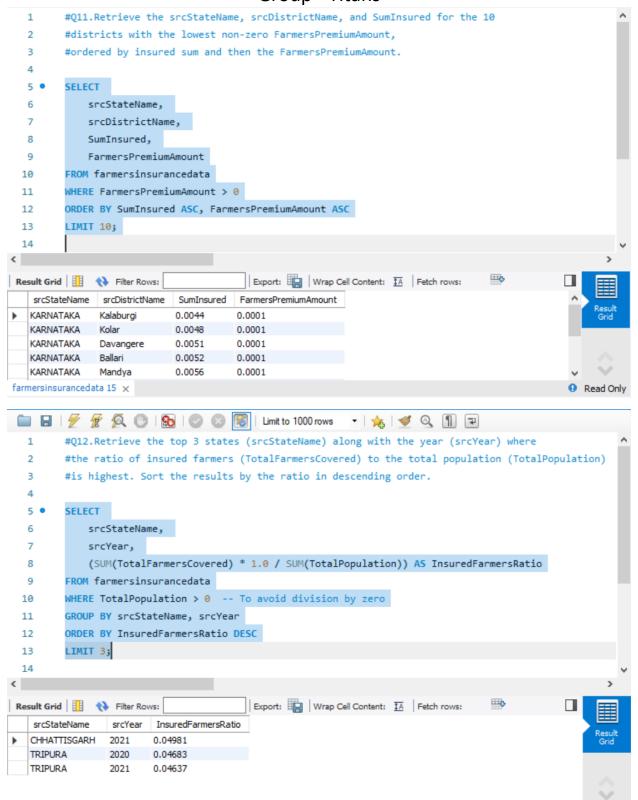
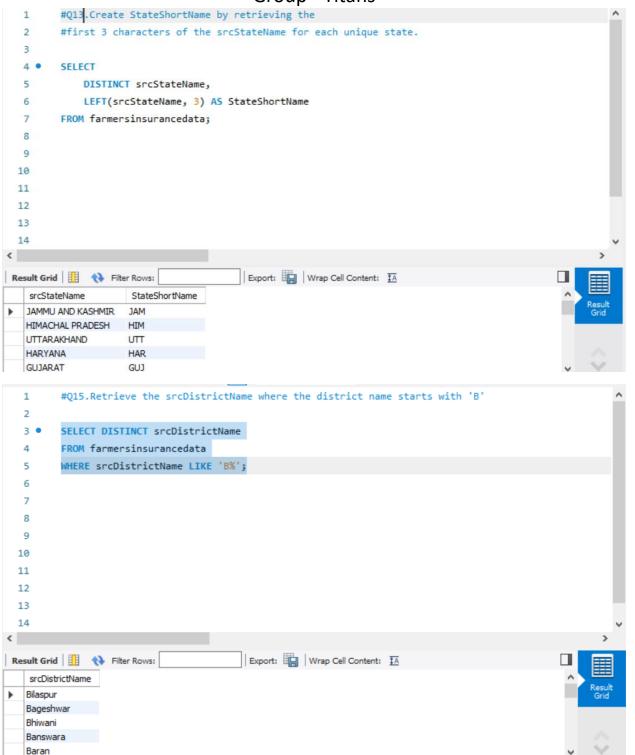


