STUDENT PERFORMANCE ANALYSIS USING SQL

Project Overview

This project showcases SQL-driven analysis of student engagement and performance data from an online learning platform. By leveraging structured queries, the project examines key variables such as education level, learning style, course enrollment, and interaction metrics to uncover patterns that influence academic outcomes. The objective is to identify correlations between student behavior and final exam performance, as well as factors contributing to dropout likelihood. These insights can provide data-backed strategies to enhance student retention and optimize learning experiences.

Dataset Overview

Column Name	Description	
Student_ID	Unique identifier for each student	
Age	Age of the student	
Gender	Student's gender	
Education_Level	Academic qualification (e.g., High School, Undergraduate)	
Course_Name	Course the student is enrolled in	
Time_Spent_on_Videos	Total time (in minutes) spent watching video lectures	
Quiz_Attempts	Number of quizzes attempted	
Quiz_Scores	Average score obtained in quizzes	
Forum_Participation	Level of interaction in course discussion forums	
Assignment_Completion_Rate	Percentage of assignments completed	
Engagement_Level	Overall engagement level (Low / Medium / High)	
Final_Exam_Score	Score obtained in the final exam	
Learning_Style	Preferred learning method (e.g., Visual, Reading/Writing)	
Feedback_Score	Feedback rating provided by the student	
Dropout_Likelihood	Indicates if the student is likely to drop out (Yes / No)	

Problem Statement

1.Identify all the undergraduate students and display their id, gender and course name.

SQL Query

select Student_ID,Gender,Course_Name from `student-learning-case-study.students_dataset.students` where Education_Level = 'Undergraduate';

Sample Output

Row	Student_ID ▼	Gender ▼	Course_Name ▼
1	S00180	Male	Python Basics
2	S00229	Male	Data Science
3	S00268	Male	Machine Learning
4	S00318	Male	Cybersecurity

Explanation

Used Dataset Table

student-learning-case-study.students_dataset.students

Each row represents a unique student with demographic, behavioral, and academic data.

Query Logic

- **SELECT Student_ID, Gender, Course_Name**: Retrieves only the essential columns for analysis.
- **FROM**: Specifies the dataset/table.
- WHERE Education_Level = 'Undergraduate': Filters rows where students are classified as undergraduate.

Insight

This query helps in understanding:

- Course preferences among undergraduate students
- Gender trends for specific courses
- Acts as a foundation for further analysis like dropout likelihood by course or performance trends across
 education levels.

Problem Statement

Fetch details of female students who spent over 300 minutes on video lectures and have a high likelihood of dropping out.

SQL Query

select Student_ID ,Age,Course_Name,Time_Spent_on_Videos,Dropout_Likelihood from `student-learning-case-study.students_dataset.students`

where Gender='Female' and Time_Spent_on_Videos > 300 and Dropout_Likelihood=true;

Row Student_ID ▼ Course_Name ▼ Time_Spent_on_Vide Dropout_Lik Age ▼ 1 S08453 44 Machine Learning 498 true 2 S08152 17 Python Basics 317 true S04978 3 19 Web Development 306 S05062 37 Cybersecurity 483 true

Explanation



student-learning-case-study.students_dataset.students

This table contains individual student records, including demographics, engagement metrics, and academic indicators.

Query Logic

- **SELECT Student_ID, Age, Course_Name, Time_Spent_on_Videos, Dropout_Likelihood**: Extracts relevant fields for identifying targeted student profiles.
- **FROM**: Specifies the source dataset.
- WHERE Gender = 'Female': Filters for female students only.
- AND Time_Spent_on_Videos > 300: Further restricts the results to students with high video engagement.
- AND Dropout_Likelihood = TRUE: Narrows the selection to those flagged as likely to drop out.

Insight

This query provides valuable insight into:

- Patterns of high engagement yet high dropout risk among female students.
- Potential misalignment between content delivery and learner needs.
- Early intervention opportunities to support at-risk yet active learners.
 These findings can inform targeted support strategies and improve retention efforts.

