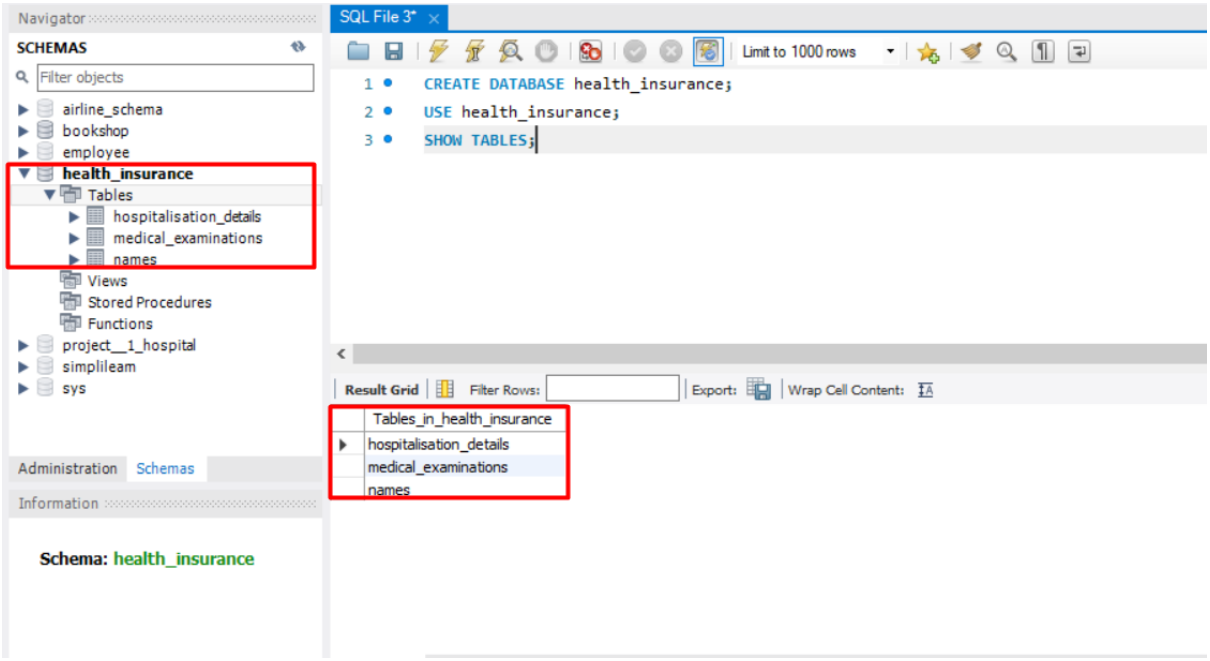
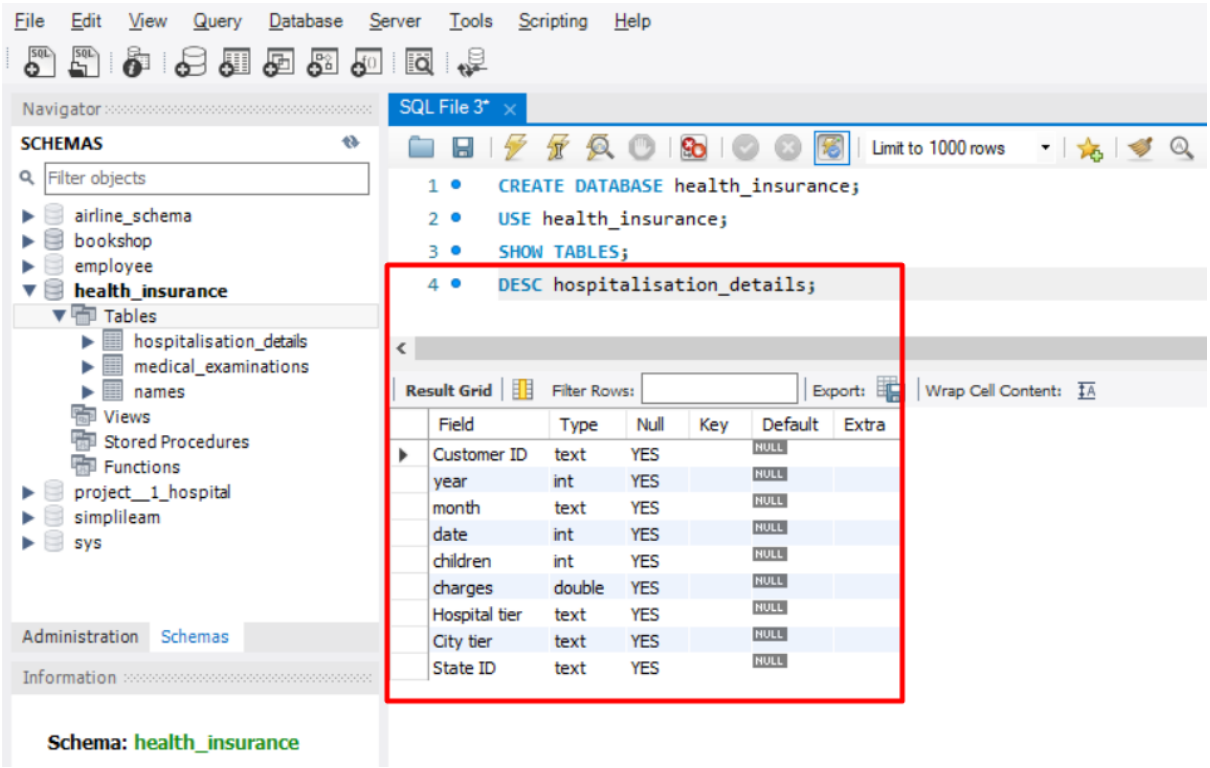
1. Created the database and imported the tables



