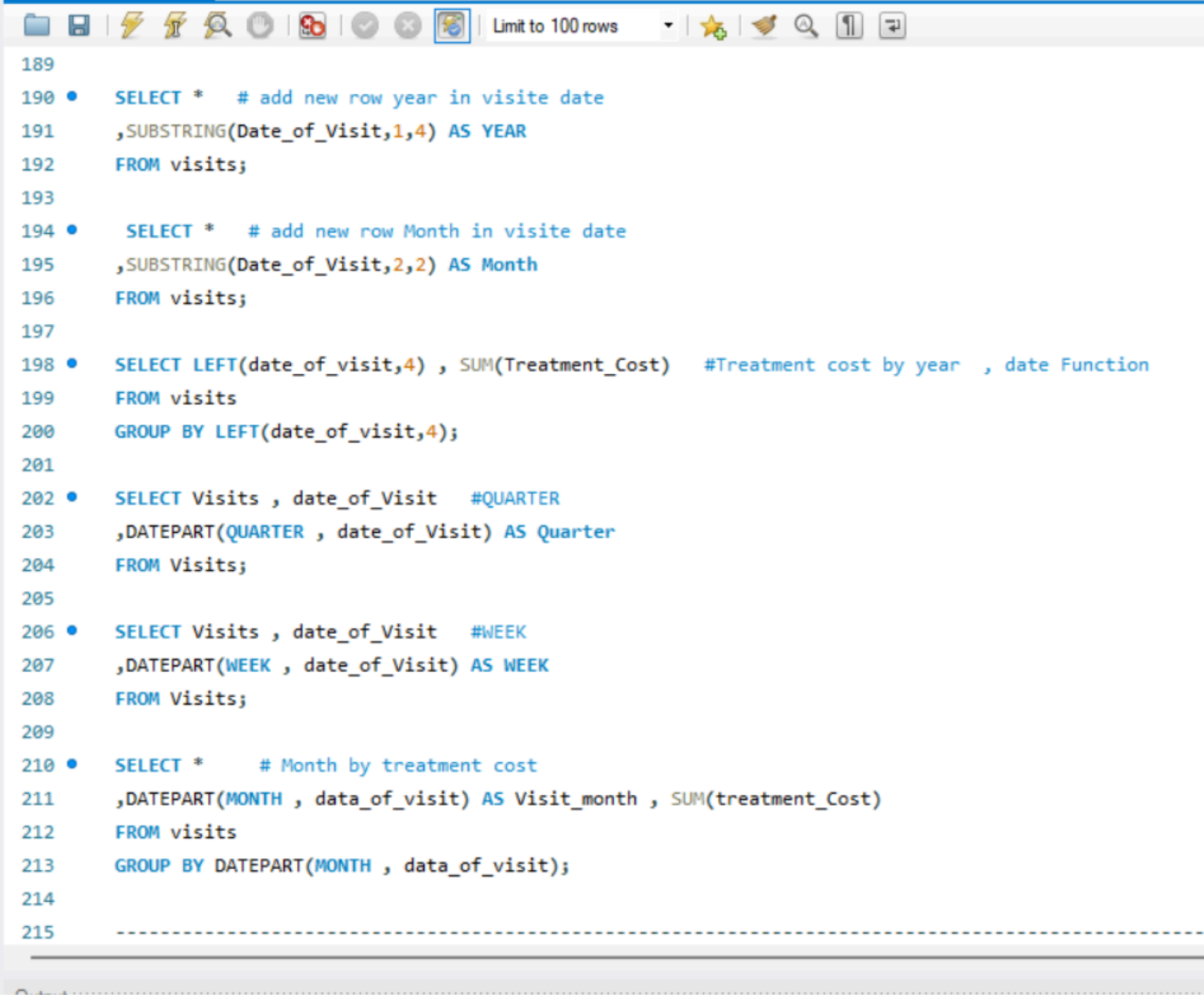
No object selected

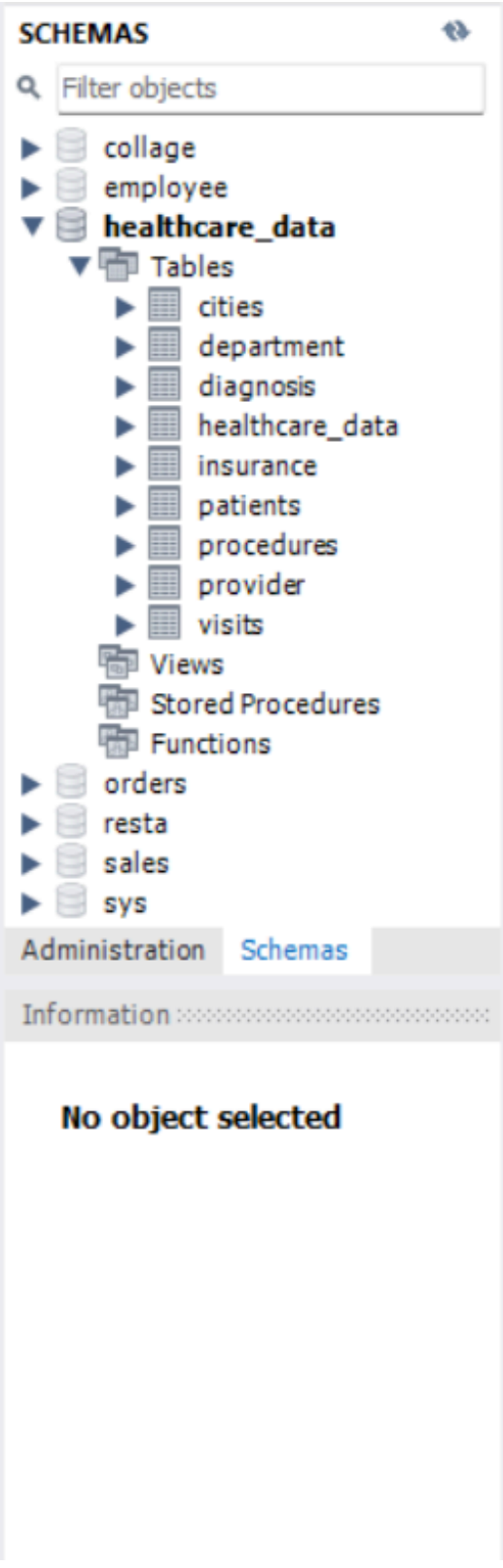
SQL Healthcar_data

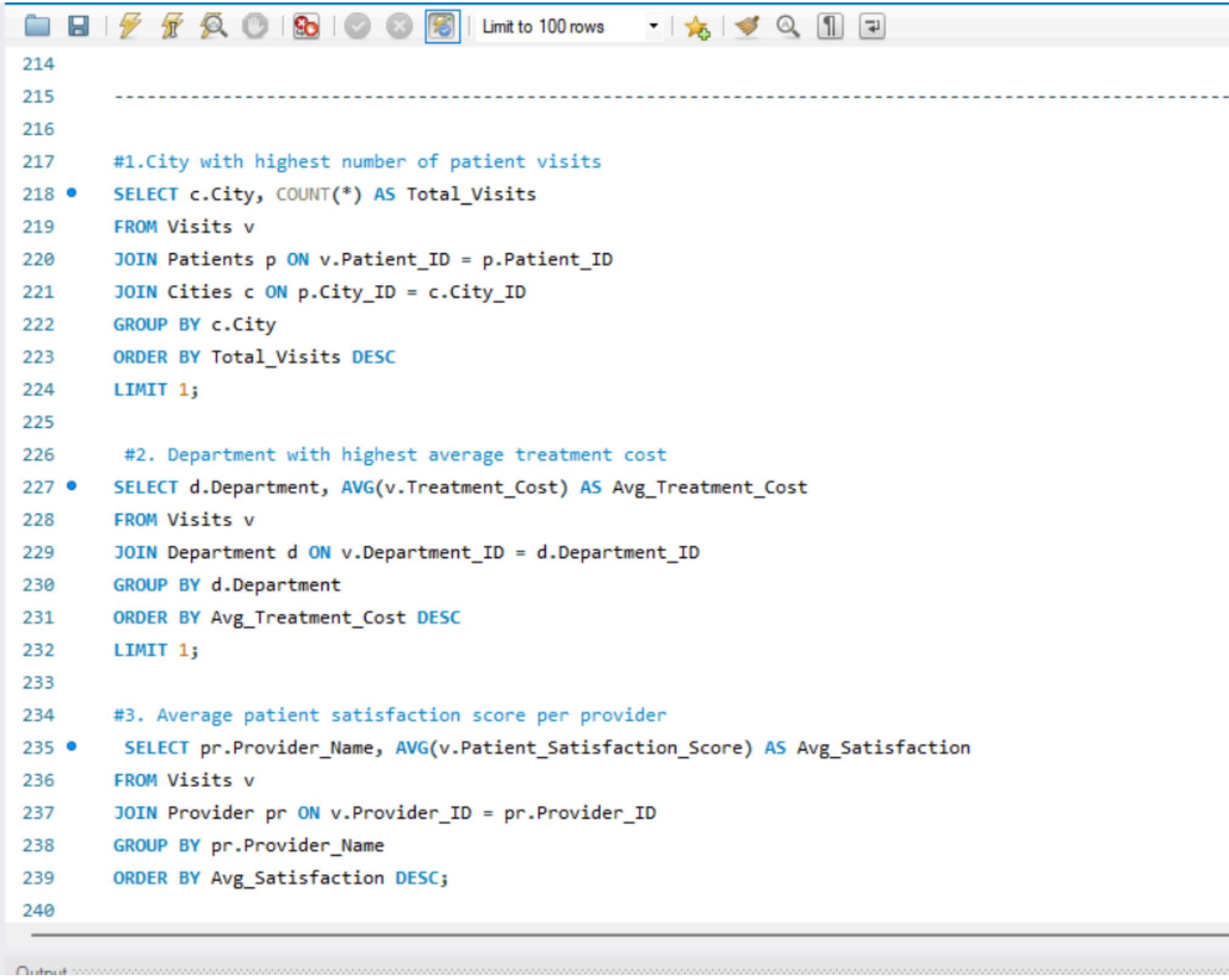
Limit to 100 rows

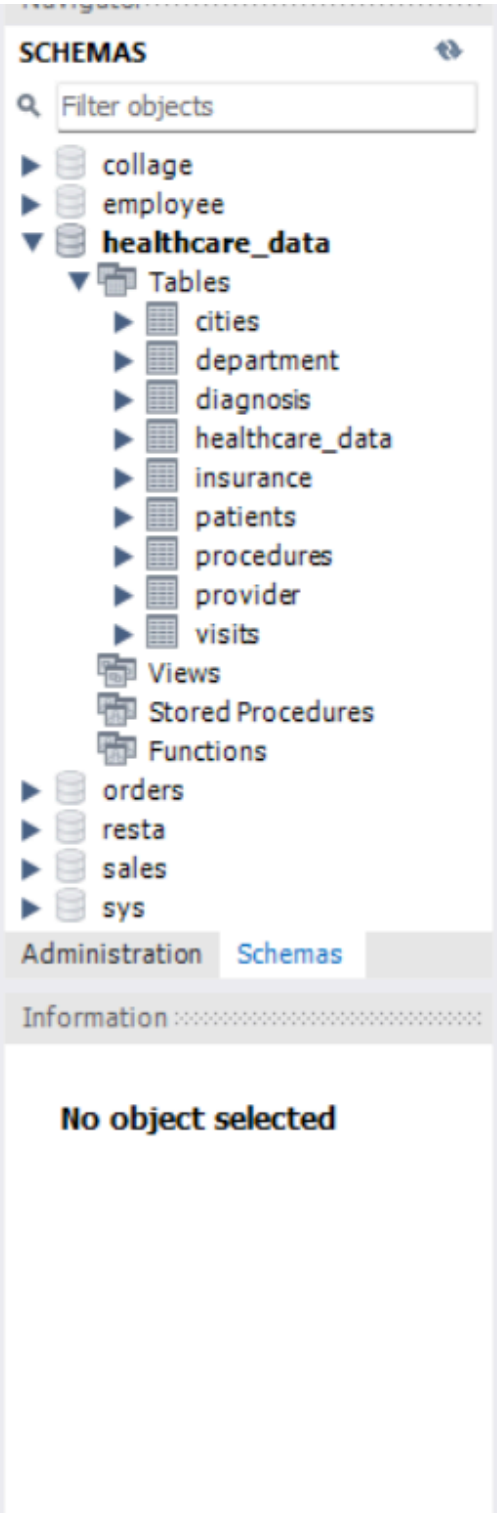
```
169
170 •  SELECT *                                  # add a new Column Profit Bucket
171    ,Case WHEN Treatment_Cost < 50 THEN 'Loss'
172    WHEN Treatment_Cost < 100 THEN 'Low Profit'
173    WHEN Treatment_Cost < 500 THEN 'Profit'
174    ELSE 'High Profit'
175    END AS Profit_Bucket
176    From visits;
177
178 •  SELECT *                                  # add a new Column Profit Bucket
179    ,Case WHEN Treatment_Cost < 50 THEN 'Loss'
180    WHEN Treatment_Cost BETWEEN 51 AND 100 THEN 'Low Profit'