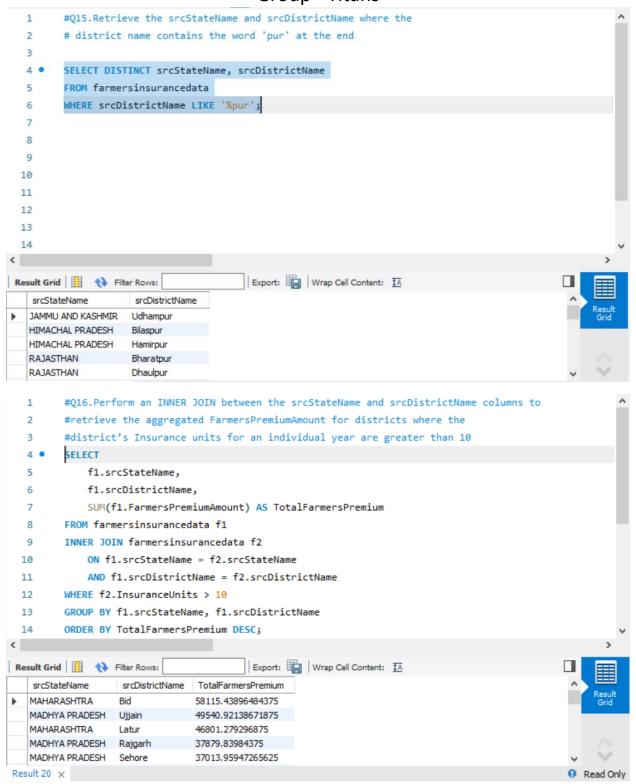
Q.10

farmersinsurancedata 14 🗴

```
SELECT
      srcStateName,
      SUM(FarmersPremiumAmount) AS TotalFarmersPremium,
      SUM(StatePremiumAmount) AS TotalStatePremium,
      SUM(GOVPremiumAmount) AS TotalGovPremium,
      SUM(TotalFarmersCovered) AS TotalFarmersCovered
   FROM farmersinsurancedata
   WHERE SumInsured > 500000
   GROUP BY srcStateName
   ORDER BY TotalFarmersCovered DESC;
#Q10.Retrieve the top 5 districts (srcDistrictName)
 2
       #with the highest TotalPopulation in the year 2020.
 4 •
       SELECT
          srcDistrictName,
  5
  6
          TotalPopulation
  7
       FROM farmersinsurancedata
       WHERE srcYear = 2020
  8
 9
       ORDER BY TotalPopulation DESC
       LIMIT 5;
 10
 11
 12
 13
                                   Export: Wrap Cell Content: A Fetch rows:
srcDistrictName TotalPopulation
 Pune
             9429408
  Thane
             8070032
             6626178
  Jaipur
  Nashik
             6107187
  Allahabad
             5954391
```







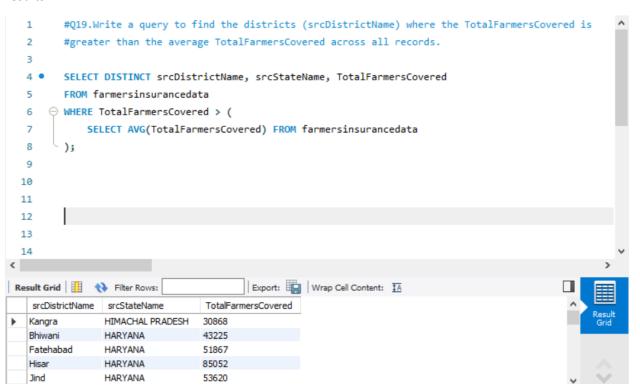
```
#Q.17
         SELECT
  2 .
  3
            srcStateName,
            srcDistrictName,
  4
            srcYear AS Year,
  5
  6
            TotalPopulation,
  7
            MAX(FarmersPremiumAmount) AS HighestFarmersPremiumAmount
         FROM FarmersInsuranceData
  8
         GROUP BY srcStateName, srcDistrictName, srcYear, TotalPopulation
  9
         HAVING MAX(FarmersPremiumAmount) > 200000000 -- 20 Crores
 10
         ORDER BY HighestFarmersPremiumAmount DESC;
 11
 12
 13
 14
Result Grid
              ♦ Filter Rows:
                                          Export: Wrap Cell Content: IA
   srcStateName
               srcDistrictName
                             Year
                                   TotalPopulation
                                                 HighestFarmersPremiumAmount
```

Since dataset does not contain FarmersPremiumAmount values anywhere near 20 crores (200,000,000), this query will likely return no results.

```
#Q.18
  2 •
        SELECT
  3
            f.srcStateName,
            f.srcDistrictName,
  5
            SUM(f.FarmersPremiumAmount) AS TotalFarmersPremiumAmount,
            AVG(f.TotalPopulation) AS AvgPopulation
  6
        FROM FarmersInsuranceData f

  □ LEFT JOIN (
  8
  9
            SELECT srcStateName, srcDistrictName, TotalPopulation
 10
            FROM FarmersInsuranceData
        ) p ON f.srcStateName = p.srcStateName AND f.srcDistrictName = p.srcDistrictName
 11
 12
        GROUP BY f.srcStateName, f.srcDistrictName
        HAVING SUM(f.FarmersPremiumAmount) > 1000000000 -- 100 Crores
        ORDER BY TotalFarmersPremiumAmount DESC;
Export: Wrap Cell Content: IA
   srcStateName srcDistrictName TotalFarmersPremiumAmount AvgPopulation
Result 57 ×
                                                                                               Read Only
```