2. Checking the columns in the table 'hospitalisation_details'

Code: DESC hospitalisation_details;

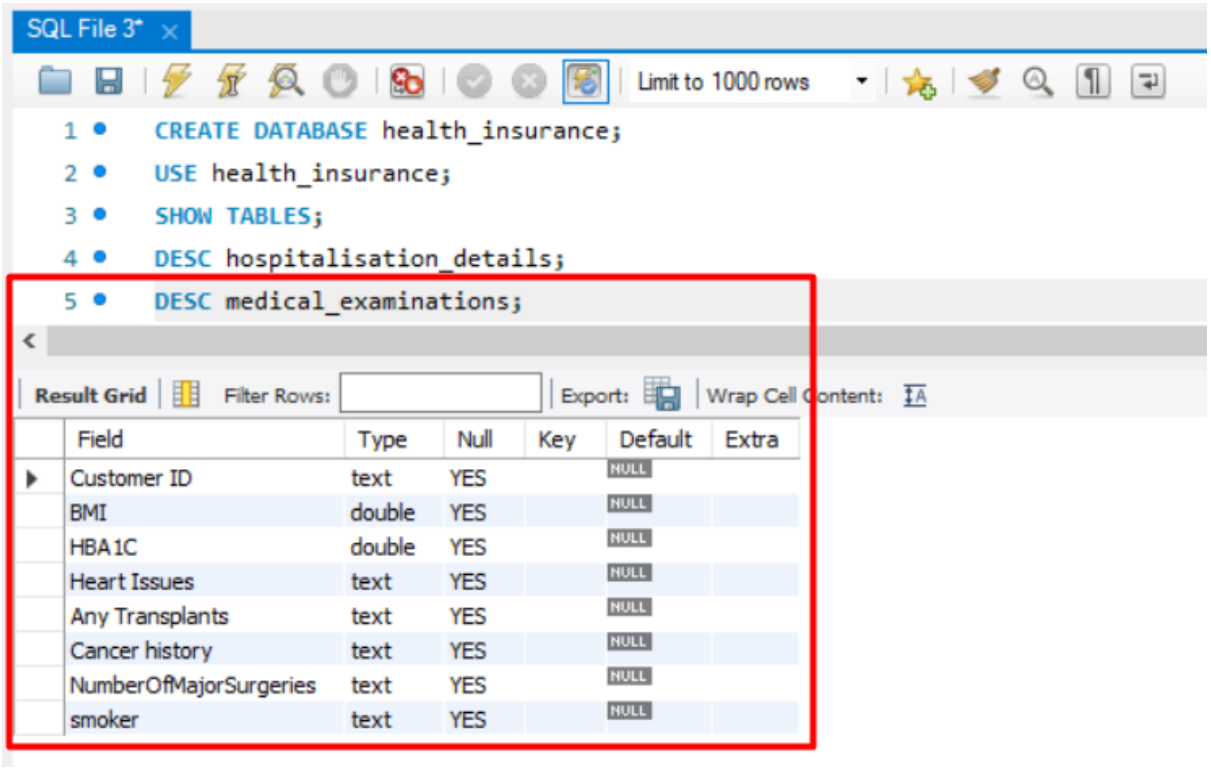
OUTPUT:



3. Checking the columns in the table ‘ medical_examinations ’

   Code: DESC medical_examinations;