181    WHEN Treatment_Cost BETWEEN 101 AND 499 THEN 'Profit'
182    ELSE 'High Profit'
183    END AS Profit_Bucket
184    From visits;
185
186 •  SELECT *    # Patients Name Lenght
187    ,LEN(Patient_Name) AS Patient_name_lenght
188    From patients;
189
190 •  SELECT *   # add new row year in visite date
191    ,SUBSTRING(Date_of_Visit,1,4) AS YEAR
192    FROM visits;
193
194 •   SELECT *   # add new row Month in visite date
195    ,SUBSTRING(Date_of_Visit,2,2) AS Month
```

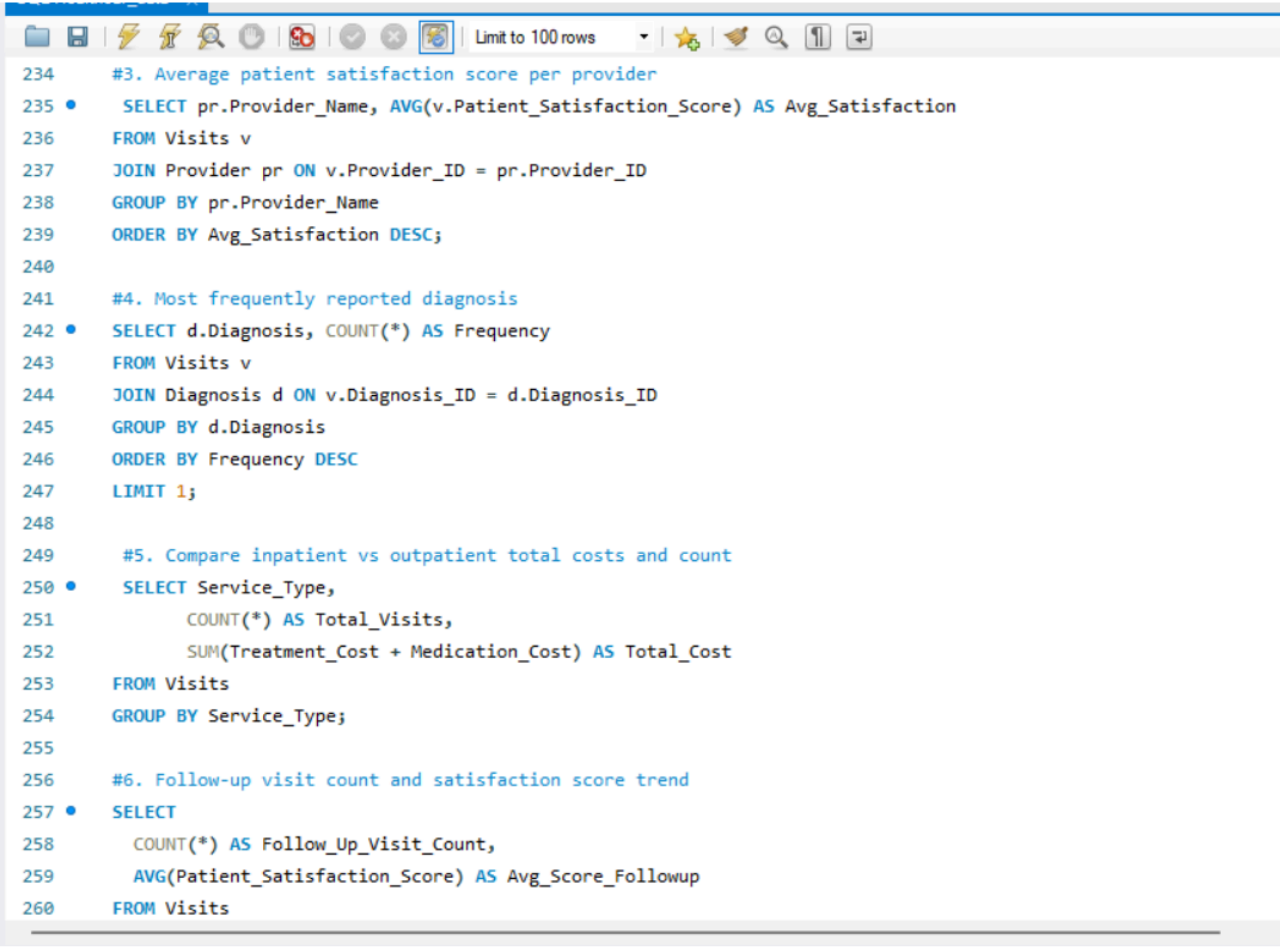

SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected

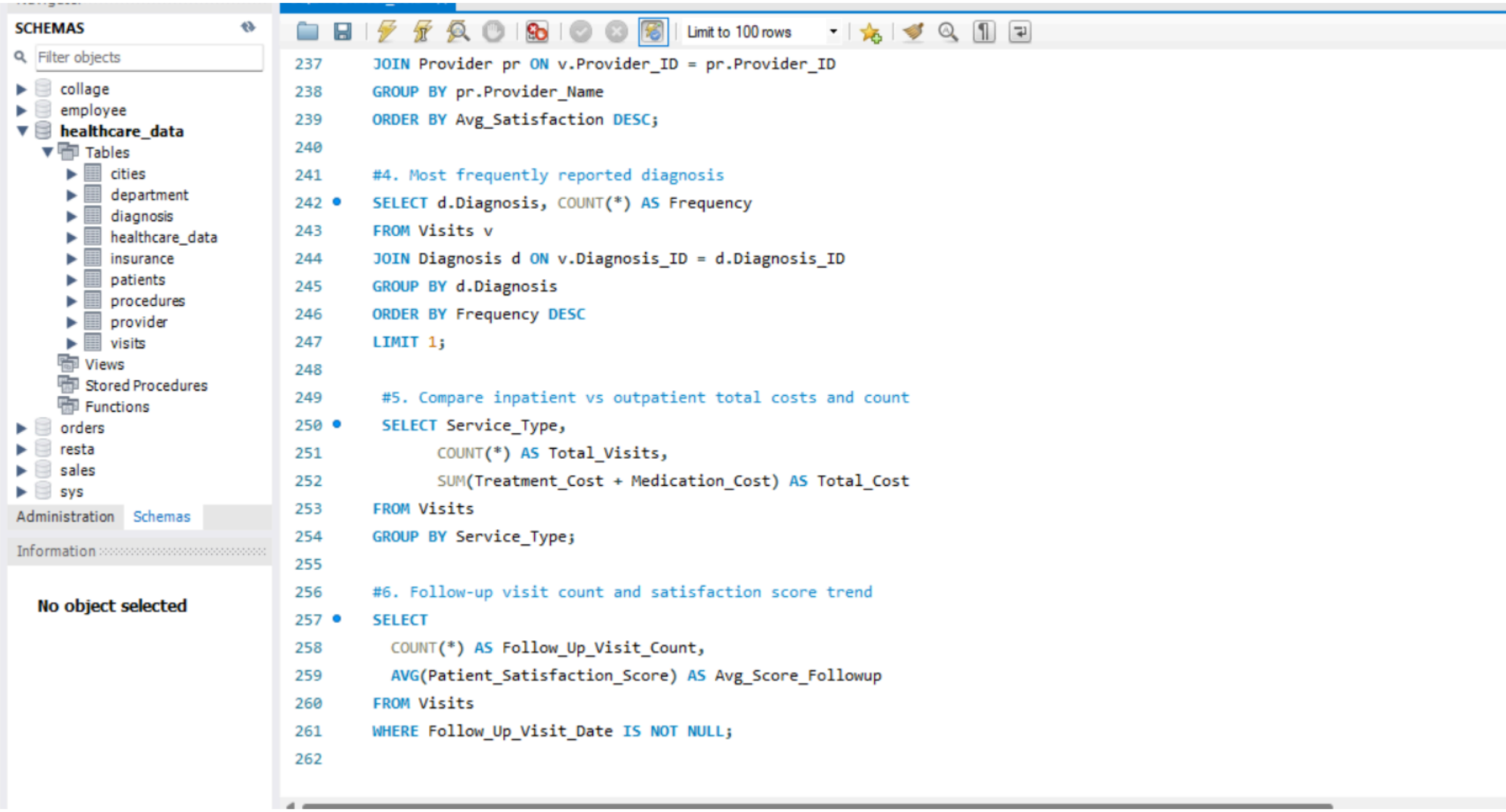

Limit to 100 rows
189
190 SELECT * # add new row year in visite date
191 ,SUBSTRING(Date_of_Visit,1,4) AS YEAR
192 FROM visits;
193
194 SELECT * # add new row Month in visite date
195 ,SUBSTRING(Date_of_Visit,2,2) AS Month
196 FROM visits;
197
198 SELECT LEFT(date_of_visit,4) , SUM(Treatment_Cost) #Treatment cost by year , date Function
199 FROM visits
200 GROUP BY LEFT(date_of_visit,4);
201
202 SELECT Visits , date_of_Visit #QUARTER
203 ,DATEPART(QUARTER , date_of_Visit) AS Quarter
204 FROM Visits;
205
206 SELECT Visits , date_of_Visit #WEEK
207 ,DATEPART(WEEK , date_of_Visit) AS WEEK
208 FROM Visits;
209
210 SELECT * # Month by treatment cost
211 ,DATEPART(MONTH , data_of_visit) AS Visit_month , SUM(treatment_Cost)
212 FROM visits
213 GROUP BY DATEPART(MONTH , data_of_visit);
214
215 ----------


SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected


Limit to 100 rows
214
215 ----------
216
217 #1.City with highest number of patient visits
218 SELECT c.City, COUNT(*) AS Total_Visits
219 FROM Visits v
220 JOIN Patients p ON v.Patient_ID = p.Patient_ID
221 JOIN Cities c ON p.City_ID = c.City_ID
222 GROUP BY c.City
223 ORDER BY Total_Visits DESC
224 LIMIT 1;
225
226 #2. Department with highest average treatment cost
227 SELECT d.Department, AVG(v.Treatment_Cost) AS Avg_Treatment_Cost
228 FROM Visits v
229 JOIN Department d ON v.Department_ID = d.Department_ID
230 GROUP BY d.Department
231 ORDER BY Avg_Treatment_Cost DESC
232 LIMIT 1;
233
234 #3. Average patient satisfaction score per provider
235 SELECT pr.Provider_Name, AVG(v.Patient_Satisfaction_Score) AS Avg_Satisfaction
