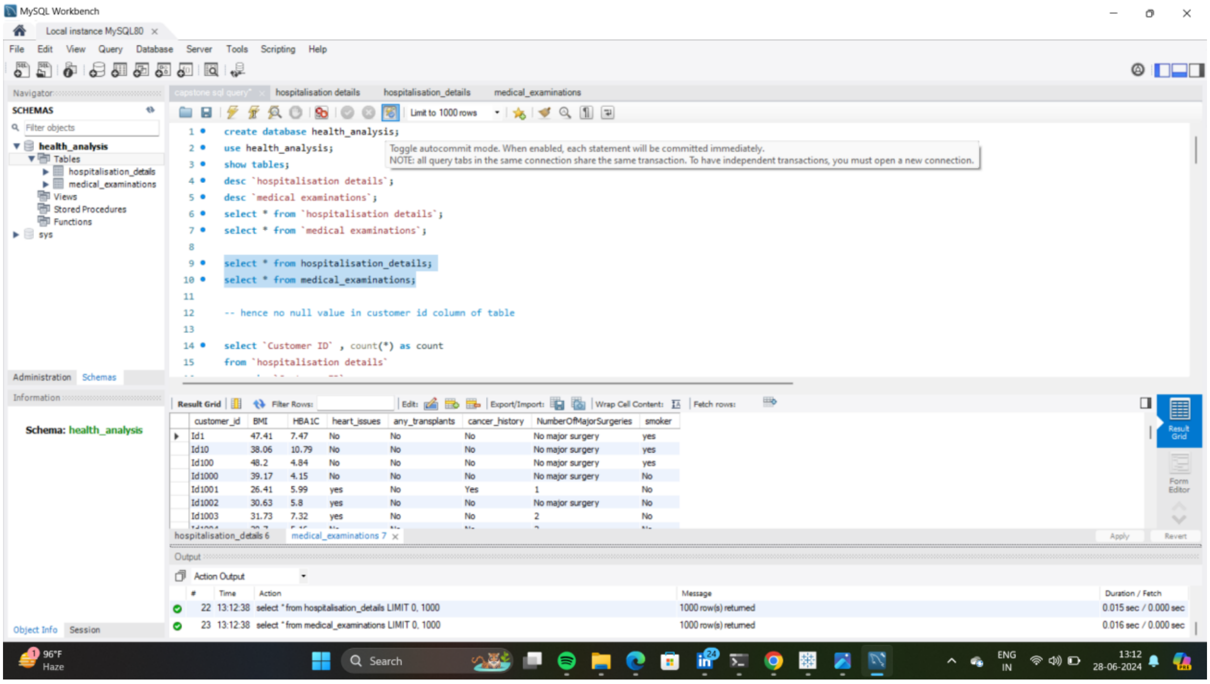
```
create database health_analysis;

use health_analysis;

show tables;

desc `hospitalisation details`;

desc `medical examinations`;

select * from `hospitalisation details`;

select * from `medical examinations`;


select * from hospitalisation_details;

select * from medical_examinations;
```



```
create database health_analysis;
use health_analysis;
show tables;
desc `hospitalisation details`;
desc `medical examinations`;
select * from `hospitalisation details`;
select * from `medical examinations`;

select * from hospitalisation_details;
select * from medical_examinations;

-- hence no null value in customer id column of table

select `Customer ID` , count(*) as count
from `hospitalisation details`
```

Toggle autocommit mode. When enabled, each statement will be committed immediately.
NOTE: all query tabs in the same connection share the same transaction. To have independent transactions, you must open a new connection.

