

# **CENSUS OF INDIA 2011 ANALYSIS**

**DATA ANALYSIS CASE  
STUDY**

**By: SHUBHAM MODI**



# INTRODUCTION

The Census of India is a comprehensive national survey conducted every ten years by the Registrar General and Census Commissioner of India, under the Ministry of Home Affairs. Its primary objective is to gather and analyze demographic, social, and economic data of every resident in the country.

The census provides crucial insights into population size, distribution, literacy rates, and socio-economic conditions across various states and union territories. It plays a pivotal role in shaping government policies, planning developmental initiatives, and allocating resources effectively.

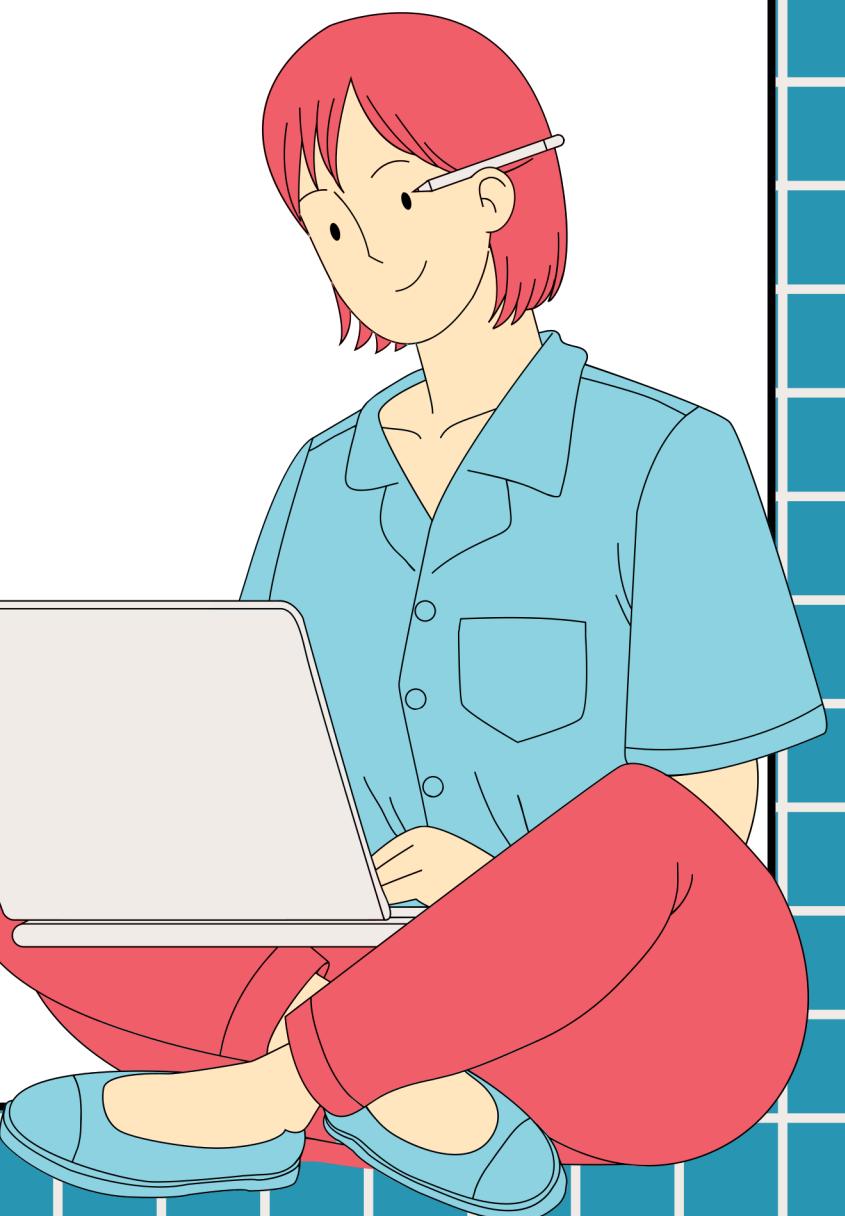


# INTRODUCTION

In this analysis, we delve into the Census of India dataset using SQL (Structured Query Language) to extract meaningful insights. SQL offers a powerful toolset to query, manipulate, and analyze structured data, making it ideal for exploring large-scale datasets such as the Census. By leveraging SQL queries, we aim to uncover patterns, correlations, and trends that shed light on India's demographic landscape.

Key areas of exploration include population distribution across states and regions, demographic characteristics such as age and gender distribution, literacy rates, urbanization trends, and socio-economic indicators such as employment patterns and household characteristics.

Through this analysis, we seek to not only summarize the data but also to derive actionable insights that can inform policy-making, urban planning, resource allocation, and social interventions aimed at addressing the diverse needs of India's population.



# DATA OVERVIEW

We have access to two distinct datasets focusing on Indian district demographics.

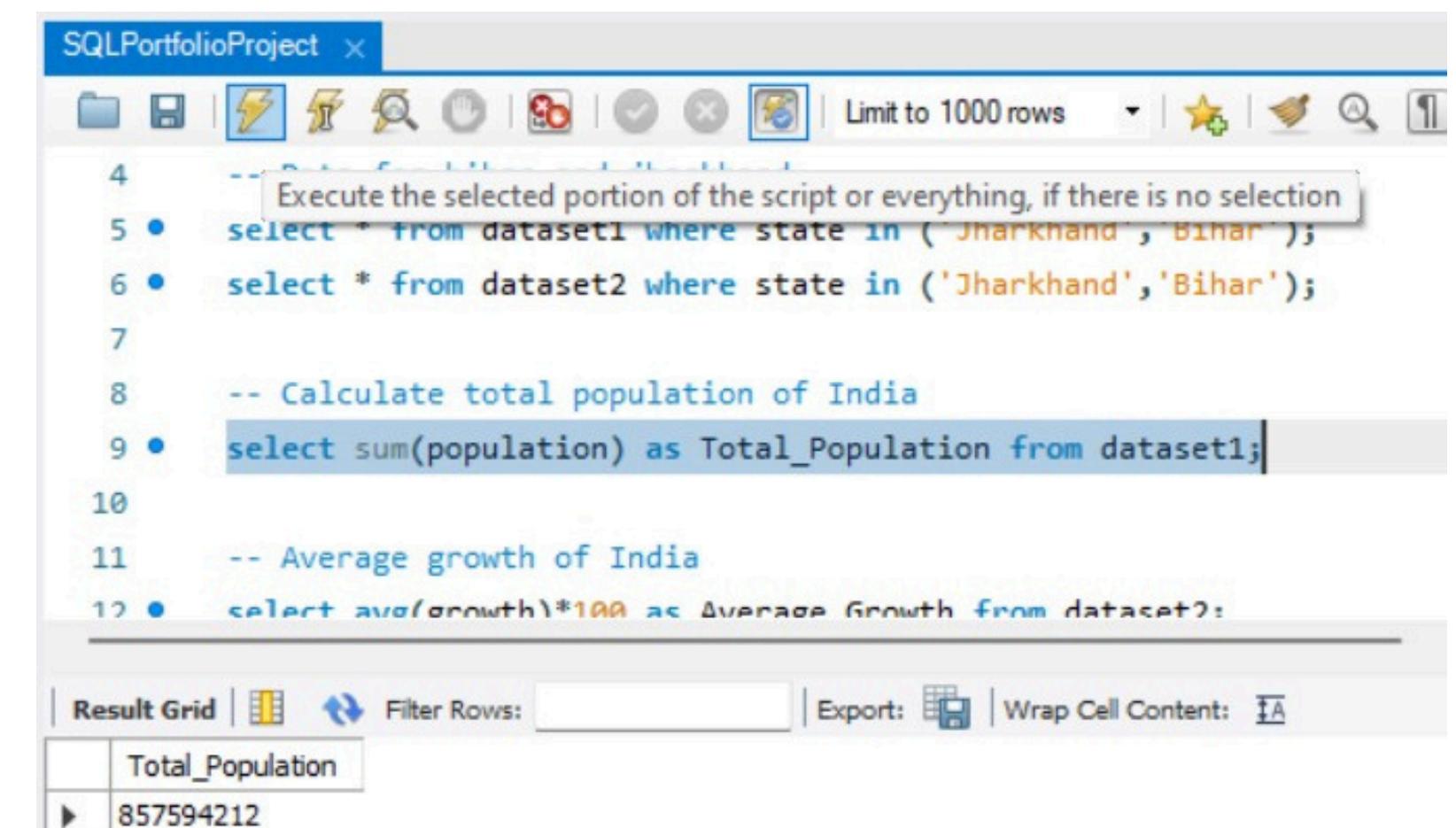
The first dataset encompasses Indian district names, their respective states, and population figures. This dataset serves as a foundational resource, providing essential information on population distribution across India's districts. It allows us to analyze regional population densities and demographic shifts within states over time.

