

```
CREATE TABLE studentdata (  
    id INT,  
    gender VARCHAR(10),  
    race_ethnicity VARCHAR(20),  
    parental_education VARCHAR(30),  
    lunch_type VARCHAR(15),  
    test_preparation VARCHAR(10),  
    math_score INT,  
    reading_score INT,  
    writing_score INT,  
    overall_average NUMERIC,  
    math_tier VARCHAR(15),  
    reading_tier VARCHAR(15),  
    writing_tier VARCHAR(15),  
    overall_tier VARCHAR(15)  
);
```

```
SELECT * from studentdata;
```

```
/*  
average scores by subject  
Swap 'gender' for any other demographic column  
*/
```

```
SELECT  
    gender,  
    COUNT(*) AS total_students,  
    ROUND(AVG(math_score), 2) AS avg_math,  
    ROUND(AVG(reading_score), 2) AS avg_reading,  

```

```
/*
```

```
Student performance breakdown by segment
```

```
Swap 'gender' for any other demographic column
```

```
*/
```

```
SELECT
```

```
gender                AS segment,  
COUNT(*) FILTER (WHERE math_tier = 'Elite')    AS maths_elite,  
COUNT(*) FILTER (WHERE math_tier = 'Above Average') AS maths_above_avg,  
COUNT(*) FILTER (WHERE math_tier = 'Average')   AS maths_average,  
COUNT(*) FILTER (WHERE math_tier = 'Below Average') AS maths_below_avg,  
COUNT(*) FILTER (WHERE reading_tier = 'Elite')   AS reading_elite,  
COUNT(*) FILTER (WHERE reading_tier = 'Above Average') AS reading_above_avg,  
COUNT(*) FILTER (WHERE reading_tier = 'Average')   AS reading_average,  
COUNT(*) FILTER (WHERE reading_tier = 'Below Average') AS reading_below_avg,  
COUNT(*) FILTER (WHERE writing_tier = 'Elite')     AS writing_elite,  
COUNT(*) FILTER (WHERE writing_tier = 'Above Average') AS writing_above_avg,  
COUNT(*) FILTER (WHERE writing_tier = 'Average')     AS writing_average,  
COUNT(*) FILTER (WHERE writing_tier = 'Below Average') AS writing_below_avg,  
COUNT(*) FILTER (WHERE overall_tier = 'Elite')     AS overall_elite,  
COUNT(*) FILTER (WHERE overall_tier = 'Above Average') AS overall_above_avg,  
COUNT(*) FILTER (WHERE overall_tier = 'Average')     AS overall_average,  
COUNT(*) FILTER (WHERE overall_tier = 'Below Average') AS overall_below_avg
```

```
FROM studentdata
```

```
GROUP BY gender
```

```
ORDER BY gender;
```

```
/*
```

```
Segment Top Academic Performers using Window functions
```

```
Swap 'gender' for any other demographic column
```

```
*/
```

```
WITH RankedStudents AS (  
    SELECT  
        parental_education,  
        lunch_type,  
        overall_average,  
        DENSE_RANK() OVER (PARTITION BY parental_education ORDER BY overall_average DESC)  
    AS internal_rank  
    FROM studentdata  
)  
SELECT *  
FROM RankedStudents  
WHERE internal_rank <= 3;
```