customer_id	BMI	HBA1C	heart_issues	any_transplants	cancer_history	NumberOfMajorSurgeries	smoker
Id1	47.41	7.47	No	No	No	No major surgery	yes
Id10	38.06	10.79	No	No	No	No major surgery	yes
Id100	48.2	4.84	No	No	No	No major surgery	yes
Id1000	39.17	4.15	No	No	No	No major surgery	No
Id1001	26.41	5.99	yes	No	Yes	1	No
Id1002	30.63	5.8	yes	No	No	No major surgery	No
Id1003	31.73	7.32	yes	No	No	2	No

```
#   Time      Action                                               Message                 Duration / Fetch
22  13:12:38  select * from hospitalisation_details LIMIT 0, 1000  1000 row(s) returned    0.015 sec / 0.000 sec
23  13:12:38  select * from medical_examinations LIMIT 0, 1000     1000 row(s) returned    0.016 sec / 0.000 sec
```

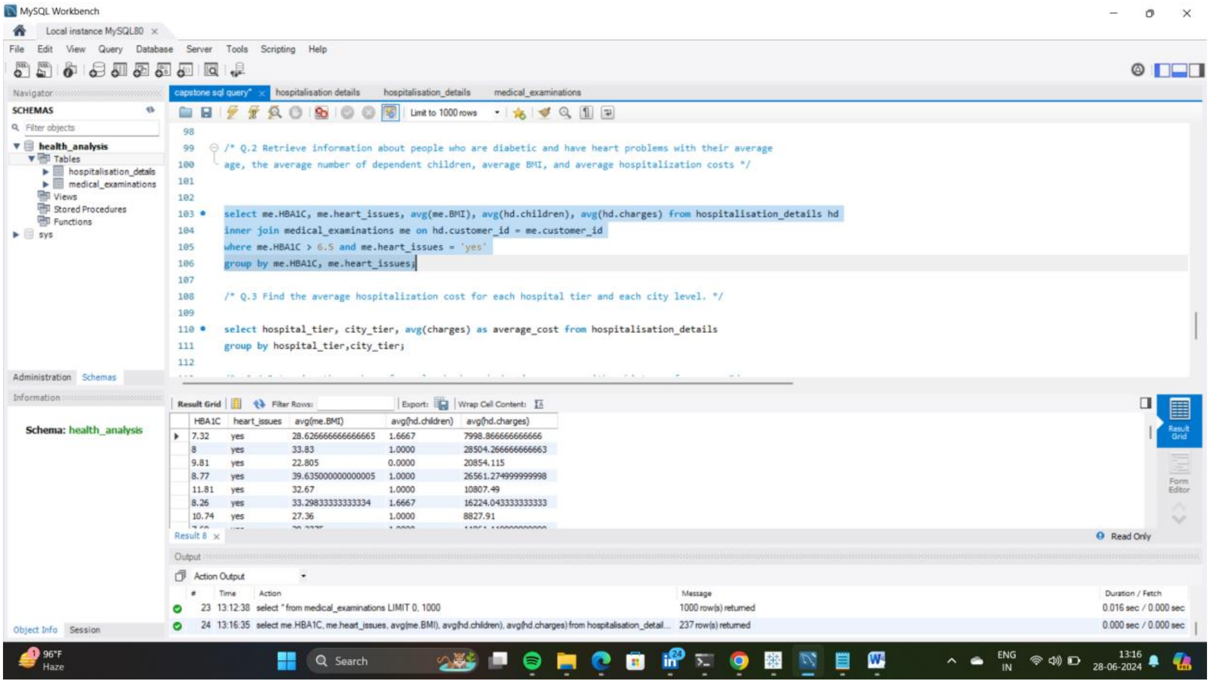
```
-- hence no null value in customer id column of table


select `Customer ID` , count(*) as count

from `hospitalisation details`

group by `Customer ID`

having count > 1;


SET SQL_SAFE_UPDATES = 0;
```

```sql
delete from `hospitalisation details`
where `Customer ID` = '?';

select `Customer ID` from `hospitalisation details`
where `Customer ID`= '?';

-- know we conclude that no null value and no duplicate values

desc `hospitalisation details`;

-- Rename the table

RENAME TABLE `hospitalisation details` TO hospitalisation_details;

RENAME TABLE `medical examinations` TO medical_examinations;

desc hospitalisation_details;
desc medical_examinations;

-- rename column to remove space between them for a better analysis

alter table hospitalisation_details
rename column `Customer ID` to customer_id;

alter table hospitalisation_details
modify column customer_id varchar(255);

alter table hospitalisation_details
rename column `Hospital tier` to hospital_tier;
```