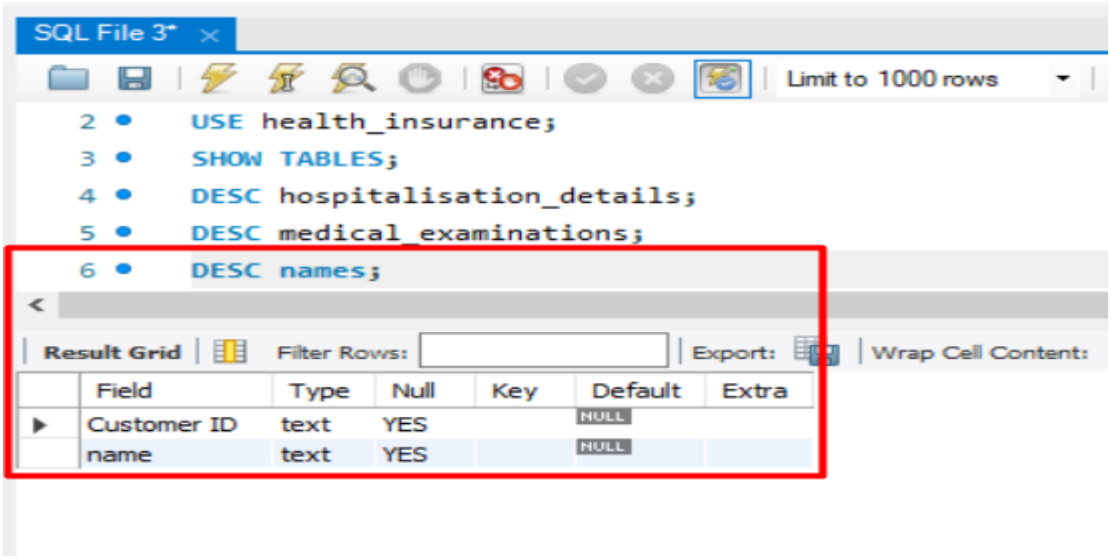
   OUTPUT:



4. Checking the columns in the table ‘ names ’

   Code: DESC names;

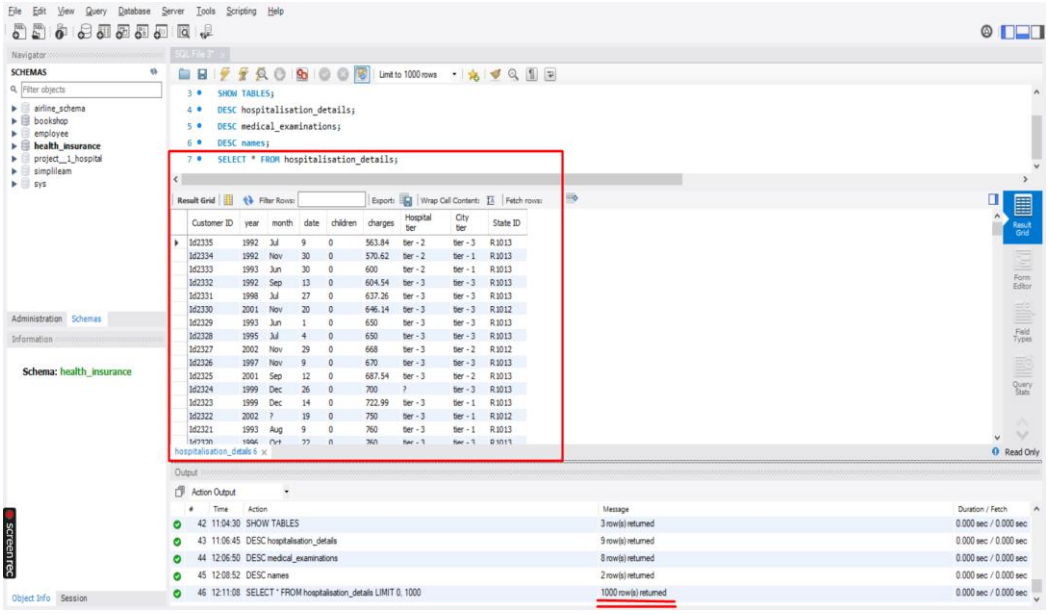
   OUTPUT



5. Checking the values in ‘hospitalisation_details’ table

   CODE: SELECT * FROM hospitalisation_details;