The second dataset complements the first by including additional vital statistics such as population growth rates, sex ratios, and literacy rates for each district. These metrics offer a more nuanced understanding of socio-economic dynamics within districts, highlighting trends in population change, gender demographics, and educational attainment levels. Together, these datasets form a comprehensive basis for exploring and understanding the socio-demographic landscape of India at the district level, facilitating informed decision-making and policy formulation.

# QUERY 1: TOTAL POPULATION OF INDIA

Here, we are trying to find the total population of India according to the census of 2011 in the given dataset. We have used the sum function to find the total population of India.

The sum function gives us the sum of all entries of a particular column containing numeric data. In our case the 'Population' column was used to find the total population of India



```
SQLPortfolioProject x
4   -- Execute the selected portion of the script or everything, if there is no selection
5 • select * from dataset1 where state in ('Jharkhand', 'Bihar');
6 • select * from dataset2 where state in ('Jharkhand','Bihar');
7
8 -- Calculate total population of India
9 • select sum(population) as Total_Population from dataset1;
10
11 -- Average growth of India
12 • select avg(growth)*100 as Average_Growth from dataset2;
```

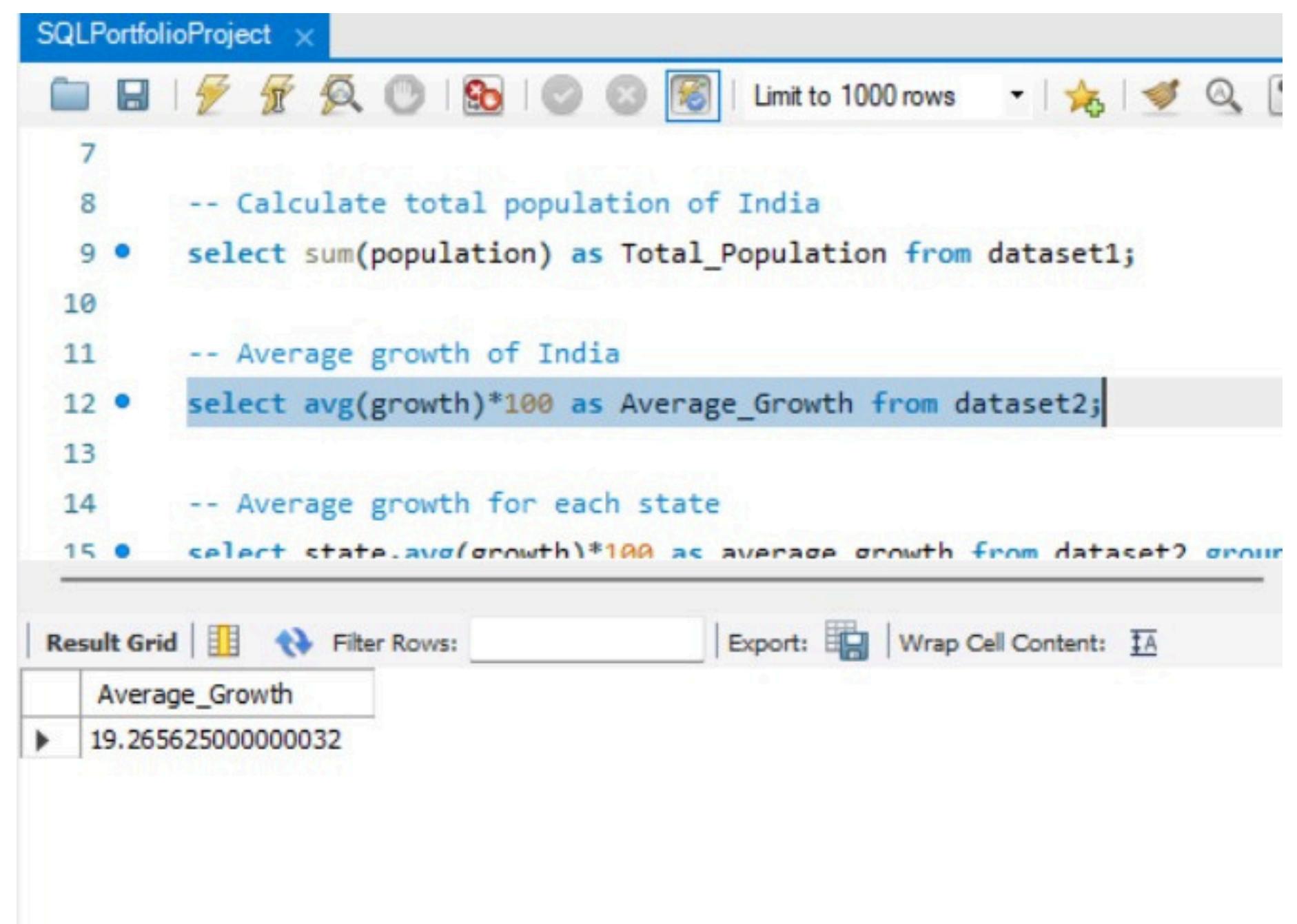
Result Grid | Filter Rows: \_\_\_\_\_ | Export: | Wrap Cell Content:

Total_Population
857594212

## QUERY 2: AVERAGE GROWTH RATE OF INDIA

Here, we are trying to find the average population growth rate of India. We have used the avg function to find the average population growth rate.

The avg function gives us the average of all entries of a particular column containing numeric data. In our case the 'Growth' column was used to find the average population growth rate of India



The screenshot shows a SQL query editor window titled "SQLPortfolioProject". The query itself is:

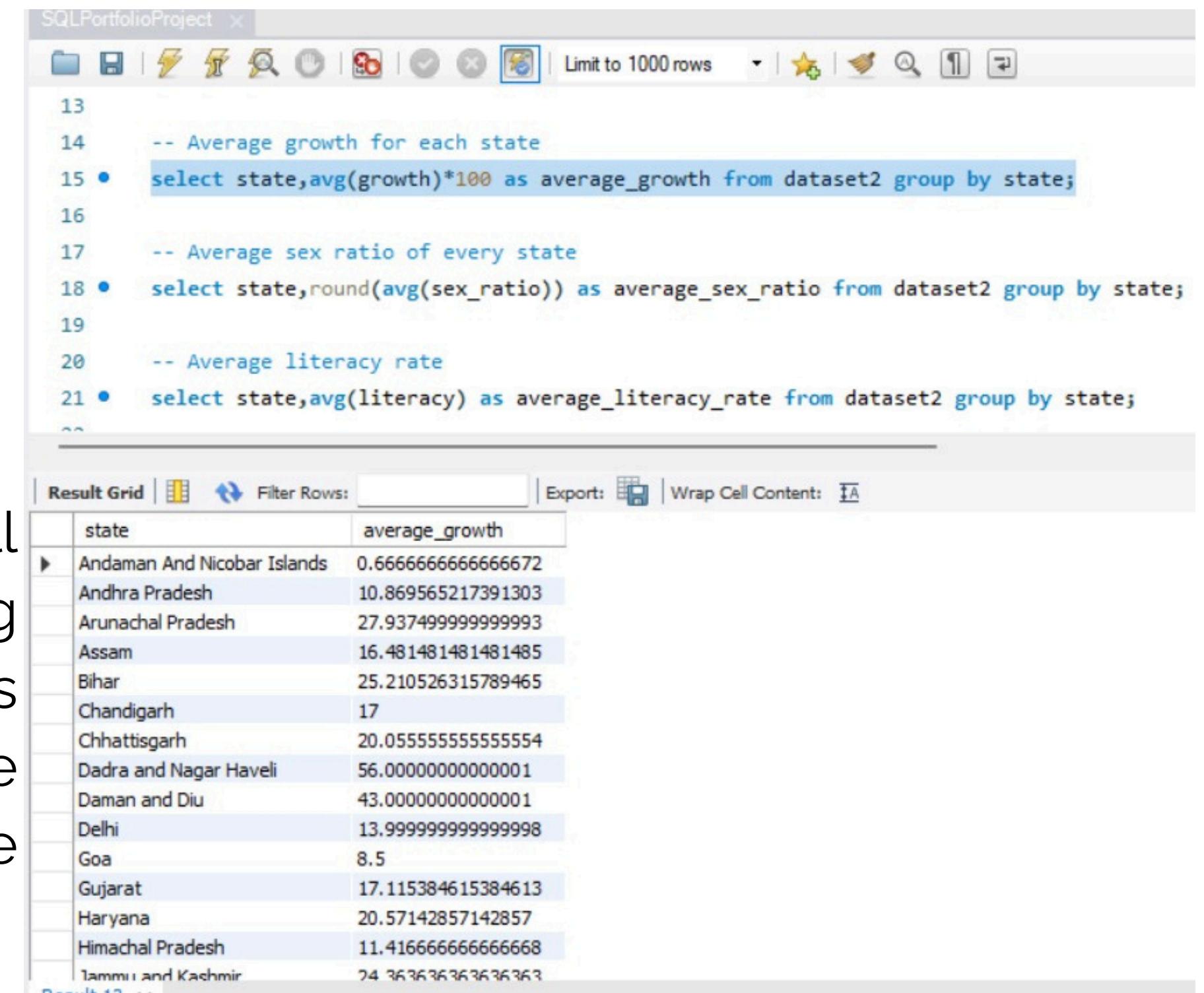
```
7  
8    -- Calculate total population of India  
9 •  select sum(population) as Total_Population from dataset1;  
10  
11   -- Average growth of India  
12 •  select avg(growth)*100 as Average_Growth from dataset2;  
13  
14   -- Average growth for each state  
15 •  select state.avg(growth)*100 as average_growth from dataset2 group by state;
```

The result grid below the query shows one row with the value "19.265625000000032" under the column "Average\_Growth".

# QUERY 3: AVERAGE GROWTH RATE OF EACH STATE OF INDIA

Here, we are trying to find the average population growth rate of each state of India. We have used the avg function and the group by statement to find the average population growth rate of each state.

The avg function gives us the average of all entries of a particular column containing numeric data and group by statement groups them according to every state. In our case the 'Growth' column was used to find the average population growth rate of each state



The screenshot shows a SQL development environment with the following details:

**SQL Portfolio Project** window title.

Toolbar icons include: folder, table, lightning bolt, magnifying glass, search, refresh, and others.

Code pane (Line numbers 13-21):

```
13  
14    -- Average growth for each state  
15 •  select state,avg(growth)*100 as average_growth from dataset2 group by state;  
16  
17    -- Average sex ratio of every state  
18 •  select state,round(avg(sex_ratio)) as average_sex_ratio from dataset2 group by state;  
19  
20    -- Average literacy rate  
21 •  select state,avg(literacy) as average_literacy_rate from dataset2 group by state;  
--
```

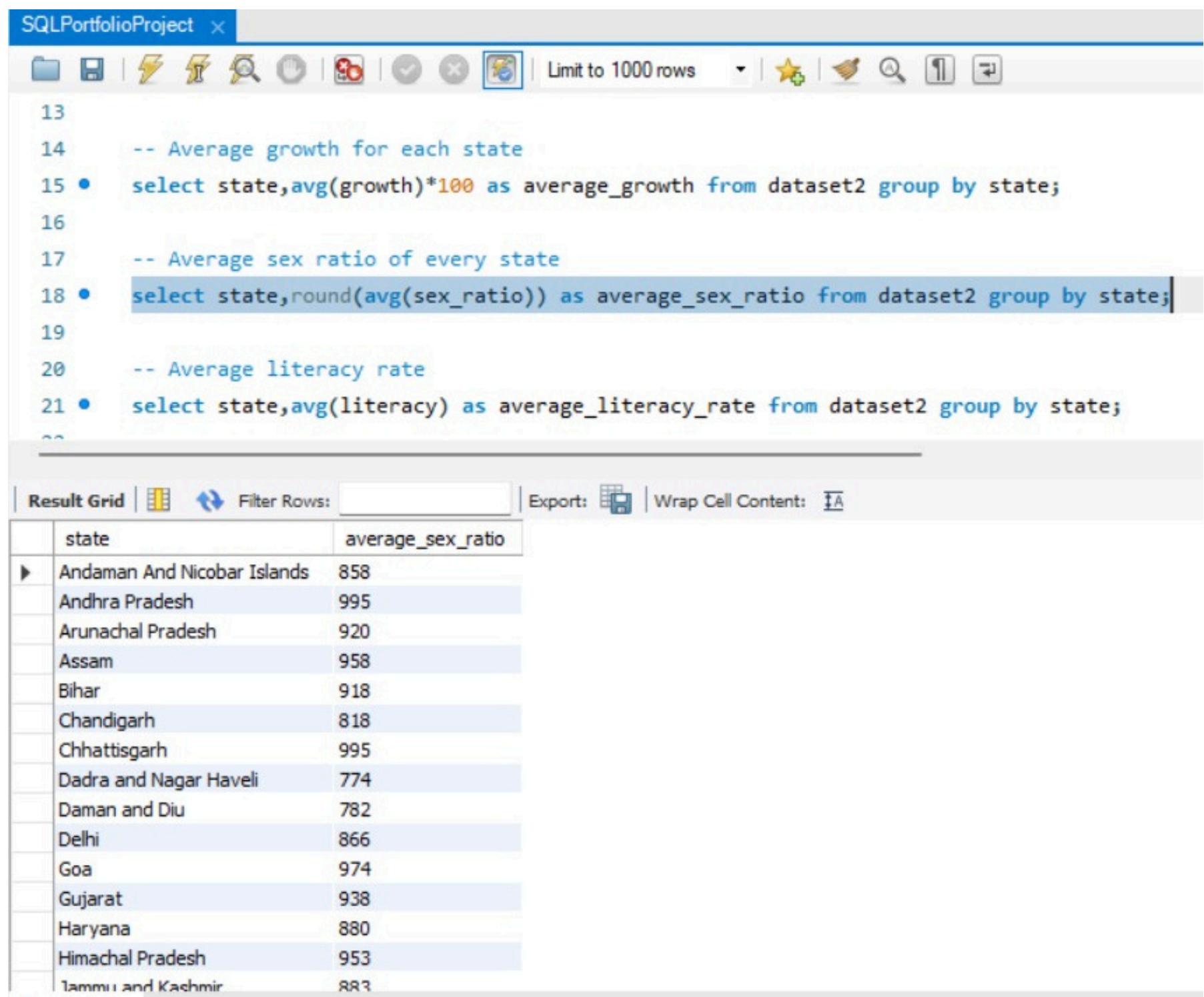
Result Grid pane:

state	average_growth
Andaman And Nicobar Islands	0.666666666666672
Andhra Pradesh	10.869565217391303
Arunachal Pradesh	27.937499999999993
Assam	16.481481481485
Bihar	25.210526315789465
Chandigarh	17
Chhattisgarh	20.055555555555554
Dadra and Nagar Haveli	56.00000000000001
Daman and Diu	43.00000000000001
Delhi	13.99999999999998
Goa	8.5
Gujarat	17.115384615384613
Haryana	20.57142857142857
Himachal Pradesh	11.416666666666668
Jammu and Kashmir	24.363636363636363

# QUERY 4: AVERAGE SEX RATIO OF EACH STATE OF INDIA

Here, we are trying to find the average sex ratio of each state of India. We have used the avg function and group by statement to find the sex ratio of each state.