Healthcare Insurance Analysis

```
alter table hospitalisation_details
rename column `City tier` to city_tier;

alter table hospitalisation_details
rename column `State ID` to state_id;

desc hospitalisation_details;

select * from medical_examinations;

alter table medical_examinations
rename column `Customer ID` to customer_id;

alter table medical_examinations
modify column customer_id varchar(225);

alter table medical_examinations
rename column `Heart Issues` to heart_issues;

alter table medical_examinations
rename column `Any Transplants` to any_transplants;

alter table medical_examinations
rename column `Cancer history` to cancer_history;

-- Adding primary key to customer by checking null and duplicate values
```

```
select customer_id
from hospitalisation_details
where customer_id is null;

-- not found Null value in customer_id

-- apply primary key function

alter table hospitalisation_details
add constraint pk_customer_id primary key (customer_id);

alter table medical_examinations
add constraint pk_customer_id primary key (customer_id);

-- merging the tables

select * from hospitalisation_details hd
inner join medical_examinations me
on hd.customer_id = me.customer_id;


/* Q.2 Retrieve information about people who are diabetic and have heart problems with their average
age, the average number of dependent children, average BMI, and average hospitalization costs */


select me.HBA1C, me.heart_issues, avg(me.BMI), avg(hd.children), avg(hd.charges) from
hospitalisation_details hd
inner join medical_examinations me on hd.customer_id = me.customer_id
where me.HBA1C > 6.5 and me.heart_issues = 'yes'
group by me.HBA1C, me.heart_issues;
```



```
/* Q.2 Retrieve information about people who are diabetic and have heart problems with their average
age, the average number of dependent children, average BMI, and average hospitalization costs */


select me.HBA1C, me.heart_issues, avg(me.BMI), avg(hd.children), avg(hd.charges) from hospitalisation_details hd
inner join medical_examinations me on hd.customer_id = me.customer_id
where me.HBA1C > 6.5 and me.heart_issues = 'yes'
group by me.HBA1C, me.heart_issues;

/* Q.3 Find the average hospitalization cost for each hospital tier and each city level. */

select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details
group by hospital_tier,city_tier;
```

