**Analyzing Education & Census Data**

**Project Tasks**

**1) Explore Tables :**

1. Start by exploring each table separately to understand the data.

**SELECT \***

**FROM `stoked-producer-388612.12.census\_data`**

**LIMIT 20;**

**SELECT \***

**FROM `stoked-producer-388612.12.public\_hs`**

**LIMIT 100;**

**2) Public High Schools Count :**

1. Determine the number of public high schools in each zip code.

**SELECT zip\_code, COUNT(\*) as school\_count**

**FROM `stoked-producer-388612.12.public\_hs`**

**GROUP BY zip\_code;**

1. Determine the number of public high schools in each state.

**SELECT state\_code, COUNT(\*) as school\_count\_by\_state**

**FROM `stoked-producer-388612.12.public\_hs`**

**GROUP BY state\_code;**

- Use the locale\_code column to understand urbanization levels. Use the CASE statement to display the corresponding locale\_text and locale\_size in your query result.

- Hint: Use the **substr()** function to examine parts of the locale\_code for determining locale\_text and locale\_size.

> The locale\_code column uses numerical codes to represent different types of areas where the high schools are located. These codes can be broken down into two parts:

> The \*first digit\* represents the \*locale\_text\*, indicating the general type of area (e.g., city, suburb, town, rural). It ranges for 1 to 4.

> The \*second digit\* represents the \*locale\_size\*, indicating the size of the area (e.g., large, midsize, small).

> So, we will translate the \*locale\_code\* from \*\*public\_hs\*\* table into useful description to understand the level of urbanization for each high school.

**SELECT**

**school\_id,**

-- first digit of local\_code is locale\_text

**CASE**

**WHEN SUBSTR(locale\_code,1,1) = '1' THEN 'City'**

**WHEN SUBSTR(locale\_code,1,1) = '2' THEN 'Suburb'**

**WHEN SUBSTR(locale\_code,1,1) = '3' THEN 'Town'**

**WHEN SUBSTR(locale\_code,1,1) = '4' THEN 'Rural'**

**ELSE 'Unknown'**

**END AS locale\_text,**

**CASE**

**WHEN SUBSTR(locale\_code,2,1) = '1' THEN 'Large'**

**WHEN SUBSTR(locale\_code,2,1) = '2' THEN 'Midsize'**

**WHEN SUBSTR(locale\_code,2,1) = '3' THEN 'Small'**

**ELSE 'Unknown'**

**END AS locale\_size**

**FROM `stoked-producer-388612.12.public\_hs`;**

**3) Income Analysis :**

1. Calculate the minimum, maximum, and average median household income of the nation.

**SELECT**

**MIN(SAFE\_CAST(median\_household\_income AS INT64)) as min\_income,**

**MAX(SAFE\_CAST(median\_household\_income AS INT64)) as max\_income,**

**AVG(SAFE\_CAST(median\_household\_income AS INT64)) as avg\_income**

**FROM `stoked-producer-388612.12.census\_data`**

**WHERE median\_household\_income IS NOT NULL**

**AND SAFE\_CAST(median\_household\_income AS INT64) IS NOT NULL;**

- Calculate the minimum, maximum, and average median household income for each state.

> Utilize the SAFE\_CAST function, which returns NULL instead of failing when the conversion cannot be performed.

**SELECT**

**state\_code,**

**MIN(SAFE\_CAST(median\_household\_income AS INT64)) as min\_income,**

**MAX(SAFE\_CAST(median\_household\_income AS INT64)) as max\_income,**

**AVG(SAFE\_CAST(median\_household\_income AS INT64)) as avg\_income**

**FROM `stoked-producer-388612.12.census\_data`**

**WHERE median\_household\_income IS NOT NULL**

**AND SAFE\_CAST(median\_household\_income AS INT64) IS NOT NULL**

**GROUP BY state\_code;**

**4) Joint Analysis :**

1. Join tables to analyze further. Investigate if characteristics of the zip-code area, such as median household income, influence students' performance in high school.

**WITH income\_ranges AS (**

**SELECT**

**h.school\_id,**

**l.zip\_code,**

**l.median\_household\_income,**

**CASE**

**WHEN SAFE\_CAST(l.median\_household\_income AS INT64) < 50000 THEN '<$50K'**

**WHEN SAFE\_CAST(l.median\_household\_income AS INT64) BETWEEN 50000 AND 100000 THEN '<$50K-$100K'**

**ELSE '>$100K'**

**END AS income\_range,**

**h.pct\_proficient\_math,**

**h.pct\_proficient\_reading**

**FROM `stoked-producer-388612.12.census\_data` l**

**JOIN `stoked-producer-388612.12.public\_hs` h ON l.zip\_code = h.zip\_code**

**)**

**SELECT**

**income\_range,**

**AVG(SAFE\_CAST(pct\_proficient\_math AS INT64)) AS pct\_proficient\_math,**

**AVG(SAFE\_CAST(pct\_proficient\_reading AS INT64)) AS pct\_proficient\_reading**

**FROM income\_ranges**

**GROUP BY income\_range;**

- \*Hint\*: Use the CASE statement to divide median\_household\_income into income ranges (e.g., <$50k, $50k-$100k, $100k+) and find the average exam scores for each range.

**5) Intermediate Challenge :**

1. Determine if students perform better on the math or reading exam on average.Find the number of states where students perform better in math exams compared to reading exams, and vice versa.

**WITH avg\_exam\_scores AS (**

**SELECT**

**state\_code,**

**AVG(SAFE\_CAST(pct\_proficient\_math AS INT64)) AS avg\_math\_pct,**

**AVG(SAFE\_CAST(pct\_proficient\_reading AS INT64)) AS avg\_reading\_pct,**

**CASE**

**WHEN AVG(SAFE\_CAST(pct\_proficient\_math AS INT64)) > AVG(SAFE\_CAST(pct\_proficient\_reading AS INT64)) THEN 'Math'**

**WHEN AVG(SAFE\_CAST(pct\_proficient\_math AS INT64)) < AVG(SAFE\_CAST(pct\_proficient\_reading AS INT64)) THEN 'Reading'**

**ELSE 'No Exam Data'**

**END AS Higher\_Performance\_In,**

**FROM `stoked-producer-388612.12.public\_hs`**

**GROUP BY state\_code**

**)**

**SELECT Higher\_Performance\_In,**

**COUNT(\*) AS number\_of\_states**

**FROM avg\_exam\_scores**

**GROUP BY Higher\_Performance\_In;**

- \*Hint\*: Use the WITH clause to create a temporary table of average exam scores for each state, including a column indicating whether the math or reading average is higher. Include an option for "No Exam Data" for states without standardized assessments.

**6) Advanced Challenge :**

1. Calculate the average proficiency on state assessment exams for each zip code. Compare the average proficiency to other zip codes within the same state.
2. Note: Exam standards may vary by state, so limit comparisons within states. Some states may not have exams.
3. \*Hint\*: Use the WITH clause to create a temporary table of exam score statistics for each state (e.g., min/max/avg) and then join it to each zip-code level data for comparison.

**WITH zip\_code\_averages AS (**

**SELECT**

**zip\_code,**

**state\_code,**

**AVG(SAFE\_CAST(pct\_proficient\_math AS FLOAT64)) as zip\_avg\_math,**

**AVG(SAFE\_CAST(pct\_proficient\_reading AS FLOAT64)) as zip\_avg\_reading**

**FROM `stoked-producer-388612.12.public\_hs`**

**GROUP BY zip\_code, state\_code**

**),**

**state\_averages AS (**

**SELECT**

**state\_code,**

**AVG(SAFE\_CAST(pct\_proficient\_math AS FLOAT64)) as state\_avg\_math,**

**AVG(SAFE\_CAST(pct\_proficient\_reading AS FLOAT64)) as state\_avg\_reading**

**FROM `stoked-producer-388612.12.public\_hs`**

**group by state\_code**

**)**

**SELECT**

**z.zip\_code,**

**z.state\_code,**

**z.zip\_avg\_math,**

**z.zip\_avg\_reading,**

**s.state\_avg\_math,**

**s.state\_avg\_reading,**

**(z.zip\_avg\_math - s.state\_avg\_math) AS math\_proficiency\_diff,**

**(z.zip\_avg\_reading - s.state\_avg\_reading) AS reading\_proficiency\_diff**

**FROM**

**zip\_code\_averages z**

**JOIN**

**state\_averages s**

**on**

**z.state\_code = s.state\_code**

**ORDER BY**

**z.state\_code, z.zip\_code;**