- **How many entries do you have in your database who have applied for Fall 2025?**

```
query = """
        SELECT COUNT(term) FROM applicants
        WHERE term = 'Fall 2025';
        """
```

Since the question asks “how many”,we are looking for a numeric result. To find this, we use the `COUNT()` function on the “term” column to count how many entries match the criteria. We query the applicants table and filter for rows where the term is equal to 'Fall 2025'.

- **What percentage of entries are from international students (not American or Other) (to two decimal places)?**

```
query = """
        SELECT ROUND(
         100.0 * COUNT(us_or_international)
         FILTER (WHERE us_or_international = 'International')/ COUNT(*), 2)
        FROM applicants;
         """
```

To calculate the percentage, we want to find the ratio of international applicants to the total number of applicants, then round the result to two decimal places.

- `COUNT(us_or_international) FILTER (WHERE us_or_international = 'International')` – this command counts only the filtered rows, where ‘us_or_international’ is equal to ‘International’.
- `COUNT(*)` – counts all rows in applicants (the total number of applicants)
- `SELECT ROUND(…,2)` – returns a numerical value, rounded to two decimal places.
- To get a percentage, we multiply the ratio by 100.

- **What is the average GPA, GRE, GRE V, GRE AW of applicants who provide these metrics?**

```
query = """
        SELECT AVG(GPA), AVG(GRE), AVG(GRE_V), AVG(GRE_AW)
        FROM applicants
        WHERE term = 'Fall 2025'
        AND us_or_international = 'International'
        AND GPA BETWEEN 0 AND 4.0
        AND GRE BETWEEN 260 AND 340
        AND GRE BETWEEN 130 AND 170
        AND GRE_AW BETWEEN 0 AND 6
        AND MOD((GRE_AW * 10)::INT, 5) = 0;
        """
```

  We want to retrieve the average GPA, GRE, GRE V and GRE AW scores of international applicants who applied for Fall 2025. The `SELECT` statement specifies the averages we want to calculate, and we query the applicants table and filter for rows where 'term' is 'Fall 2025' and 'us_or_international' is 'International'. Moreover, to ensure averages are based on valid score inputs, I applied the following filters:

  - `AND GPA BETWEEN 0 AND 4.0` – ensures GPA is within the standard US scale
  - `AND GRE BETWEEN 260 AND 340` – ensures GRE is within the valid range
  - `AND GRE_V BETWEEN 130 AND 170` – ensures GRE V is within the valid range
  - `AND GRE_AW BETWEEN 0 AND 6` – ensures GRE AW is within the valid range
  - `AND MOD((GRE_AW * 10)::INT, 5) = 0` – ensures GRE AW values are in 0.5 increments. Since the `MOD` function works only with integers, we multiply `GRE_AW` by `10` and cast it as an integer to the `MOD` function. We then use `MOD(..., 5) = 0` to check if the number is divisible by 5.

- **What is the average GPA of American students in Fall 2025?**

```
query = """
        SELECT AVG(GPA) FROM applicants
        WHERE term = 'Fall 2025'
        AND us_or_international = 'American'
        AND GPA BETWEEN 0 AND 4.0;
        """
```

Similar to the previous query, the `SELECT` statement specifies that we want the average GPA as an output. We query the applicants table and filter for rows where ‘term’ is ‘Fall 2025’, ‘us_or_international’ is ‘American’ and we ensure that GPA is within the standard US scale.

- **What percent of entries for Fall 2025 are Acceptances (to two decimal places)?**

```
query = """
        SELECT ROUND(
        100.0 * COUNT(status)
        FILTER (WHERE status LIKE '%Accepted%' AND term = 'Fall 2025') /
        COUNT(*), 2)
        FROM applicants;
        """
```

To calculate the percentage, we want to find the ratio of applicants who applied to Fall 2025 and were accepted to the total number of applicants, then round the result to two decimal places.

  - `COUNT(status) FILTER (WHERE status LIKE '%Accepted%' AND term = 'Fall 2025')` – this command counts only the filtered rows, where ‘status’ contains the string ‘Accepted’ and the ‘term’ equals ‘Fall 2025’. The `LIKE` operator is used to search for patterns within the column, and the ‘%’ symbol represents any sequence of characters before or after the specified text. This allows us to retrieve all entries that include ‘Accepted…’.
  - `COUNT(*)` – counts all rows in applicants (the total number of applicants)
  - `SELECT ROUND(…,2)` – returns a numerical value, rounded to two decimal places.
  - To get a percentage, we multiply the ratio by 100.

- **What is the average GPA of applicants who applied for Fall 2025 who are Acceptances?**

```
query = """
        SELECT AVG(GPA) FROM applicants
        WHERE term = 'Fall 2025'
        AND status = status LIKE '%Accepted%'
        AND GPA BETWEEN 0 AND 4.0;
        """
```

The `SELECT` statement specifies that we want the average GPA as an output. We query the applicants table and filter for rows where 'term' is 'Fall 2025' and 'status' contains the word 'Accepted'. The `LIKE` operator is used to search for patterns within the column, and the '%' symbol represents any sequence of characters before or after the specified text. This allows us to retrieve all entries that include 'Accepted on…'. Additionally, we filter for GPA values that fall within the standard US scale by including `AND GPA BETWEEN 0 AND 4.0`.

- **How many entries are from applicants who applied to JHU for a master's degrees in Computer Science?**

```
query = """
        SELECT COUNT(*) FROM applicants
        WHERE (program LIKE '%JHU%' OR LOWER(program) LIKE '%hopkins%')
        AND (program LIKE '% CS %' OR LOWER(program) LIKE '%computer science%')
        AND degree = 'Masters';
         """
```

Since the question asks "how many", the result should return a numeric value. To find this, I use the `COUNT(*)` function to count how many entries match the criteria. We query the applicants table and filter for rows where the 'degree' is 'Masters', and the 'program' contains wither 'JHU' or 'hopkins', and either ' CS ' or 'computer science'. To make the search case-insensitive, I apply `LOWER(program)` to convert the full program name and university to lowercase. As in the previous query, the `LIKE` operator is used to search for patterns within the column, and the '%' symbol represents any sequence of characters before or after the specified text.