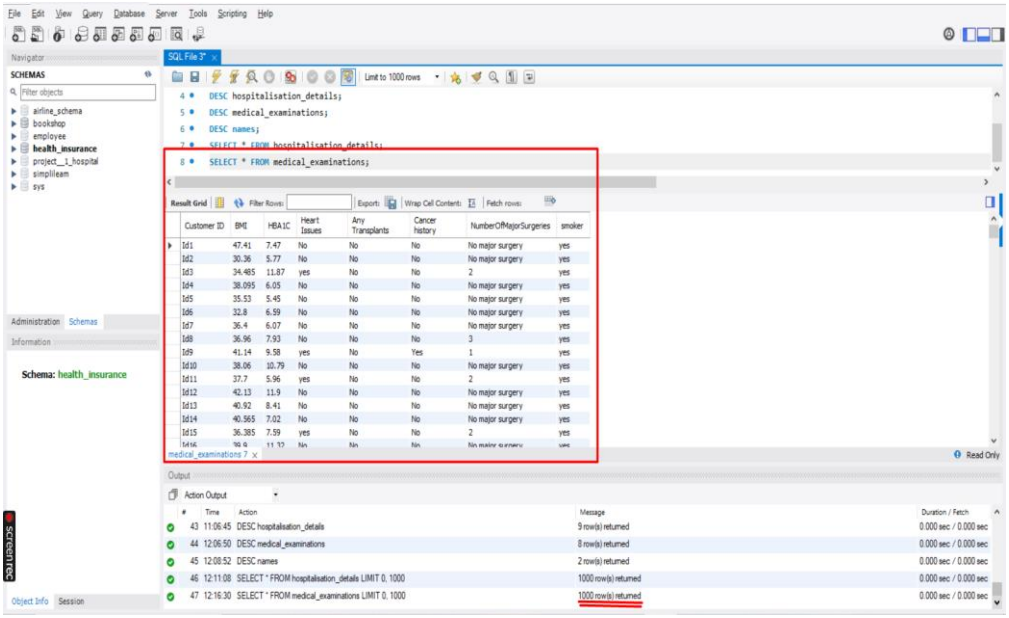
   OUTPUT:



6. Checking the values in ‘medical_examinations’ table

   CODE: SELECT * FROM medical_examinations;

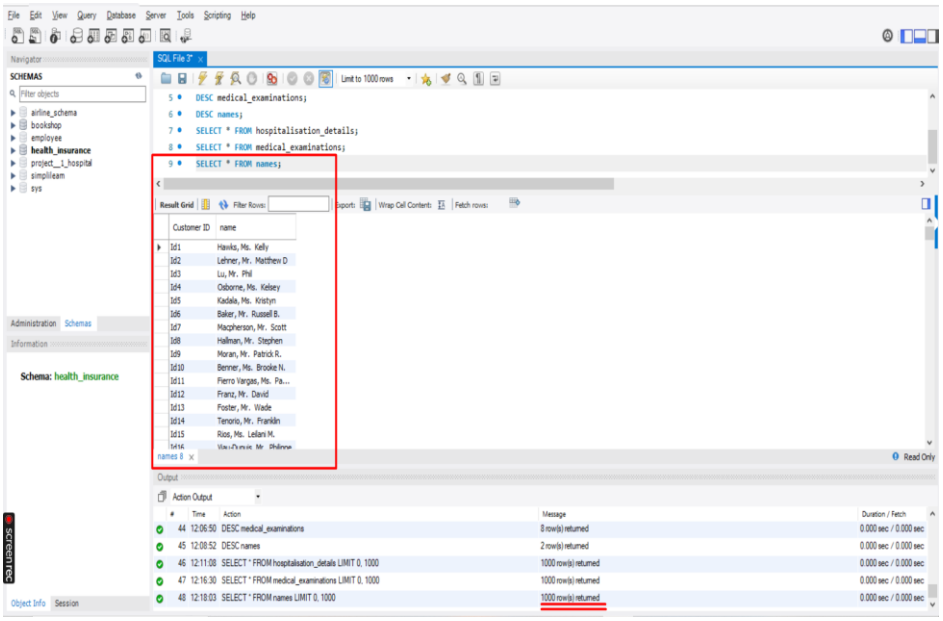
   OUTPUT:



7. Checking the values in ‘names’ table

   CODE:

OUTPUT:



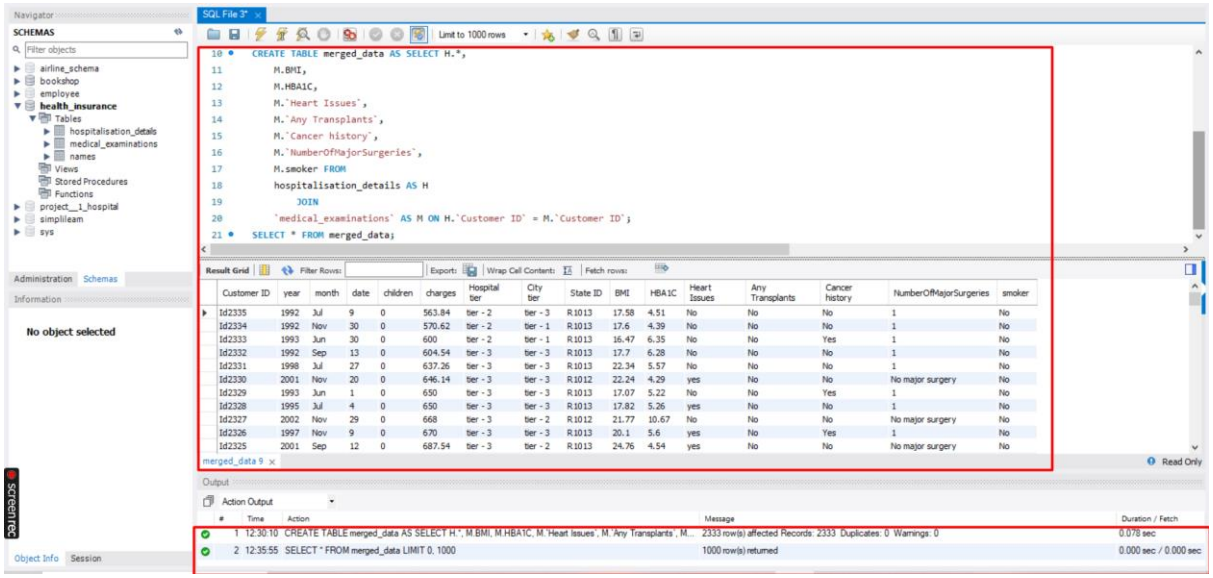
TASK: 1

STEP 1: Merge the two tables by first identifying the columns in the data tables

CODE:

```
CREATE TABLE merged_data AS
        SELECT H.*,
                M.BMI,
                 M.HBA1C,
                M.`Heart Issues`,
                M.`Any Transplants`,
                 M.`Cancer history`,
                 M.`NumberOfMajorSurgeries`,
                M.smoker
        FROM hospitalisation_details AS H
JOIN `medical_examinations` AS M ON H.`Customer ID` = M.`Customer ID`;
SELECT * FROM merged_data;
```

OUTPUT:



STEP 2: Add a Primary Key constraint for these columns

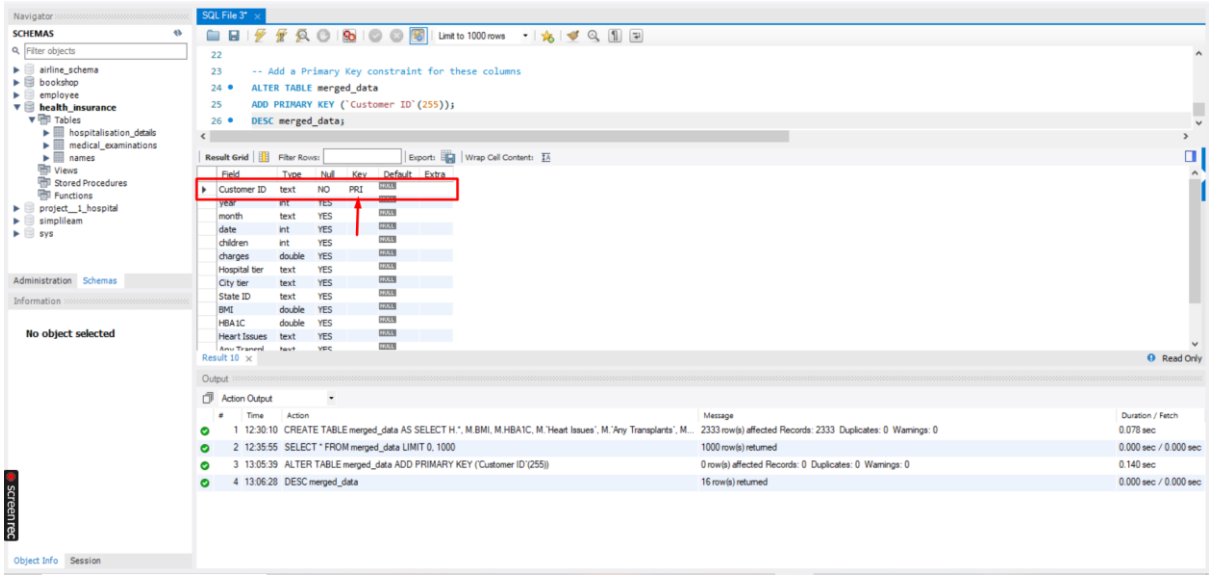
CODE:

ALTER TABLE merged_data

ADD PRIMARY KEY (`Customer ID`(255));

DESC merged_data;

OUTPUT:

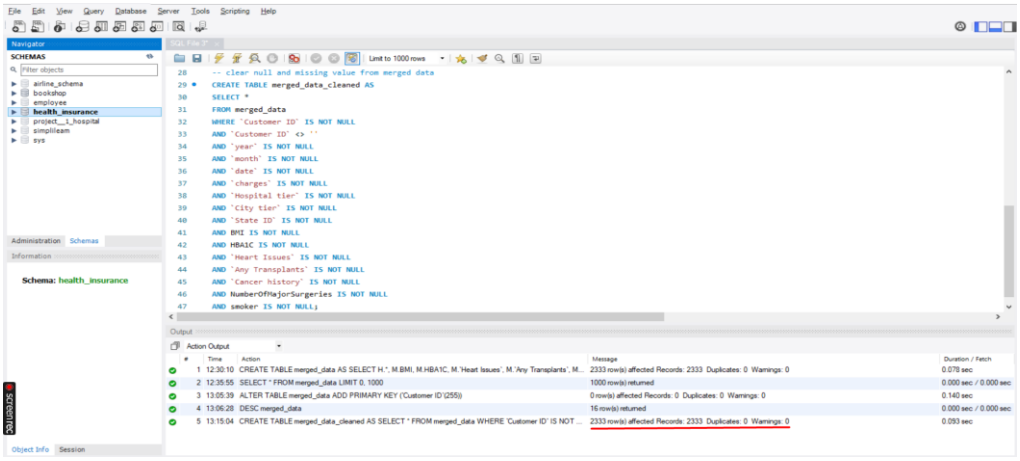


STEP 3: clear null and missing value from merged data

CODE:

```
CREATE TABLE merged_data_cleaned AS
SELECT *
FROM merged_data
WHERE `Customer ID` IS NOT NULL
AND `Customer ID` <> ''
AND `year` IS NOT NULL
AND `month` IS NOT NULL
AND `date` IS NOT NULL
AND `charges` IS NOT NULL
AND `Hospital tier` IS NOT NULL
AND `City tier` IS NOT NULL
AND `State ID` IS NOT NULL
AND BMI IS NOT NULL
AND HBA1C IS NOT NULL
AND `Heart Issues` IS NOT NULL
AND `Any Transplants` IS NOT NULL
AND `Cancer history` IS NOT NULL
AND NumberOfMajorSurgeries IS NOT NULL
AND smoker IS NOT NULL;
```

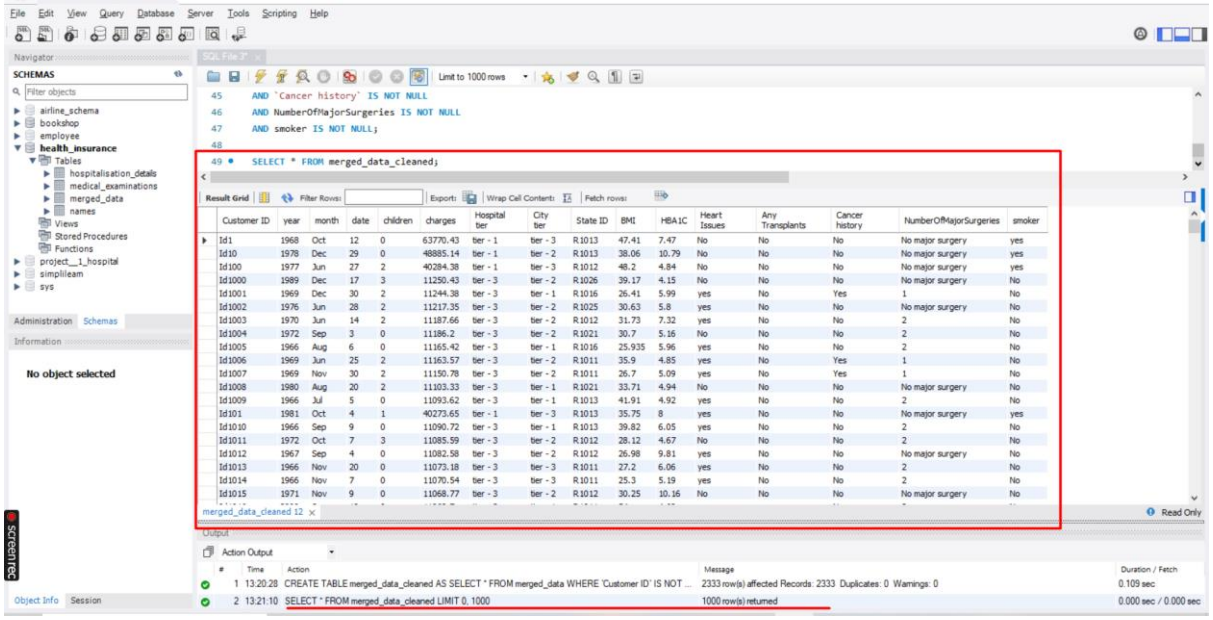
OUTPUT:



CODE:

SELECT * FROM merged_data_cleaned;

OUTPUT:

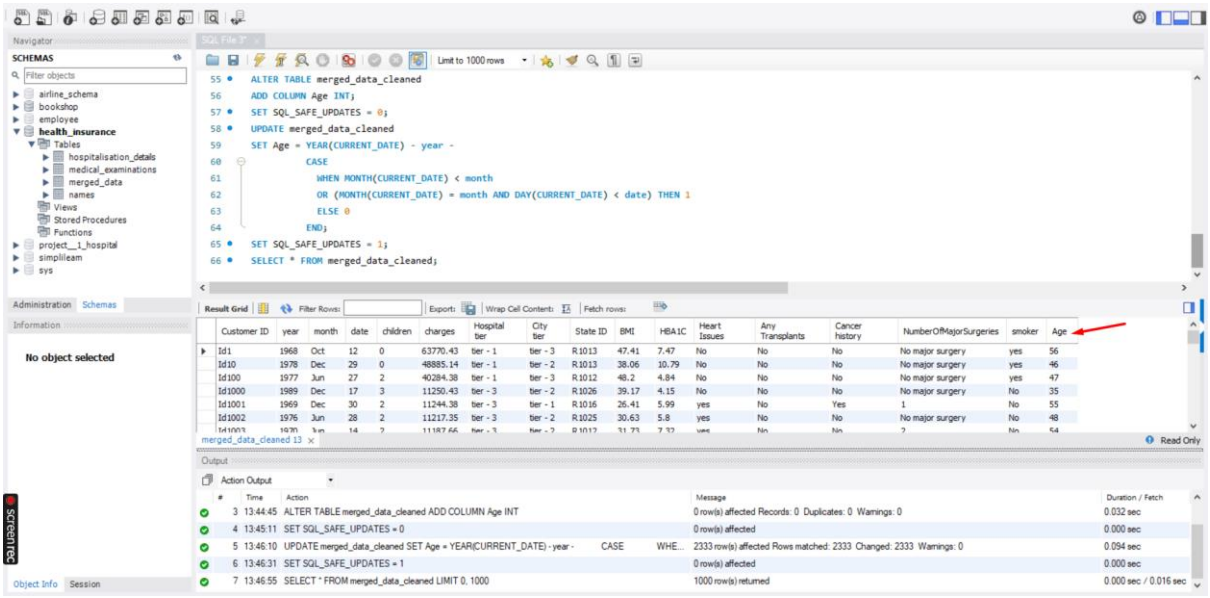


STEP 4:

CODE:

```
ALTER TABLE merged_data_cleaned
ADD COLUMN Age INT;
SET SQL_SAFE_UPDATES = 0;
UPDATE merged_data_cleaned
SET Age = YEAR(CURRENT_DATE) - year -
        CASE
          WHEN MONTH(CURRENT_DATE) < month
          OR (MONTH(CURRENT_DATE) = month AND DAY(CURRENT_DATE) < date) THEN 1
          ELSE 0
        END;
SET SQL_SAFE_UPDATES = 1;
SELECT * FROM merged_data_cleaned;
```

OUTPUT:

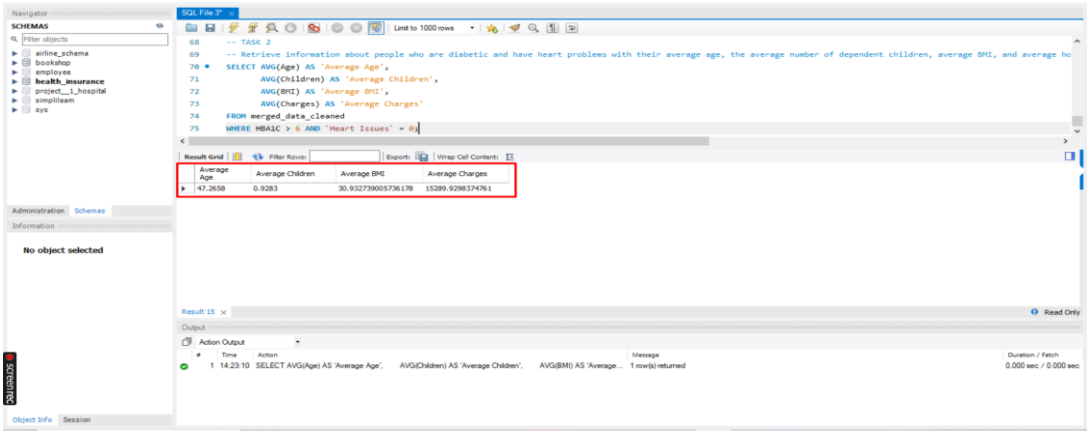


TASK 2: Retrieve information about people who are diabetic and have heart problems with their average age, the average number of dependent children, average BMI, and average hospitalization costs