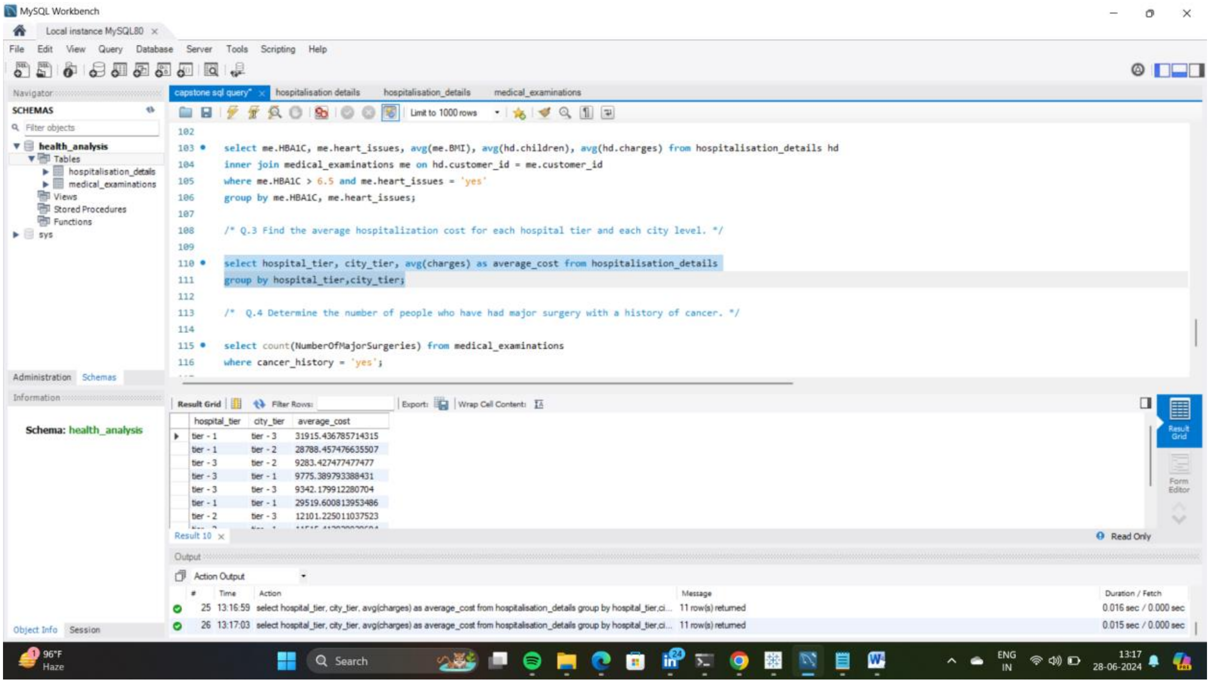
HBA1C	heart_issues	avg(me.BMI)	avg(hd.children)	avg(hd.charges)
7.32	yes	28.626666666666665	1.6667	7998.866666666666
8	yes	33.83	1.0000	28504.266666666663
9.81	yes	22.805	0.0000	20854.115
8.77	yes	39.635000000000005	1.0000	26561.274999999998
11.81	yes	32.67	1.0000	10807.49
8.26	yes	33.29833333333334	1.6667	16224.043333333333
10.74	yes	27.36	1.0000	8827.91

```
#   Time      Action                                                                                          Message              Duration / Fetch
23  13:12:38  select * from medical_examinations LIMIT 0, 1000                                                1000 row(s) returned  0.016 sec / 0.000 sec
24  13:16:35  select me.HBA1C, me.heart_issues, avg(me.BMI), avg(hd.children), avg(hd.charges) from hospitalisation_detail...  237 row(s) returned  0.000 sec / 0.000 sec
```

/* Q.3 Find the average hospitalization cost for each hospital tier and each city level. */

select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details

group by hospital_tier,city_tier;



```
select me.HBA1C, me.heart_issues, avg(me.BMI), avg(hd.children), avg(hd.charges) from hospitalisation_details hd
inner join medical_examinations me on hd.customer_id = me.customer_id
where me.HBA1C > 6.5 and me.heart_issues = 'yes'
group by me.HBA1C, me.heart_issues;

/* Q.3 Find the average hospitalization cost for each hospital tier and each city level. */

select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details
group by hospital_tier,city_tier;

/*  Q.4 Determine the number of people who have had major surgery with a history of cancer. */

select count(NumberOfMajorSurgeries) from medical_examinations
where cancer_history = 'yes';
```

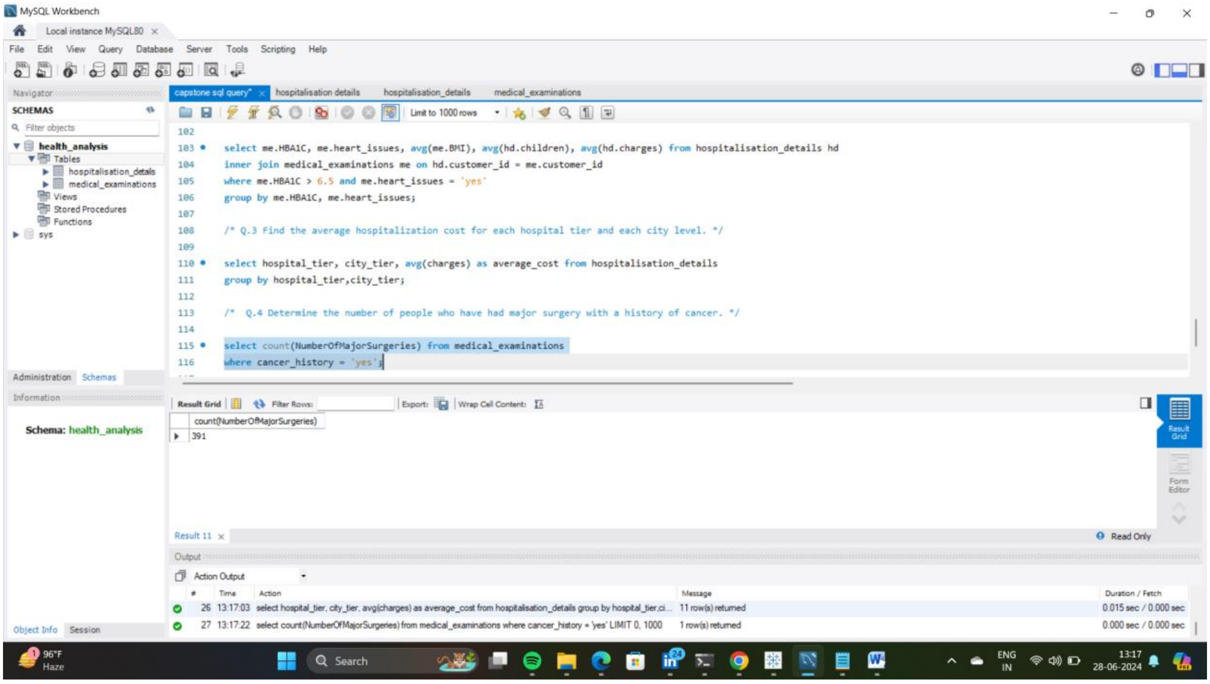
hospital_tier	city_tier	average_cost
tier - 1	tier - 3	31915.436785714315
tier - 1	tier - 2	28788.457476635507
tier - 3	tier - 2	9283.427477477477
tier - 3	tier - 1	9775.389793388431
tier - 3	tier - 3	9342.179912280704
tier - 1	tier - 1	29519.600813953486
tier - 2	tier - 3	12101.225011037523

```
#   Time      Action                                                                                          Message             Duration / Fetch
25  13:16:59  select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details group by hospital_tier,ci...  11 row(s) returned  0.016 sec / 0.000 sec
26  13:17:03  select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details group by hospital_tier,ci...  11 row(s) returned  0.015 sec / 0.000 sec
```

/* Q.4 Determine the number of people who have had major surgery with a history of cancer. */

select count(NumberOfMajorSurgeries) from medical_examinations

where cancer_history = 'yes';



```
select me.HBA1C, me.heart_issues, avg(me.BMI), avg(hd.children), avg(hd.charges) from hospitalisation_details hd
inner join medical_examinations me on hd.customer_id = me.customer_id
where me.HBA1C > 6.5 and me.heart_issues = 'yes'
group by me.HBA1C, me.heart_issues;

/* Q.3 Find the average hospitalization cost for each hospital tier and each city level. */

select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details
group by hospital_tier,city_tier;

/*  Q.4 Determine the number of people who have had major surgery with a history of cancer. */

select count(NumberOfMajorSurgeries) from medical_examinations
where cancer_history = 'yes';
```