The avg function gives us the average of all entries of a particular column containing numeric data and group by statement groups them according to the states. In our case the 'Sex\_Ratio' column was used to find the sex ratio rate of every state



The screenshot shows a SQL query editor window titled "SQLPortfolioProject". The code is as follows:

```
13  
14    -- Average growth for each state  
15 •  select state,avg(growth)*100 as average_growth from dataset2 group by state;  
16  
17    -- Average sex ratio of every state  
18 •  select state,round(avg(sex_ratio)) as average_sex_ratio from dataset2 group by state;  
19  
20    -- Average literacy rate  
21 •  select state,avg(literacy) as average_literacy_rate from dataset2 group by state;  
~~
```

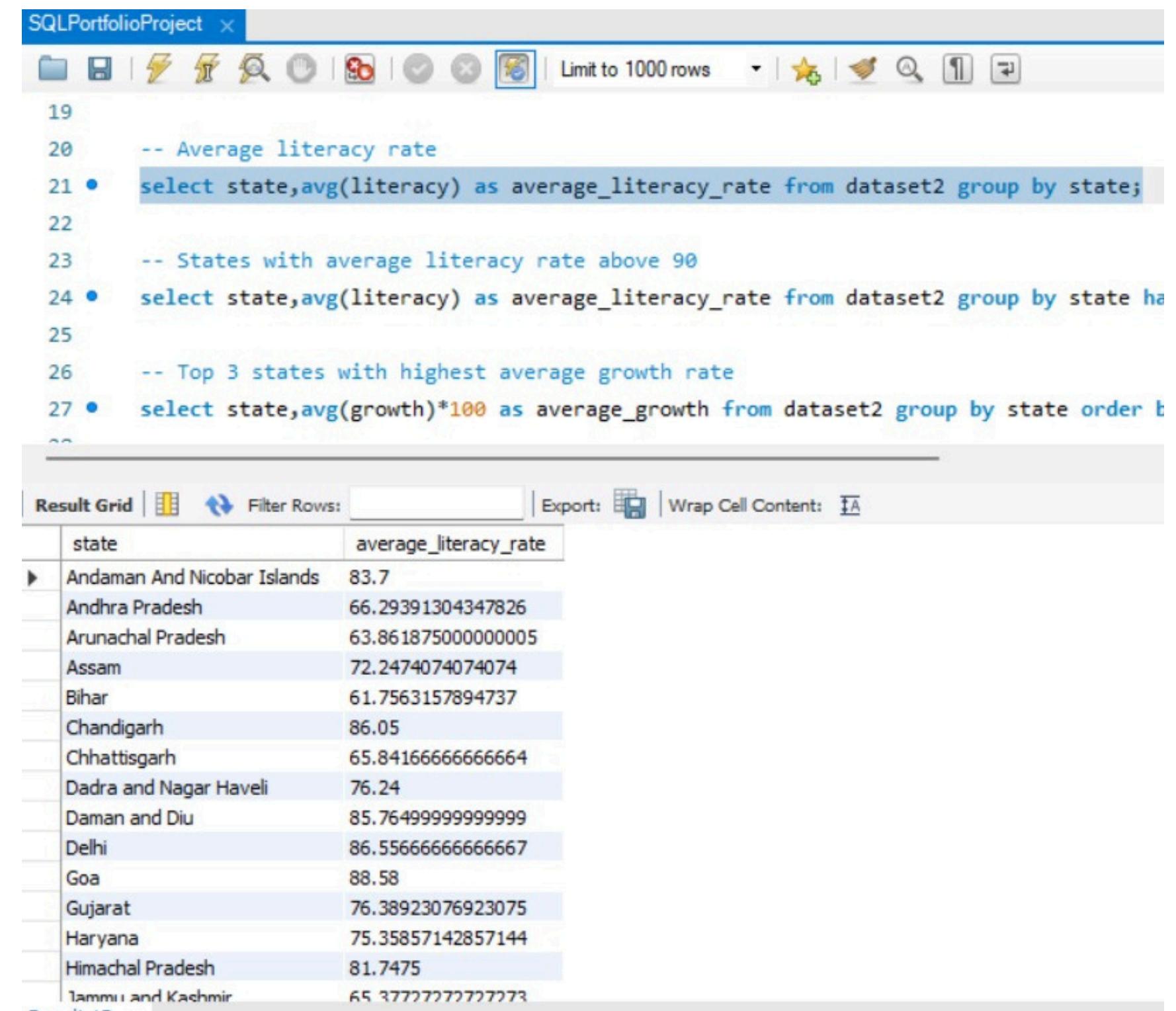
The result grid displays the following data:

state	average_sex_ratio
Andaman And Nicobar Islands	858
Andhra Pradesh	995
Arunachal Pradesh	920
Assam	958
Bihar	918
Chandigarh	818
Chhattisgarh	995
Dadra and Nagar Haveli	774
Daman and Diu	782
Delhi	866
Goa	974
Gujarat	938
Haryana	880
Himachal Pradesh	953
Jammu and Kashmir	993

# QUERY 5: AVERAGE LITERACY RATE OF EACH STATE OF INDIA

Here, we are trying to find the average literacy rate of each state of India. We have used the avg function and group by statement to find the literacy rate of each state.

The avg function gives us the average of all entries of a particular column containing numeric data and group by statement groups them according to the states. In our case the 'Literacy' column was used to find the literacy rate of every state.



The screenshot shows a SQL development environment with the following details:

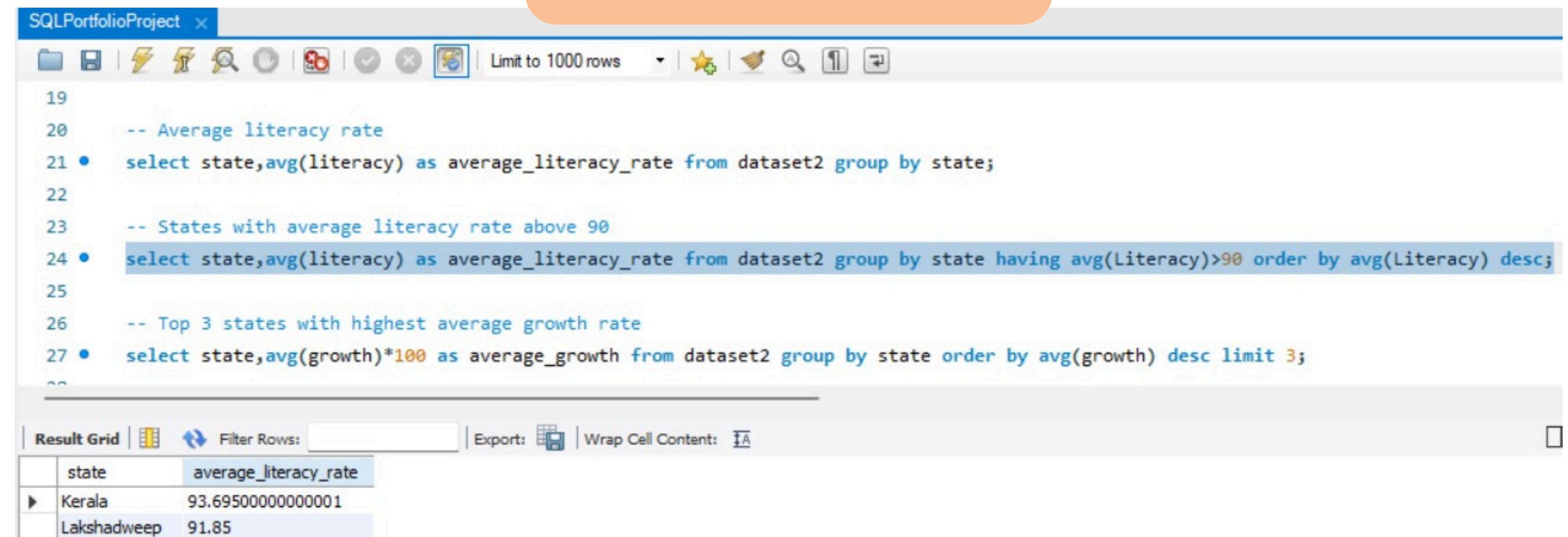
- SQLPortfolioProject** is the current project.
- The code editor contains the following SQL queries:

```
19
20    -- Average literacy rate
21 •     select state,avg(literacy) as average_literacy_rate from dataset2 group by state;
22
23    -- States with average literacy rate above 90
24 •     select state,avg(literacy) as average_literacy_rate from dataset2 group by state ha
25
26    -- Top 3 states with highest average growth rate
27 •     select state,avg(growth)*100 as average_growth from dataset2 group by state order b
```

- The result grid displays the following data:

state	average_literacy_rate
Andaman And Nicobar Islands	83.7
Andhra Pradesh	66.29391304347826
Arunachal Pradesh	63.861875000000005
Assam	72.2474074074074
Bihar	61.7563157894737
Chandigarh	86.05
Chhattisgarh	65.84166666666664
Dadra and Nagar Haveli	76.24
Daman and Diu	85.76499999999999
Delhi	86.55666666666667
Goa	88.58
Gujarat	76.38923076923075
Haryana	75.35857142857144
Himachal Pradesh	81.7475
Jammu and Kashmir	65.37777777777773

# QUERY 6: STATES HAVING LITERACY RATE ABOVE 90%



The screenshot shows a SQL editor window titled "SQLPortfolioProject". The code area contains the following SQL queries:

```
19
20    -- Average literacy rate
21 •  select state,avg(literacy) as average_literacy_rate from dataset2 group by state;
22
23    -- States with average literacy rate above 90
24 •  select state,avg(literacy) as average_literacy_rate from dataset2 group by state having avg(Literacy)>90 order by avg(Literacy) desc;
25
26    -- Top 3 states with highest average growth rate
27 •  select state,avg(growth)*100 as average_growth from dataset2 group by state order by avg(growth) desc limit 3;
```

The result grid shows the following data:

state	average_literacy_rate
Kerala	93.69500000000001
Lakshadweep	91.85

The `avg` function gives us the average of all entries of a particular column containing numeric data and having clause checks whether the average of the entries are above 90 or not. In our case the 'Literacy' column was used to find the states with literacy rate above 90%.

Here, we are trying to find the states with average literacy above 90%. We have used the `avg` function and `having` clause statement to find the states with literacy rate above 90%.

# QUERY 7: TOP 3 STATES WITH HIGHEST AVERAGE POPULATION GROWTH RATE

The avg function gives us the average of all entries of a particular column containing numeric data. In our case the 'Growth' column was used to find the average population growth rate of each state

```
25
26      -- Top 3 states with highest average growth rate
27 •   select state,avg(growth)*100 as average_growth from dataset2 group by state order by avg(growth) desc limit 3;
28
29      -- Top 3 states with lowest sex ratio
30 •   select state,round(avg(sex_ratio)) as average_sex_ratio from dataset2 group by state order by avg(sex_ratio) lim
31
32      -- States starting with 'A'
33 •   select distinct state from dataset1 where state like 'A%';
--
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows:

state	average_growth
Nagaland	82.27272727272728
Dadra and Nagar Haveli	56.00000000000001
Daman and Diu	43.00000000000001

Here, we are trying to find the top three states with highest average population growth rate. We have used the avg function and order by statement to order our output and limit statement to get only 3 outputs

# QUERY 8: TOP 3 STATES WITH LOWEST SEX RATIO

The avg function gives us the average of all entries of a particular column containing numeric data. In our case the 'Sex\_Ratio' column was used to find the average sex ratio of each state

```
28
29  -- Top 3 states with lowest sex ratio
30 • select state,round(avg(sex_ratio)) as average_sex_ratio from dataset2 group by state order by avg(sex_ratio) limit 3;
31
32  -- States starting with 'A'
33 • select distinct state from dataset1 where state like 'A%';
34
35  -- States starting with 'A' or 'B'
36 • select distinct state from dataset1 where state like 'A%' or state like 'B%';
--
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows:

state	average_sex_ratio
Dadra and Nagar Haveli	774
Daman and Diu	782
Chandigarh	818

Here, we are trying to find the top three states with lowest sex ratio. We have used the avg function and order by statement to order our output and limit statement to get only 3 outputs.

# QUERY 9: TOP 3 DISTRICTS WITH HIGHEST LITERACY RATE

Here, we are trying to find the top three districts with highest literacy rate. We have used the `dense_rank` function to find the rank of each district in a state and where clause to find the districts with rank 1,2 and 3.

The `dense_rank` function gives us the rank of all entries of a particular column according to a condition. In our case the 'Literacy' column was used to find the literacy rate of each district.