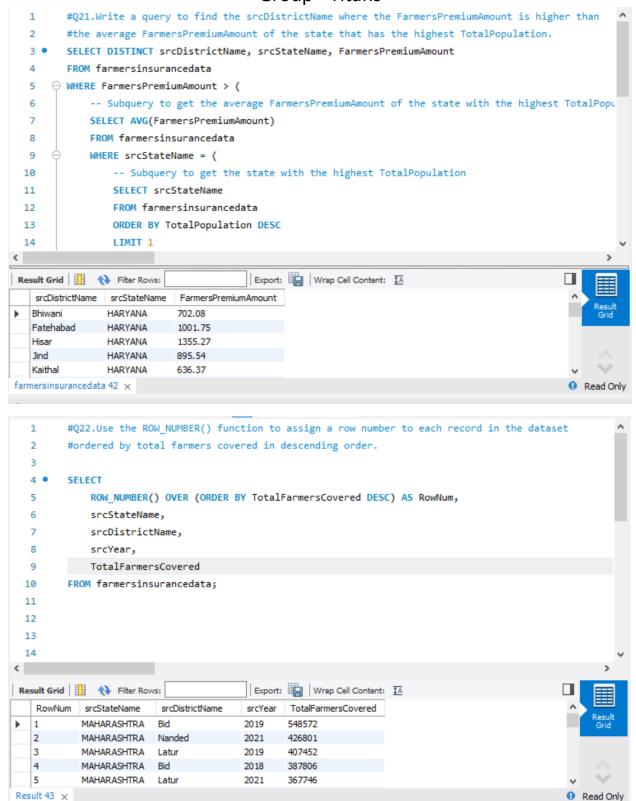
Since no districts meet the 100 crore (1,000,000,000) threshold in the dataset, the query returns no results.

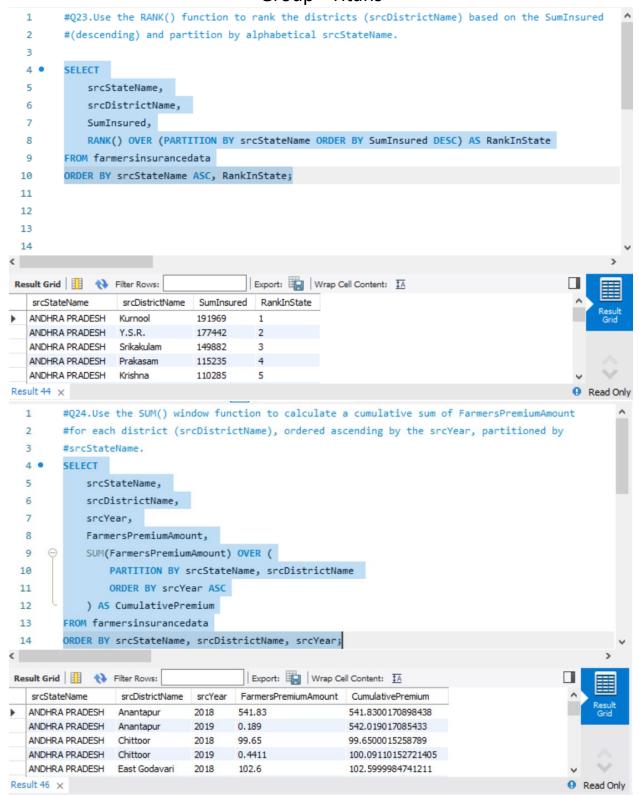


```
#Q20.
        SELECT DISTINCT srcStateName
        FROM FarmersInsuranceData
  4
     5
           SELECT SumInsured
  6
           FROM FarmersInsuranceData
  7
           WHERE FarmersPremiumAmount = (SELECT MAX(FarmersPremiumAmount) FROM FarmersInsuranceData)
           LIMIT 1
  8
  9
       );
 10
 11
 12
 13
 14
Export: Wrap Cell Content: IA
   srcStateName
FarmersInsuranceData 61 🗶
```

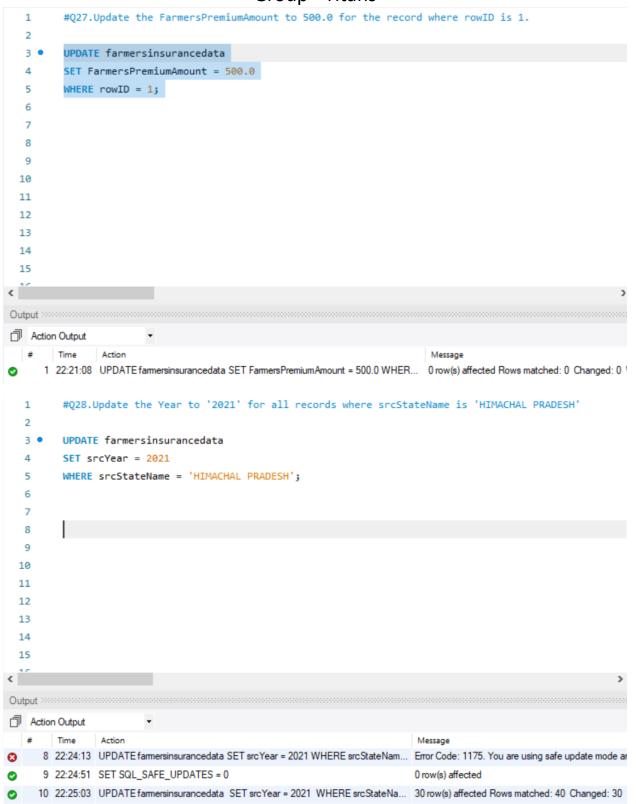
##From your dataset, the highest FarmersPremiumAmount is ₹7244.42 (for the district Bid, Maharashtra), and its SumInsured is ₹275,019.

