

Module 3: Database Query Analysis

Generated on: 2026-02-07 10:48:41

Q: How many total applications were submitted for the Fall 2025 term?

Answer:

30 applications

Logic: Counts all rows where the term starts with 'Fall 2025'.

SQL:

```
SELECT COUNT(*) FROM applicants WHERE term LIKE 'Fall 2025%';
```

Q: What percentage of all applicants are international (non-American)?

Answer:

44.24%

Logic: Calculates the ratio of applicants who are not 'American' or 'Other' against the total count.

SQL:

```
SELECT ROUND(100.0 * COUNT(*) / (SELECT COUNT(*) FROM applicants), 2) FROM applicants WHERE us_or_international NOT IN ('American', 'Other');
```

Q: What are the average GPA and GRE scores (Quant, Verbal, AW) across the entire dataset?

Answer:

GPA: 3.80 | GRE Quant: 279.68 | GRE Verbal: 161.24 | GRE AW: 4.20

Logic: Computes the average for GPA and all GRE sections, rounding to 2 decimal places.

SQL:

```
SELECT ROUND(AVG(gpa)::numeric, 2), ROUND(AVG(gre)::numeric, 2), ROUND(AVG(gre_v)::numeric, 2), ROUND(AVG(gre_aw)::numeric, 2) FROM applicants;
```

Q: What is the average GPA of American students who applied for Fall 2025?

Answer:

3.78

Logic: Filters for American students in Fall 2025 and averages their GPA.

SQL:

```
SELECT ROUND(AVG(gpa)::numeric, 2) FROM applicants WHERE us_or_international = 'American' AND term LIKE 'Fall 2025%';
```

Q: What is the overall acceptance rate for the Fall 2025 term?

Answer:

50.00%

Logic: Divides the number of 'Accepted' students by the total number of applicants for Fall 2025.

SQL:

```
SELECT ROUND(100.0 * COUNT(*) / NULLIF((SELECT COUNT(*) FROM applicants WHERE term LIKE 'Fall 2025%'), 0), 2) FROM applicants WHERE status = 'Accepted' AND term LIKE 'Fall 2025%';
```

Q: What is the average GPA of students who were accepted for Fall 2025?

Answer:

3.77

Logic: Averages the GPA only for students with 'Accepted' status in Fall 2025.

SQL:

```
SELECT ROUND(AVG(gpa)::numeric, 2) FROM applicants WHERE status = 'Accepted' AND term LIKE 'Fall 2025%';
```

Q: How many applicants for a Masters in Computer Science applied to Johns Hopkins (JHU)?

Answer:

1 applicants

Logic: Filters for JHU (using wildcards) and Masters CS programs.

SQL:

```
SELECT COUNT(*) FROM applicants WHERE (university ILIKE '%JHU%' OR university ILIKE '%Johns Hopkins%') AND degree = 'Masters' AND program ILIKE '%Computer Science%';
```

Q: Using original fields: How many CS PhD applicants were accepted to Georgetown, MIT, Stanford, or CMU in 2025?

Answer:

0 applicants

Logic: JUSTIFICATION: We used the 'date_added' field because the question asks for acceptances 'in 2025' (the calendar year the result was received), rather than 'for the term of 2025'. This distinguishes immediate results from future term starts.

SQL:

```
SELECT COUNT(*) FROM applicants WHERE EXTRACT(YEAR FROM date_added) = 2025 AND status = 'Accepted' AND university IN ('Georgetown University', 'MIT', 'Stanford University', 'Carnegie Mellon University') AND degree = 'PhD' AND program ILIKE '%Computer Science%';
```

Q: Using LLM fields: How many CS PhD applicants were accepted to Georgetown, MIT, Stanford, or CMU in 2025?

Answer:

0 applicants

Logic: JUSTIFICATION: Same logic as Q8 (using 'date_added' for calendar year). We use the LLM-cleaned university names here to capture variations like 'Carnegie Mellon' vs 'CMU'.

SQL:

```
SELECT COUNT(*) FROM applicants WHERE EXTRACT(YEAR FROM date_added) = 2025 AND status = 'Accepted' AND llm_generated_university IN ('Georgetown University', 'MIT', 'Stanford University', 'Carnegie Mellon University') AND degree = 'PhD' AND llm_generated_program ILIKE '%Computer Science%';
```

Q: Extra Q1: Which academic program has the highest volume of application entries?

Answer:

Clinical Psychology (352 applications)

Logic: Groups by program name and sorts descending to find the most popular one.

SQL:

```
SELECT llm_generated_program, COUNT(*) as count FROM applicants GROUP BY 1 ORDER BY 2 DESC LIMIT 1;
```

Q: Extra Q2: How does the average GRE Quantitative score compare between PhD and Masters applicants?

Answer:

Masters: 268.53 | PhD: 283.84

Logic: Groups by degree type to compare average GRE scores side-by-side.

SQL:

```
SELECT degree, ROUND(AVG(gre)::numeric, 2) as avg_q FROM applicants WHERE degree IN ('PhD', 'Masters') GROUP BY 1;
```