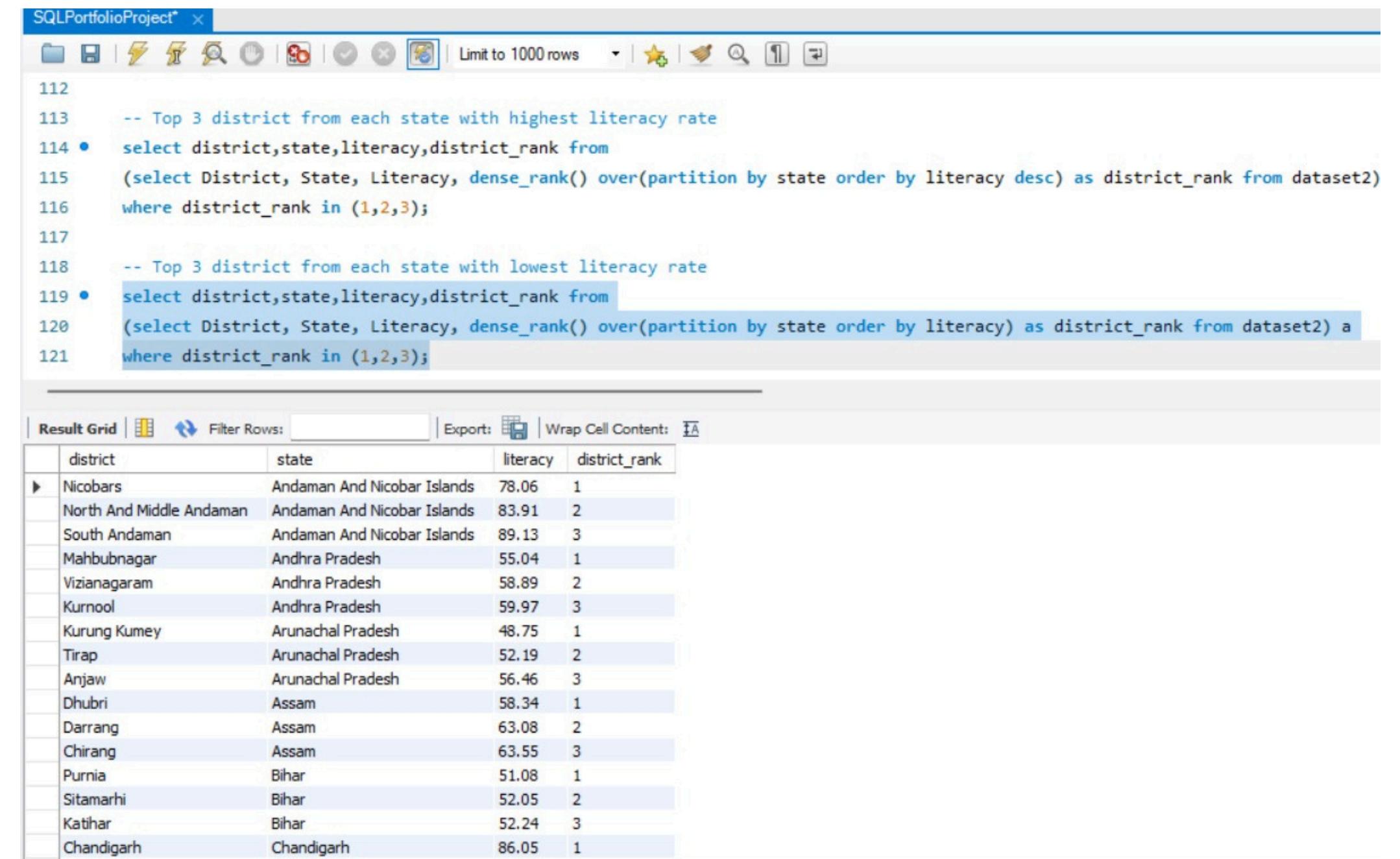
```
112
113    -- Top 3 district from each state with highest literacy rate
114 • select district,state,literacy,district_rank from
115      (select District, State, Literacy, dense_rank() over(partition by state order by literacy desc) as district_rank from dataset2) a
116    where district_rank in (1,2,3);
117
118    -- Top 3 district from each state with lowest literacy rate
119 • select district,state,literacy,district_rank from
120      (select District, State, Literacy, dense_rank() over(partition by state order by literacy) as district_rank from dataset2) a
121    where district_rank in (1,2,3);
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:	
		district	state	literacy	district_rank
▶	South Andaman	Andaman And Nicobar Islands	89.13	1	
	North And Middle Andaman	Andaman And Nicobar Islands	83.91	2	
	Nicobars	Andaman And Nicobar Islands	78.06	3	
	Hyderabad	Andhra Pradesh	83.25	1	
	Rangareddy	Andhra Pradesh	75.87	2	
	West Godavari	Andhra Pradesh	74.63	3	
	Papumpare	Arunachal Pradesh	79.95	1	
	Lower Subansiri	Arunachal Pradesh	74.35	2	
	East Siang	Arunachal Pradesh	72.54	3	
	Kamrup Metropolitan	Assam	88.71	1	
	Jorhat	Assam	82.15	2	
	Sivasagar	Assam	80.41	3	
	Rohtas	Bihar	73.37	1	
	Patna	Bihar	70.68	2	
	Bhojpur	Bihar	70.47	3	
	Chandigarh	Chandigarh	86.05	1	

# QUERY 10: TOP 3 DISTRICTS WITH LOWEST LITERACY RATE

Here, we are trying to find the top three districts with lowest literacy rate. We have used the dense\_rank function to find the rank of each district in a state and where clause to find the districts with rank 1,2 and 3.

The dense\_rank function gives us the rank of all entries of a particular column according to a condition. In our case the 'Literacy' column was used to find the literacy rate of each district.



The screenshot shows a SQL query in the SQL Server Management Studio (SSMS) interface. The query is designed to find the top 3 districts with the lowest literacy rates from each state. It uses the dense\_rank() window function to rank districts by literacy rate within each state, and then filters for the top 3 ranks (1, 2, and 3).

```
SQLPortfolioProject* x
112
113    -- Top 3 district from each state with highest literacy rate
114 •  select district,state,literacy,district_rank from
115     (select District, State, Literacy, dense_rank() over(partition by state order by literacy desc) as district_rank from dataset2)
116     where district_rank in (1,2,3);
117
118    -- Top 3 district from each state with lowest literacy rate
119 •  select district,state,literacy,district_rank from
120     (select District, State, Literacy, dense_rank() over(partition by state order by literacy) as district_rank from dataset2) a
121     where district_rank in (1,2,3);
```

The result grid displays the following data:

district	state	literacy	district_rank
Nicobars	Andaman And Nicobar Islands	78.06	1
North And Middle Andaman	Andaman And Nicobar Islands	83.91	2
South Andaman	Andaman And Nicobar Islands	89.13	3
Mahbubnagar	Andhra Pradesh	55.04	1
Vizianagaram	Andhra Pradesh	58.89	2
Kurnool	Andhra Pradesh	59.97	3
Kurung Kumey	Arunachal Pradesh	48.75	1
Tirap	Arunachal Pradesh	52.19	2
Anjaw	Arunachal Pradesh	56.46	3
Dhubri	Assam	58.34	1
Darrang	Assam	63.08	2
Chirang	Assam	63.55	3
Purnia	Bihar	51.08	1
Sitamarhi	Bihar	52.05	2
Katihar	Bihar	52.24	3
Chandigarh	Chandigarh	86.05	1

# QUERY 11: STATES STARTING WITH 'A'

```
32    -- States starting with 'A'  
33 •  select distinct state from dataset1 where state like 'A%';  
34  
35    -- States starting with 'A' or 'B'  
36 •  select distinct state from dataset1 where state like 'A%' or state like 'B%';  
37  
38    -- Number of districts in each state  
39 •  select state,count(district) as number_of_districts from dataset2 group by state order by count(district) desc;  
--
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