count(NumberOfMajorSurgeries)
391

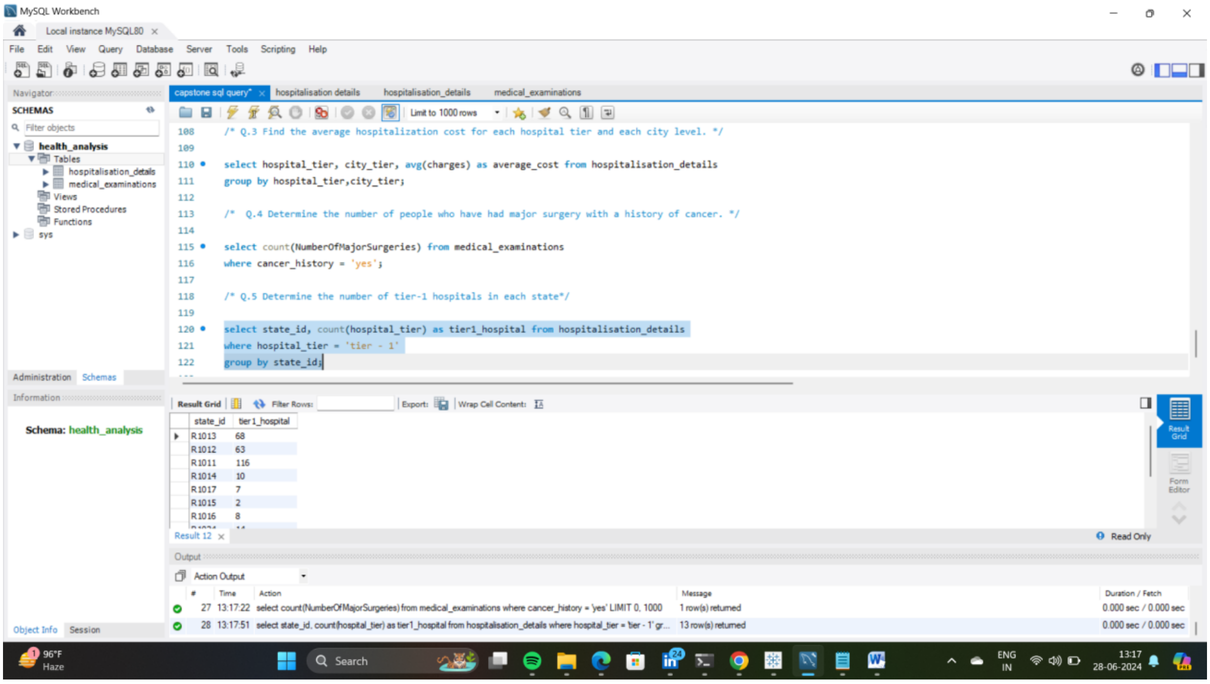
```
#   Time      Action                                                                                                   Message             Duration / Fetch
26  13:17:03  select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details group by hospital_tier,ci...  11 row(s) returned  0.015 sec / 0.000 sec
27  13:17:22  select count(NumberOfMajorSurgeries) from medical_examinations where cancer_history = 'yes' LIMIT 0, 1000      1 row(s) returned   0.000 sec / 0.000 sec
```

/* Q.5 Determine the number of tier-1 hospitals in each state*/

select state_id, count(hospital_tier) as tier1_hospital from hospitalisation_details

where hospital_tier = 'tier - 1'

group by state_id;



```
/* Q.3 Find the average hospitalization cost for each hospital tier and each city level. */

select hospital_tier, city_tier, avg(charges) as average_cost from hospitalisation_details
group by hospital_tier,city_tier;

/*  Q.4 Determine the number of people who have had major surgery with a history of cancer. */

select count(NumberOfMajorSurgeries) from medical_examinations
where cancer_history = 'yes';

/* Q.5 Determine the number of tier-1 hospitals in each state*/

select state_id, count(hospital_tier) as tier1_hospital from hospitalisation_details
where hospital_tier = 'tier - 1'
group by state_id;
```

state_id	tier1_hospital
R1013	68
R1012	63
R1011	116
R1014	10
R1017	7
R1015	2
R1016	8

```
#   Time      Action                                                                                                  Message              Duration / Fetch
27  13:17:22  select count(NumberOfMajorSurgeries) from medical_examinations where cancer_history = 'yes' LIMIT 0, 1000   1 row(s) returned    0.000 sec / 0.000 sec
28  13:17:51  select state_id, count(hospital_tier) as tier1_hospital from hospitalisation_details where hospital_tier = 'tier - 1' gr...   13 row(s) returned   0.000 sec / 0.000 sec
```

commit;

------------- END --------------