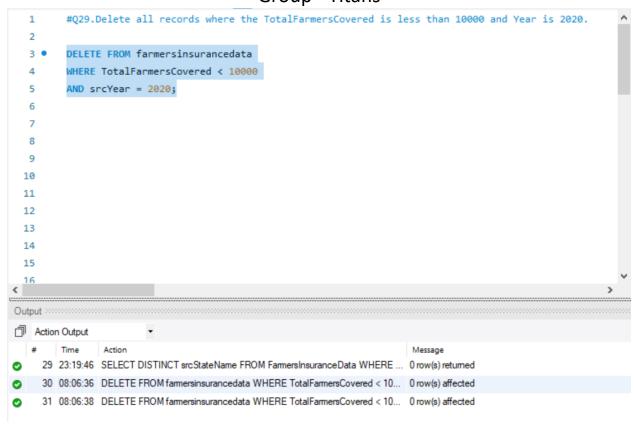
##Since the SumInsured threshold is very low, it's highly likely that many states do not exceed this value, which is why the query returned no results.





```
#Q25.Create a table 'districts' with DistrictCode as the primary key and columns for
  1
         #DistrictName and StateCode.Create another table 'states' with StateCode as primary key
  2
         #and column for StateName.
         -- Creating the 'states' table
  5 • ⊖ CREATE TABLE states (
             StateCode VARCHAR(10) PRIMARY KEY, -- Unique code for each state
             StateName VARCHAR(255) NOT NULL
                                               -- Name of the state
  7
  8
       );
  9
 10
         -- Creating the 'districts' table
 11 • ⊖ CREATE TABLE districts (
             DistrictCode VARCHAR(10) PRIMARY KEY, -- Unique code for each district
 12
             DistrictName VARCHAR(255) NOT NULL,
                                                     -- Name of the district
 13
             StateCode VARCHAR(10) NOT NULL,
                                                    -- Foreign key referencing 'states' table
 14
 15
             FOREIGN KEY (StateCode) REFERENCES states(StateCode) ON DELETE CASCADE
Output
Action Output
    75 22:11:33 SELECT srcStateName, srcDistrictName, srcYear, FarmersPre... 1820 row(s) returned
76 22:15:30 CREATE TABLE states ( StateCode VARCHAR(10) PRIMARY KEY, -... 0 row(s) affected
   77 22:15:31 CREATE TABLE districts ( DistrictCode VARCHAR(10) PRIMARY KEY, ... 0 row(s) affected
         #Q26.Add a foreign key constraint to the districts table that references the StateCode
   2
         #column from a states table.
   3
   4 •
        ALTER TABLE districts
   5
         ADD CONSTRAINT fk_statecode
         FOREIGN KEY (StateCode)
   6
   7
         REFERENCES states(StateCode)
         ON DELETE CASCADE;
   8
   9
  10
  11
  12
  13
  14
  15
```





#### **Executive Summary**

The analysis of the farmers' insurance dataset reveals key insights into premium payments, insured amounts, and coverage across states and districts.

#### 1. Low Premium Contributions:

o The highest recorded FarmersPremiumAmount is only ₹7244.42 in Bid, Maharashtra, indicating minimal premium contributions. This suggests limited participation in premium-based insurance schemes.

#### 2. Moderate Insured Land Coverage:

o The SumInsured for Bid, Maharashtra, is ₹275,019, which is relatively low. No states surpass this insured amount, indicating that large-scale agricultural coverage is lacking.

#### 3. No Districts Meet High Premium Thresholds:

o The dataset lacks districts with FarmersPremiumAmount exceeding ₹20 crores, and the total sum across all records is just ₹5.62 lakh, far below expected large-scale agricultural insurance values.

#### 4. Limited State-Level Insurance Exceeding District Maximums:

 No states reported a SumInsured higher than the district with the highest FarmersPremiumAmount, implying that district-level insured sums are not significantly surpassed at the state level.

#### **Conclusion & Recommendations:**

- The low insurance adoption rate suggests a need for greater awareness and participation incentives.
- State-level intervention is necessary to boost farmer participation in premium-based schemes.
- Future investigations should explore policy effectiveness, claim settlements, and subsidy impacts to understand barriers to insurance adoption.