state
Andhra Pradesh
Arunachal Pradesh
Assam
Andaman And Nicobar Islands

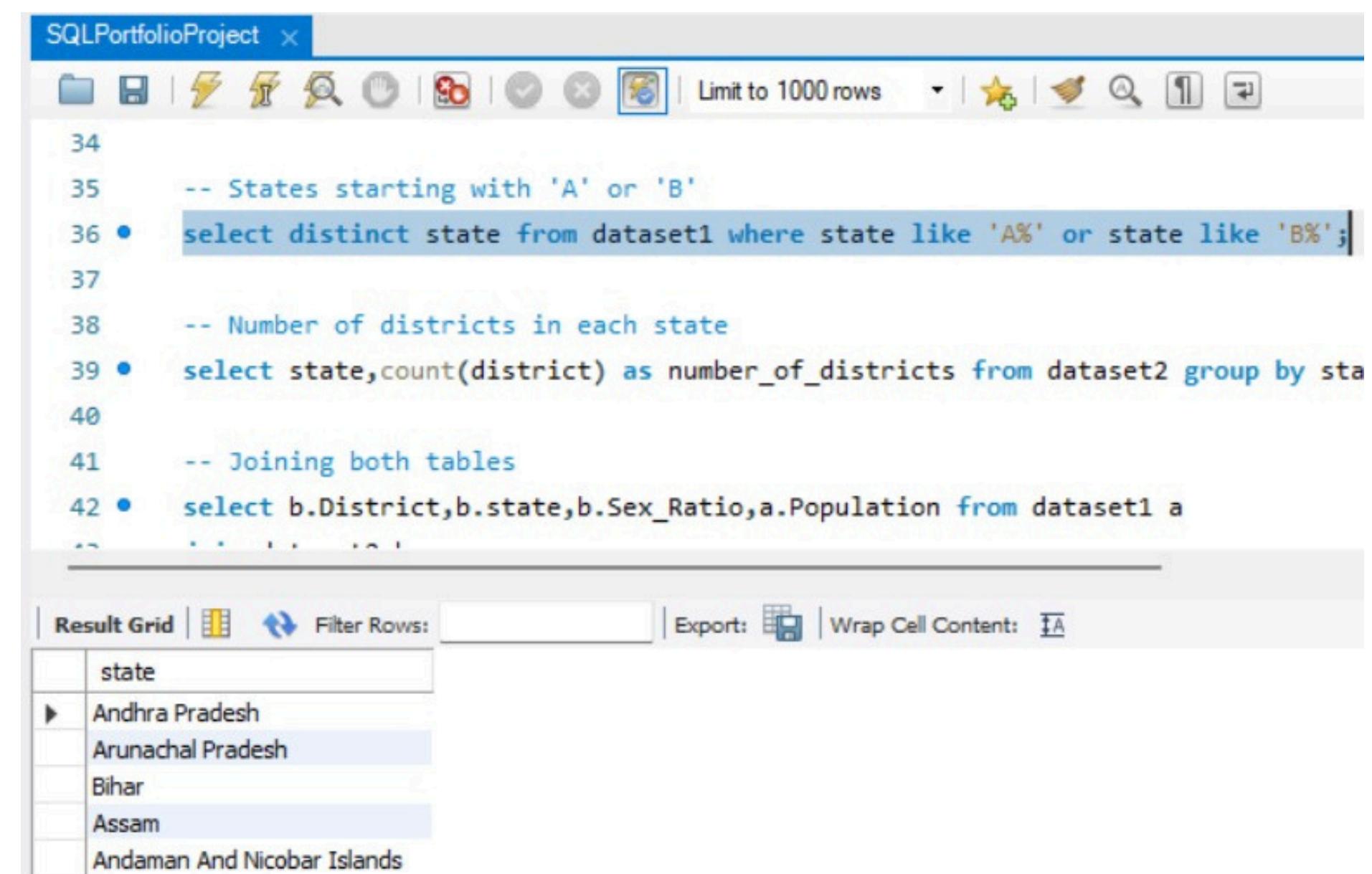
The like clause gives us the strings which are according to the condition given. In our case the 'State' column was used to find the names of the states starting with 'A'.

Here, we are trying to find the states which start with the letter 'A'. We have used the like clause here to find the states starting with 'A'.

## QUERY 12: STATES STARTING WITH 'A' OR 'B'

Here, we are trying to find the states which start with the letter 'A' or 'B'. We have used the like clause here to find the states starting with 'A' or 'B'.

The like clause gives us the strings which are according to the condition given. In our case the 'State' column was used to find the names of the states starting with 'A' or 'B'.



The screenshot shows a SQL development environment titled "SQLPortfolioProject". The query window contains the following code:

```
34
35  -- States starting with 'A' or 'B'
36 • select distinct state from dataset1 where state like 'A%' or state like 'B%';
37
38  -- Number of districts in each state
39 • select state,count(district) as number_of_districts from dataset2 group by sta
40
41  -- Joining both tables
42 • select b.District,b.state,b.Sex_Ratio,a.Population from dataset1 a
-- ...
```

The result grid displays the following data:

state
Andhra Pradesh
Arunachal Pradesh
Bihar
Assam
Andaman And Nicobar Islands

# QUERY 13: NUMBER OF DISTRICTS IN EACH STATE

Here, we are trying to find the number of districts in each state. We have used the count function and group by clause to find the number of districts in each state.

```
38      -- Number of districts in each state
39 •  select state,count(district) as number_of_districts from dataset2 group by state order by count(district) desc;
40
41      -- Joining both tables
42 •  select b.District,b.state,b.Sex_Ratio,a.Population from dataset1 a
43     join dataset2 b
44     on a.district=b.district;
45
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

state	number_of_districts
Uttar Pradesh	71
Madhya Pradesh	50
Bihar	38
Maharashtra	35
Rajasthan	33
Tamil Nadu	32
Orissa	30
Karnataka	30
Assam	27
Gujarat	26
Jharkhand	24
Andhra Pradesh	23
Jammu and Kas...	22
Haryana	21
	~

## QUERY 14: NUMBER OF MALES AND FEMALES IN EACH DISTRICT

Here, we are trying to find the population of males and females in each district. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out number of males and females for each district using 'Sex\_Ratio' and 'Population' columns from the table to find out number of males and females in each district.

```
46    -- Number of males and females in each district
47    -- population=males+females
48    -- population=males+(sex_ratio*males)
49    -- population=males(1+sex_ratio)
50    -- males=population/(1+sex_Ratio)
51
52 •   select t.district,t.state,round(t.population/(t.ratio+1)) males,round(t.population-t.population/(t.ratio+1)) females from
53   (select b.District,b.state,b.Sex_Ratio/1000 as ratio,a.Population from dataset1 a
54   join dataset2 b
55   on a.district=b.district) t;
```

District	state	males	females
Nicobars	Andaman And Nicobar Islands	20733	16109
East Godavari	Andhra Pradesh	2569440	2584856
Guntur	Andhra Pradesh	2440246	2447567
Krishna	Andhra Pradesh	2267770	2249628
Chittoor	Andhra Pradesh	2090167	2083897
Anantapur	Andhra Pradesh	2064314	2016834
Kurnool	Andhra Pradesh	2038965	2014498
Mahbubnagar	Andhra Pradesh	2050090	2002938
Hyderabad	Andhra Pradesh	2018077	1925246
Karimnagar	Andhra Pradesh	1880612	1895657
Nalgonda	Andhra Pradesh	1759359	1729450
Medak	Andhra Pradesh	1522735	1510553
Khammam	Andhra Pradesh	1391034	1406336
Adilabad	Andhra Pradesh	1369935	1371304
Nizamabad	Andhra Pradesh	1250654	1300681
Changlang	Arunachal Pradesh	76961	71265

# QUERY 15: NUMBER OF MALES AND FEMALES IN EACH STATE

Here, we are trying to find the population of males and females in each state. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out number of males and females for each district using 'Sex\_Ratio' and 'Population' columns from the table to find the number of males and females in each state.

```
58 -- Number of males and females in each state
59 • select d.state,sum(d.males) total_males,sum(d.females) total_females from
60 (select t.district,t.state,round(t.population/(t.ratio+1)) males,round(t.population-t.population/(t.ratio+1)) females from
61 (select b.District,b.state,b.Sex_Ratio/1000 as ratio,a.Population from dataset1 a
62 join dataset2 b
63 on a.district=b.district) t) d
64 group by d.state;
65
66 -- Total literacy rate for each district
67 • (select b.District,b.state,b.Literacy,a.Population from dataset1 a
```

state	total_males	total_females
Andaman And Nicobar Islands	20733	16109
Andhra Pradesh	26713398	26539445
Arunachal Pradesh	373207	357006
Assam	13265271	12705548
Bihar	37871662	34720352
Chandigarh	580556	474894
Chhattisgarh	7581606	7506801
Daman and Diu	150264	92983
Delhi	77939	64065
Goa	416713	401295
Gujarat	19034363	17715440
Haryana	10341523	9089008
Himachal Pradesh	4063362	4092232
Jammu and Kashmir	4181268	3705037
Jharkhand	12330226	11685092
Karnataka	25476595	24641950

# QUERY 16: TOTAL LITERATE POPULATION OF EACH DISTRICT

Here, we are trying to find the total literate population of each district. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out the total literate population for each district using 'Literacy' and 'Population' columns from the table to find the total literate population of each district.

```
73  -- Total literate population for each district
74  -- Total literate people=population*literacy_rate/100
75 • select c.district,c.state,c.population,round(c.lit_ratio*c.population) as Total_literate_population from
76  (select b.District,b.state,b.Literacy/100 as lit_ratio,a.Population from dataset1 a
77  join dataset2 b
78  on a.district=b.district) c;
79
80  -- Total literate population for each state
81 • select d.state,sum(d.population) as Total_population,sum(d.Total_literate_population) as Total_literate_po
82  (select c.district,c.state,c.population,round(c.lit_ratio*c.population) as Total_literate_population from
```

District	state	Population	Total_literate_population
Nicobars	Andaman And Nicobar Islands	36842	28759
East Godavari	Andhra Pradesh	5154296	3659035
Guntur	Andhra Pradesh	4887813	3294386
Krishna	Andhra Pradesh	4517398	3331129
Chittoor	Andhra Pradesh	4174064	2985708
Anantapur	Andhra Pradesh	4081148	2594386
Kurnool	Andhra Pradesh	4053463	2430862
Mahbubnagar	Andhra Pradesh	4053028	2230787
Hyderabad	Andhra Pradesh	3943323	3282816
Karimnagar	Andhra Pradesh	3776269	2422477
Nalgonda	Andhra Pradesh	3488809	2239815
Medak	Andhra Pradesh	3033288	1863045
Khammam	Andhra Pradesh	2797370	1812975
Adilabad	Andhra Pradesh	2741239	1672430
Nizamabad	Andhra Pradesh	2551335	1562693
Changlang	Arunachal Pradesh	148226	88639

# QUERY 17: TOTAL LITERATE POPULATION OF EACH STATE

Here, we are trying to find the total literate population of each state. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out the total literate population for each state using 'Literacy' and 'Population' columns from the table to find the total literate population of each state.

```
82  -- Total literate population for each state
83 • select d.state,sum(d.population) as Total_population,sum(d.Total_literate_population) as Total_literate_population from
84  (select c.district,c.state,c.population,round(c.lit_ratio*c.population) as Total_literate_population from
85  (select b.District,b.state,b.Literacy/100 as lit_ratio,a.Population from dataset1 a
86  join dataset2 b
87  on a.district=b.district) c) d
88 group by d.state;
89
90 -- Population in previous census for each district
91 -- New population=(1+growth)*old population
```

state	Total_population	Total_literate_population
Andaman And Nicobar Islands	36842	28759
Andhra Pradesh	53252843	35382544
Arunachal Pradesh	730213	468283
Assam	25970819	18794125
Bihar	72592014	45433541
Chandigarh	1055450	908215
Chhattisgarh	15088407	9942410
Daman and Diu	243247	211827
Delhi	142004	125446
Goa	818008	732690
Gujarat	36749803	28653958
Haryana	19430531	14557464
Himachal Pradesh	8155594	6848977
Jammu and Kashmir	7886305	5282300
Jharkhand	24015318	15482691
Karnataka	50118545	37992980

# QUERY 18: TOTAL POPULATION IN PREVIOUS CENSUS FOR EVERY DISTRICT

Here, we are trying to find the total population of each district during the last census. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out the total population for each district during the last census using 'Growth' and 'Population' columns from the table to find the total population of each district during the last census.

```
86      -- Population in previous census for each district
87      -- New population=(1+growth)*old population
88      -- old population=new population/(1+growth)
89 •   select c.district,c.state,c.growth,c.population,round(c.population/(c.growth+1)) as old_population from
90      (select b.District,b.state,b.Growth,a.Population from dataset1 a
91      join dataset2 b
92      on a.district=b.district) c;
93
94      -- Population in previous census for each state
95 •   select d.state,sum(d.population) as Current_Population,sum(d.old_population) as Old_Population from
```

District	state	Growth	Population	old_population
Nicobars	Andaman And Nicobar Islands	-0.12	36842	41866
East Godavari	Andhra Pradesh	0.05	5154296	4908853
Guntur	Andhra Pradesh	0.09	4887813	4484232
Krishna	Andhra Pradesh	0.08	4517398	4182776
Chittoor	Andhra Pradesh	0.11	4174064	3760418
Anantapur	Andhra Pradesh	0.12	4081148	3643882
Kurnool	Andhra Pradesh	0.15	4053463	3524750
Mahbubnagar	Andhra Pradesh	0.15	4053028	3524372
Hyderabad	Andhra Pradesh	0.03	3943323	3828469
Karimnagar	Andhra Pradesh	0.08	3776269	3496545
Nalgonda	Andhra Pradesh	0.07	3488809	3260569
Medak	Andhra Pradesh	0.14	3033288	2660779
Khammam	Andhra Pradesh	0.08	2797370	2590157
Adilabad	Andhra Pradesh	0.1	2741239	2492035
Nizamabad	Andhra Pradesh	0.09	2551335	2340674
Changlang	Arunachal Pradesh	0.18	148226	125615

# QUERY 19: TOTAL POPULATION IN PREVIOUS CENSUS FOR EVERY STATE

Here, we are trying to find the total population of each state during the last census. We have used concept of sub query to carry out this query.

We have used mathematical operations to find out the total population for each state during the last census using 'Growth' and 'Population' columns from the table to find the total population of each state during the last census.

```
94  -- Population in previous census for each state
95 • select d.state,sum(d.population) as Current_Population,sum(d.old_population) as Old_Population from
96  (select c.district,c.state,c.growth,c.population,round(c.population/(c.growth+1)) as old_population from
97  (select b.District,b.state,b.Growth,a.Population from dataset1 a
98  join dataset2 b
99  on a.district=b.district) c) d
100 group by d.state;
101
102 -- Total population of India Current vs Old
103 • select sum(e.current_population) as Total_Current_Population,sum(e.old_population) as Total_Old_Population
```

state	Current_Population	Old_Population
Andaman And Nicobar Islands	36842	41866
Andhra Pradesh	53252843	48698511
Arunachal Pradesh	730213	568761
Assam	25970819	22041342
Bihar	72592014	58003993
Chandigarh	1055450	902094
Chhattisgarh	15088407	12553035
Daman and Diu	243247	157924
Delhi	142004	179752
Goa	818008	757415
Gujarat	36749803	31474166
Haryana	19430531	15995301
Himachal Pradesh	8155594	7292408
Jammu and Kashmir	7886305	6405449
Jharkhand	24015318	19490595
Karnataka	50118545	42738151

# QUERY 20: TOTAL POPULATION IN PREVIOUS CENSUS OF INDIA

```
104    -- Total population of India Current vs Old
105 •  select sum(e.current_population) as Total_Current_Population,sum(e.old_population) as Total_Old_Population from
106   (select d.state,sum(d.population) as Current_Population,sum(d.old_population) as Old_Population from
107   (select c.district,c.state,c.growth,c.population,round(c.population/(c.growth+1)) as old_population from
108   (select b.District,b.state,b.Growth,a.Population from dataset1 a
109   join dataset2 b
110   on a.district=b.district) c) d
111   group by d.state) e;
112
113    -- Top 3 district from each state with highest literacy rate
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	Total_Current_Population	Total_Old_Population		
▶	842717165	717078939		

We have used mathematical operations to find out the total population of India during the last census using 'Growth' and 'Population' columns from the table to find the total population of India during the last census.

Here, we are trying to find the total population of India during the last census. We have used concept of sub query to carry out this query.

# CONCLUSION

In conclusion, the analysis of these two datasets—comprising district-level demographic data from the Census of India—has provided valuable insights into the socio-economic landscape of the country. By leveraging SQL queries to explore population distribution, growth rates, sex ratios, and literacy rates across Indian districts, we have uncovered significant patterns and trends. Key findings include identifying districts with rapid population growth, disparities in literacy rates between states, and variations in gender demographics across regions.

The insights derived from this analysis hold crucial implications for various stakeholders. Policymakers can use this information to allocate resources effectively, prioritize development initiatives, and formulate targeted interventions aimed at improving education, healthcare, and infrastructure in underserved districts. Urban planners can utilize these insights to forecast population trends and plan sustainable urban development strategies. Researchers and analysts can further delve into specific correlations between socio-economic factors, offering deeper insights into the drivers of demographic change in India.