CODE:

```
SELECT AVG(Age) AS 'Average Age',
        AVG(Children) AS 'Average Children',
        AVG(BMI) AS 'Average BMI',
        AVG(Charges) AS 'Average Charges'
FROM merged_data_cleaned
WHERE HBA1C > 6 AND `Heart Issues` = 0;
```

OUTPUT:

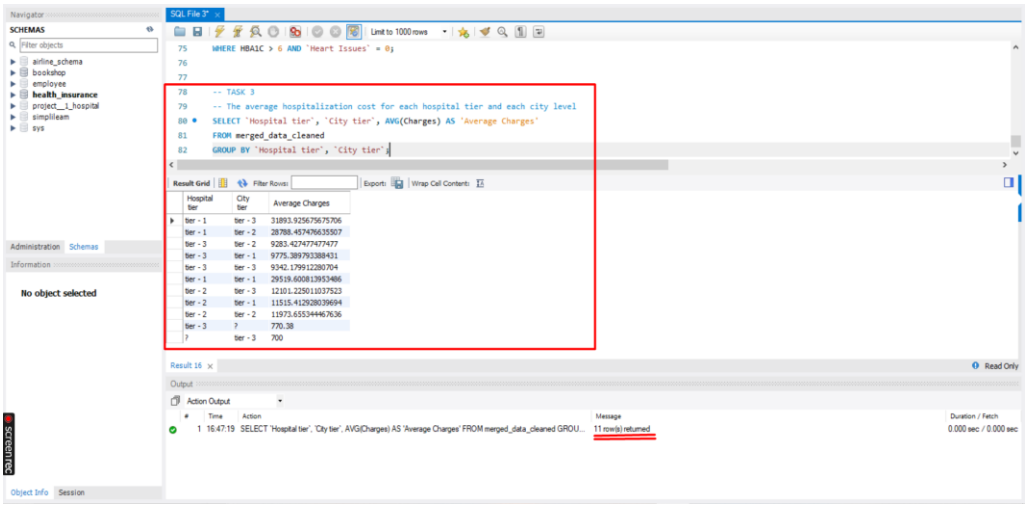


TASK 3: The average hospitalization cost for each hospital tier and each city level

CODE:

```
SELECT `Hospital tier`, `City tier`, AVG(Charges) AS 'Average Charges'
FROM merged_data_cleaned
GROUP BY `Hospital tier`, `City tier`;
```

OUTPUT:

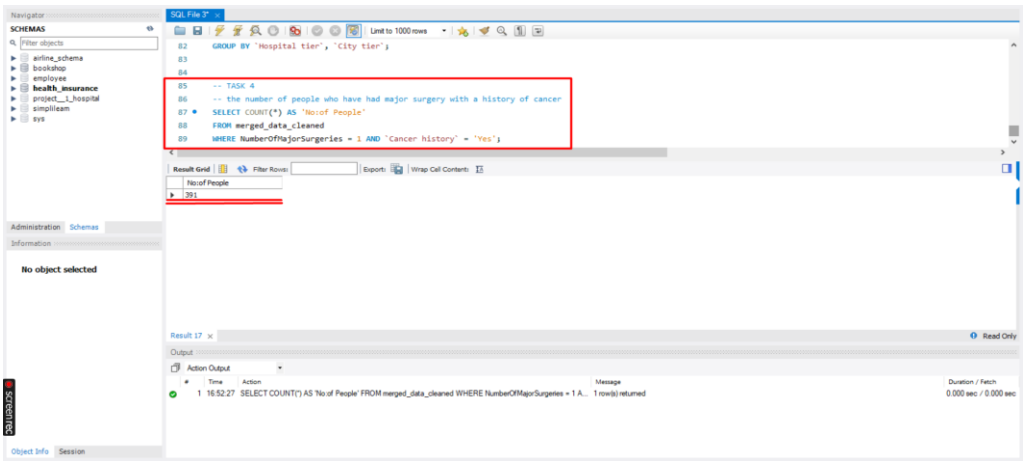


TASK 4: The number of people who have had major surgery with a history of cancer

CODE:

```
SELECT COUNT(*) AS 'No:of People'
FROM merged_data_cleaned
WHERE NumberOfMajorSurgeries = 1 AND `Cancer history` = 'Yes';
```

OUTPUT:



TASK 5: The number of tier-1 hospitals in each state

CODE:

```
SELECT `State ID`, COUNT(*) AS 'No:of Tier1 Hospitals'
FROM merged_data_cleaned
WHERE `Hospital tier` = 'tier - 1'
GROUP BY `State ID`;
```

OUTPUT: