

Admission Data Warehouse

Bilugan, Piere Paolo B.

Desacola, Allen John Y.

Leynes, Chris Michael

Pereyra, Matthew Alen P.

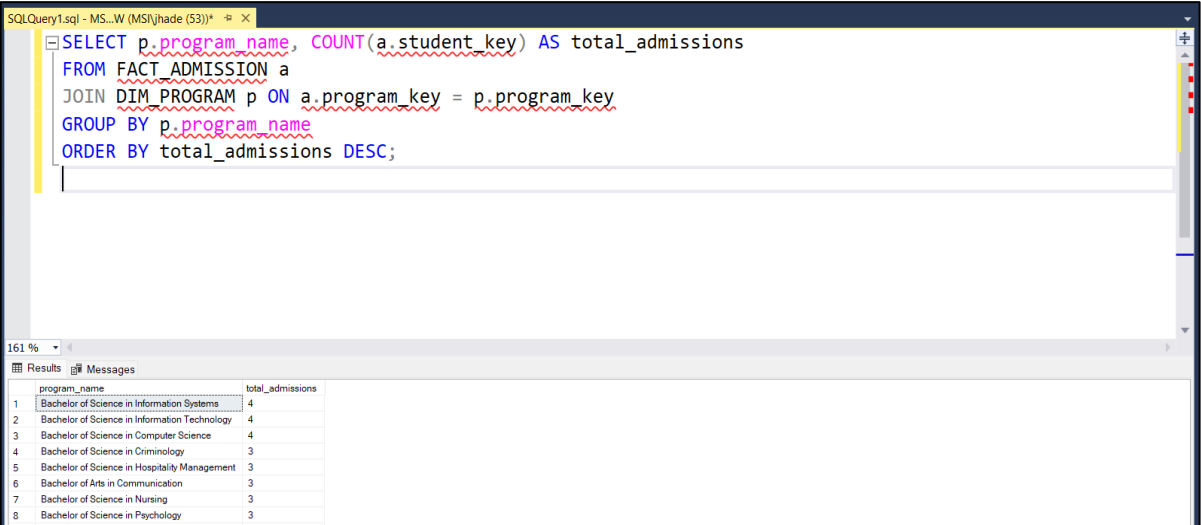
Villanueva, Jhade M.

Data Warehouse Name and its purpose

Our data warehouse is named "**Student Admission**", and its purpose is to **track and analyze student enrollments**, providing valuable insights to support informed decision-making and institutional planning.

5 Business Questions

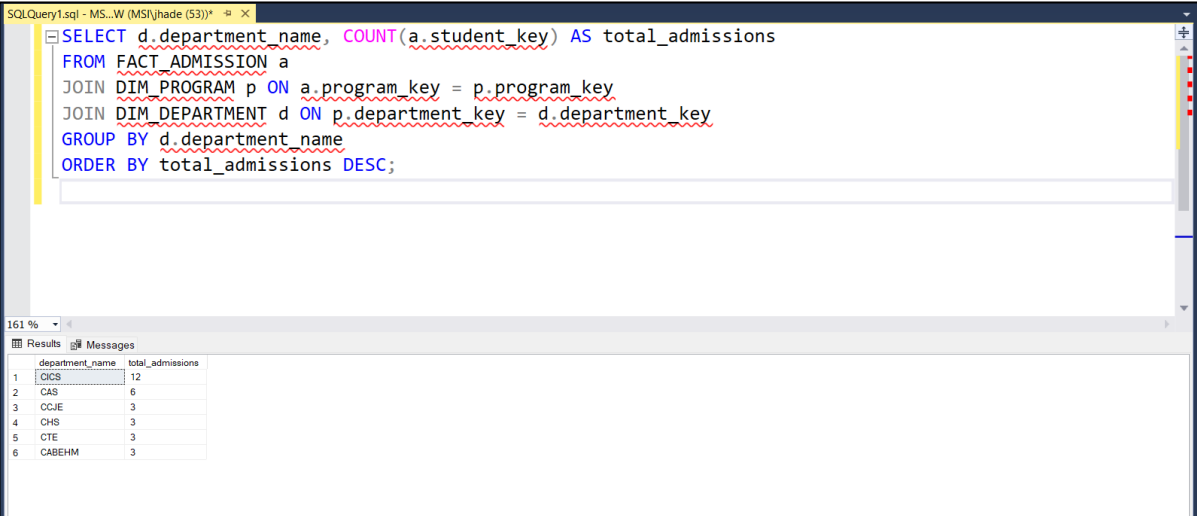
1. How many students were admitted per program?



```
SELECT p.program_name, COUNT(a.student_key) AS total_admissions
FROM FACT_ADMISSION a
JOIN DIM_PROGRAM p ON a.program_key = p.program_key
GROUP BY p.program_name
ORDER BY total_admissions DESC;
```

program_name	total_admissions
Bachelor of Science in Information Systems	4
Bachelor of Science in Information Technology	4
Bachelor of Science in Computer Science	4
Bachelor of Science in Criminology	3
Bachelor of Science in Hospitality Management	3
Bachelor of Arts in Communication	3
Bachelor of Science in Nursing	3
Bachelor of Science in Psychology	3

2. Which department has the highest number of admissions?



```
SELECT d.department_name, COUNT(a.student_key) AS total_admissions
FROM FACT_ADMISSION a
JOIN DIM_PROGRAM p ON a.program_key = p.program_key
JOIN DIM_DEPARTMENT d ON p.department_key = d.department_key
GROUP BY d.department_name
ORDER BY total_admissions DESC;
```

department_name	total_admissions
OICS	12
CAS	6
CCJE	3
CHS	3
CTE	3
CABEHM	3

3. How many students have enrolled in total?

The screenshot shows a SQL query window with the following text:

```
SELECT COUNT(*) AS total_enrollees FROM FACT_ADMISSION;
```

Below the query window, the 'Results' tab is active, displaying a single row with the value 30 in the 'total_enrollees' column.

	total_enrollees
1	30

4. What is the gender distribution of enrollees?

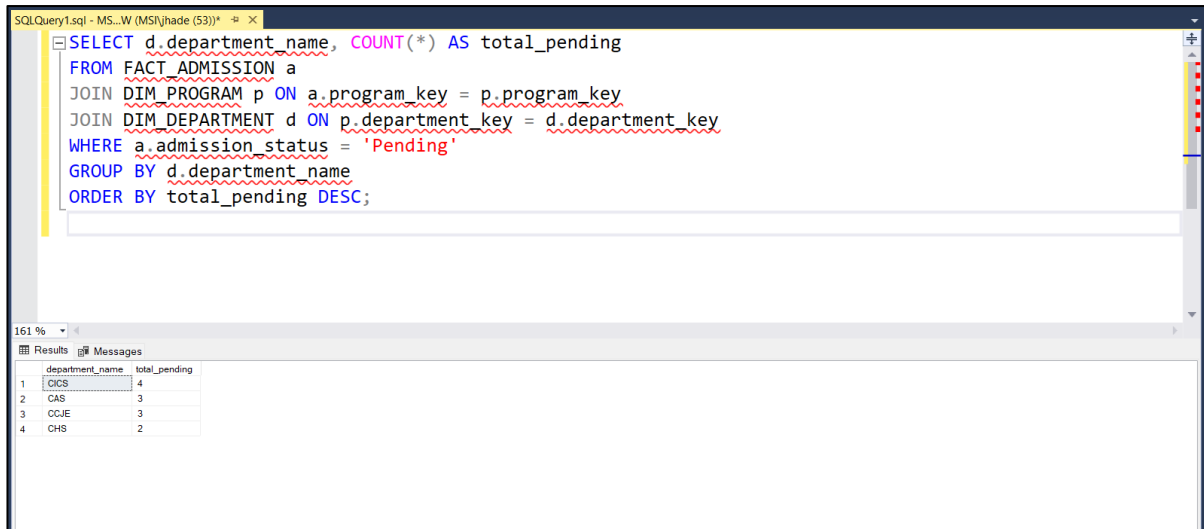
The screenshot shows a SQL query window with the following text:

```
SELECT s.gender, COUNT(DISTINCT a.student_key) AS total_enrollees  
FROM FACT_ADMISSION a  
JOIN DIM_STUDENT s ON a.student_key = s.student_key  
GROUP BY s.gender;
```

Below the query window, the 'Results' tab is active, displaying two rows of data. The first row shows 'F' for gender and 15 for total_enrollees. The second row shows 'M' for gender and 15 for total_enrollees.

	gender	total_enrollees
1	F	15
2	M	15

5. How many students have pending admissions per department?



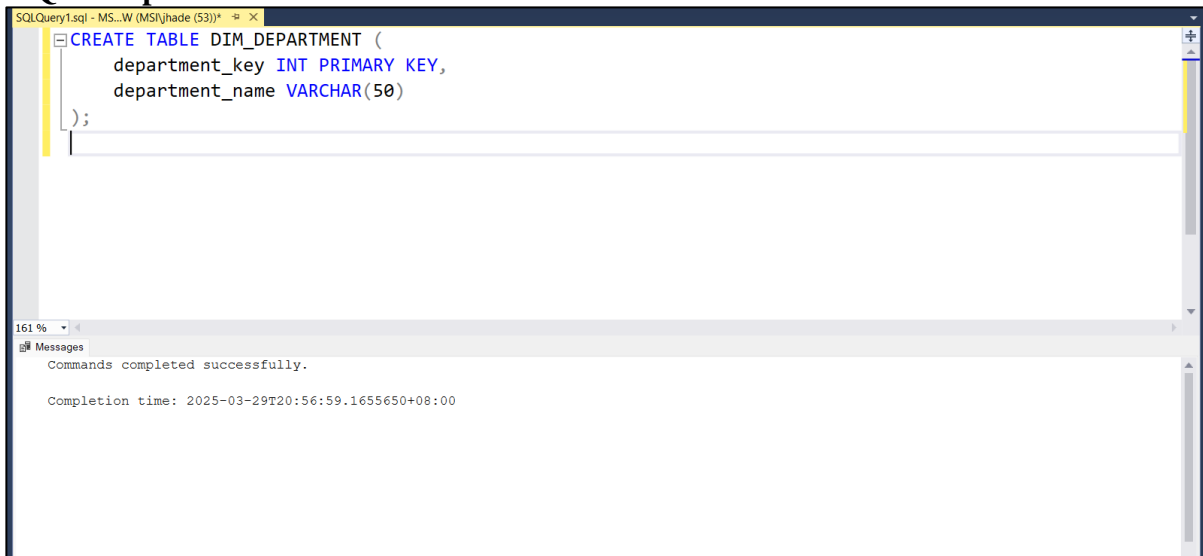
The screenshot shows a SQL query window with the following code:

```
SELECT d.department_name, COUNT(*) AS total_pending
FROM FACT_ADMISSION a
JOIN DIM_PROGRAM p ON a.program_key = p.program_key
JOIN DIM_DEPARTMENT d ON p.department_key = d.department_key
WHERE a.admission_status = 'Pending'
GROUP BY d.department_name
ORDER BY total_pending DESC;
```

Below the query window, the Results pane displays the following data:

	department_name	total_pending
1	QICS	4
2	CAS	3
3	CCJE	3
4	CHS	2

SQL Scripts to create the table



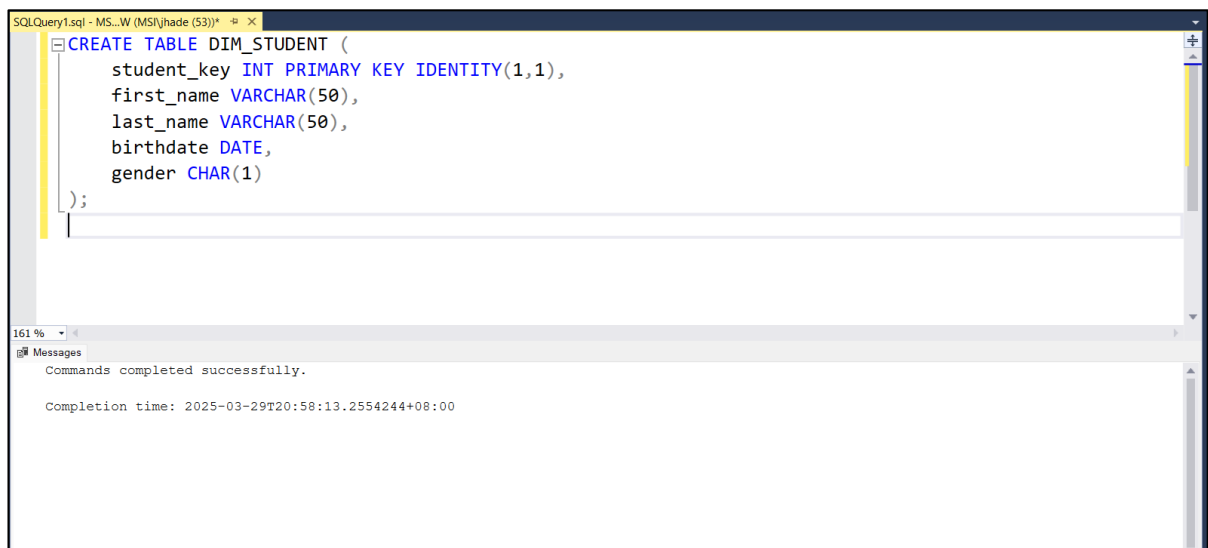
The screenshot shows a SQL query window with the following code:

```
CREATE TABLE DIM_DEPARTMENT (
    department_key INT PRIMARY KEY,
    department_name VARCHAR(50)
);
```

Below the query window, the Messages pane displays the following text:

Commands completed successfully.

Completion time: 2025-03-29T20:56:59.1655650+08:00



The screenshot shows a SQL query window with the following code:

```
CREATE TABLE DIM_STUDENT (
    student_key INT PRIMARY KEY IDENTITY(1,1),
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    birthdate DATE,
    gender CHAR(1)
);
```

Below the query window, the Messages pane displays the following text:

Commands completed successfully.

Completion time: 2025-03-29T20:58:13.2554244+08:00

```
SQLQuery1.sql - MS...W (MSI\jshade (53)) * X
CREATE TABLE DIM_PROGRAM (
    program_key INT PRIMARY KEY,
    program_name VARCHAR(100),
    department_key INT,
    FOREIGN KEY (department_key) REFERENCES DIM_DEPARTMENT(department_key)
);
```

161 %
Messages
Commands completed successfully.
Completion time: 2025-03-29T20:57:56.3653298+08:00

```
SQLQuery1.sql - MS...W (MSI\jshade (53)) * X
CREATE TABLE DIM_DATE (
    date_key INT PRIMARY KEY,
    full_date DATE,
    year INT,
    month INT,
    day INT,
    quarter INT
);
```

161 %
Messages
Commands completed successfully.
Completion time: 2025-03-29T20:58:24.3654415+08:00

```
SQLQuery1.sql - MS...W (MSI\jshade (53)) * X
CREATE TABLE FACT_ADMISSION (
    admission_id INT PRIMARY KEY IDENTITY(1,1),
    student_key INT,
    program_key INT,
    admission_date_key INT,
    admission_status VARCHAR(20),
    FOREIGN KEY (student_key) REFERENCES DIM_STUDENT(student_key),
    FOREIGN KEY (program_key) REFERENCES DIM_PROGRAM(program_key),
    FOREIGN KEY (admission_date_key) REFERENCES DIM_DATE(date_key)
);
```

161 %
Messages
Commands completed successfully.
Completion time: 2025-03-29T20:58:36.8955679+08:00

SQL Queries along with the screenshot of the queries result.

```
SQLQuery1.sql - MS...W (MSI\jhade (53)) *  X
INSERT INTO DIM_PROGRAM (program_key, program_name, department_key)
VALUES
(101, 'Bachelor of Science in Information Technology', 1),
(102, 'Bachelor of Science in Computer Science', 1),
(103, 'Bachelor of Science in Information Systems', 1),
(201, 'Bachelor of Arts in Communication', 2),
(202, 'Bachelor of Science in Psychology', 2),
(301, 'Bachelor of Science in Nursing', 3),
(401, 'Bachelor of Science in Hospitality Management', 4),
(501, 'Bachelor of Science in Criminology', 5),
(601, 'Bachelor of Science in Secondary Education', 6);

161 %
Messages
(9 rows affected)
Completion time: 2025-03-29T20:59:19.7152388+08:00
```

```
SQLQuery1.sql - MS...W (MSI\jhade (53)) *  X
INSERT INTO DIM_DATE (date_key, full_date, year, month, day, quarter)
VALUES
(20250329, '2025-03-29', 2025, 3, 29, 1),
(20250330, '2025-03-30', 2025, 3, 30, 1);

161 %
Messages
(2 rows affected)
Completion time: 2025-03-29T20:59:59.9907015+08:00
```

```
SQLQuery1.sql - MS...W (MSI\jhade (53)) *  X
INSERT INTO FACT_ADMISSION (student_key, program_key, admission_date_key, admission_status)
VALUES
(1, 101, 20250329, 'Enrolled'),
(2, 102, 20250329, 'Enrolled'),
(3, 103, 20250330, 'Pending'),
(4, 201, 20250329, 'Enrolled'),
(5, 202, 20250330, 'Pending'),
(6, 301, 20250329, 'Enrolled'),
(7, 401, 20250330, 'Enrolled'),
(8, 501, 20250329, 'Pending'),
(9, 601, 20250330, 'Enrolled'),
(10, 101, 20250330, 'Enrolled'),
(11, 102, 20250330, 'Enrolled'),
(12, 103, 20250330, 'Enrolled'),
(13, 201, 20250330, 'Enrolled'),
(14, 202, 20250330, 'Enrolled'),
(15, 301, 20250330, 'Enrolled'),
(16, 401, 20250330, 'Enrolled'),
(17, 501, 20250330, 'Enrolled'),
(18, 601, 20250330, 'Enrolled'),
(19, 101, 20250330, 'Enrolled'),
(20, 102, 20250330, 'Enrolled'),
(21, 103, 20250330, 'Enrolled'),
(22, 201, 20250330, 'Enrolled'),
(23, 202, 20250330, 'Enrolled'),
(24, 301, 20250330, 'Enrolled'),
(25, 401, 20250330, 'Enrolled'),
(26, 501, 20250330, 'Enrolled'),
(27, 601, 20250330, 'Enrolled'),
(28, 101, 20250330, 'Enrolled'),
(29, 102, 20250330, 'Enrolled'),
(30, 103, 20250330, 'Enrolled');

161 %
Messages
(30 rows affected)
Completion time: 2025-03-29T21:01:29.6284948+08:00
```