236 FROM Visits v
237 JOIN Provider pr ON v.Provider_ID = pr.Provider_ID
238 GROUP BY pr.Provider_Name
239 ORDER BY Avg_Satisfaction DESC;
240
Output


SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected


Limit to 100 rows
234 #3. Average patient satisfaction score per provider
235 SELECT pr.Provider_Name, AVG(v.Patient_Satisfaction_Score) AS Avg_Satisfaction
236 FROM Visits v
237 JOIN Provider pr ON v.Provider_ID = pr.Provider_ID
238 GROUP BY pr.Provider_Name
239 ORDER BY Avg_Satisfaction DESC;
240
241 #4. Most frequently reported diagnosis
242 SELECT d.Diagnosis, COUNT(*) AS Frequency
243 FROM Visits v
244 JOIN Diagnosis d ON v.Diagnosis_ID = d.Diagnosis_ID
245 GROUP BY d.Diagnosis
246 ORDER BY Frequency DESC
247 LIMIT 1;
248
249 #5. Compare inpatient vs outpatient total costs and count
250 SELECT Service_Type,
251 COUNT(*) AS Total_Visits,
252 SUM(Treatment_Cost + Medication_Cost) AS Total_Cost
253 FROM Visits
254 GROUP BY Service_Type;
255
256 #6. Follow-up visit count and satisfaction score trend
257 SELECT
258 COUNT(*) AS Follow_Up_Visit_Count,
259 AVG(Patient_Satisfaction_Score) AS Avg_Score_Followup
260 FROM Visits


SCHEMAS
Filter objects
collage
employee
healthcare_data
Tables
cities
department
diagnosis
healthcare_data
insurance
patients
procedures
provider
visits
Views
Stored Procedures
Functions
orders
resta
sales
sys
Administration
Schemas
Information
No object selected
Limit to 100 rows
237 JOIN Provider pr ON v.Provider_ID = pr.Provider_ID
238 GROUP BY pr.Provider_Name
239 ORDER BY Avg_Satisfaction DESC;
240
241 #4. Most frequently reported diagnosis
242 SELECT d.Diagnosis, COUNT(*) AS Frequency
243 FROM Visits v
244 JOIN Diagnosis d ON v.Diagnosis_ID = d.Diagnosis_ID
245 GROUP BY d.Diagnosis
246 ORDER BY Frequency DESC
247 LIMIT 1;
248
249 #5. Compare inpatient vs outpatient total costs and count
250 SELECT Service_Type,
251 COUNT(*) AS Total_Visits,
252 SUM(Treatment_Cost + Medication_Cost) AS Total_Cost
253 FROM Visits
254 GROUP BY Service_Type;
255
256 #6. Follow-up visit count and satisfaction score trend
257 SELECT
258 COUNT(*) AS Follow_Up_Visit_Count,
259 AVG(Patient_Satisfaction_Score) AS Avg_Score_Followup
260 FROM Visits
261 WHERE Follow_Up_Visit_Date IS NOT NULL;
262