Problem Statement

Identify students who have enrolled in either the *Machine Learning* or *Data Science* course and have either attempted at least two quizzes or scored more than 60 in quizzes. The output should be sorted by forum participation in ascending order.

SQL Query

select Course_Name,Student_id,Quiz_Attempts,Forum_Participation from `student-learning-case-study.students_dataset.students`
where Course_Name in('Machine Learning','Data Science')
and(Quiz_Attempts >=2
or Quiz_Scores >60)
order by Forum_Participation asc;

Sample Output

Row	Course_Name ▼	Student_id ▼	Quiz_Attempts ▼	Forum_Participation
1	Data Science	S07261	3	0
2	Machine Learning	S04384	1	0
3	Machine Learning	S05285	3	0
4	Data Science	S09082	3	0

Explanation

Used Dataset Table

student-learning-case-study.students_dataset.students

This table includes detailed student attributes covering courses, quiz participation, academic scores, and forum activity.

Query Logic

- **SELECT Student_ID, Course_Name, Quiz_Attempts, Quiz_Scores, Forum_Participation**: Retrieves key attributes relevant to the course and performance indicators.
- FROM: Points to the main dataset.
- WHERE Course_Name IN ('Machine Learning', 'Data Science'): Filters students enrolled in either of the two specified courses.
- AND (Quiz_Attempts >= 2 OR Quiz_Scores > 60): Ensures that only students who are academically engaged—either through multiple quiz attempts or high scores—are included.
- **ORDER BY Forum_Participation ASC**: Sorts the results starting with students who participated the least in discussion forums.

Insight

This query helps in:

- Identifying engaged learners in specific technical courses.
- Understanding the relationship between academic activity and forum engagement.
- Pinpointing students who are academically involved but not socially active in learning communities useful for designing engagement-boosting strategies.

Problem Statement

Determine the number of students who prefer learning through visual methods.

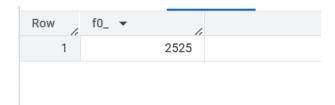
SQL Query

select count(Student_id)

from 'student-learning-case-study.students dataset.students'

where Learning_Style ='Visual';

Sample Output



Explanation

Used Dataset Table

student-learning-case-study.students_dataset.students

This dataset contains individual-level student information, including demographics, academic metrics, and preferred learning styles.

Query Logic

- **SELECT COUNT(Student_ID)**: Counts the total number of students matching the specified condition.
- FROM: Refers to the student dataset.
- WHERE Learning_Style = 'Visual': Filters records to include only those students who have selected 'Visual' as their preferred learning method.

Insight

This query reveals:

- The number of students who favor visual content, such as videos, diagrams, and visual demonstrations.
- Helps curriculum designers and educators tailor content formats to match the predominant learning preferences.
- Supports data-driven content delivery strategies for enhanced learner engagement and performance.

Problem Statement

Identify the minimum and maximum final exam scores among students who have achieved the highest assignment completion rate.

SQL Query

select max(Final_Exam_Score) as maximum_score,min(Final_Exam_Score) as Minimum_score, from `student-learning-case-study.students_dataset.students` where Assignment_Completion_Rate=(select max(Assignment_Completion_Rate) from `student-learning-case-study.students_dataset.students`);

Sample Output



Explanation



student-learning-case-study.students_dataset.students

This dataset captures student demographics, engagement metrics, and academic outcomes such as final exam scores and assignment completion rates.

- **SELECT MAX(Final_Exam_Score), MIN(Final_Exam_Score)**: Computes the highest and lowest final exam scores.
- **FROM**: References the main dataset.
- WHERE Assignment_Completion_Rate = (SELECT MAX(...)): Filters the records to include only those students who have attained the maximum assignment completion rate in the entire dataset.
- **Subquery**: Dynamically identifies the highest assignment completion rate.

This query helps to:

- Evaluate performance variability among the most diligent students in terms of assignment submission.
- Understand whether consistent assignment completion correlates with consistently high exam performance.
- Support strategies that encourage assignment participation to potentially improve exam outcomes.

Problem Statement

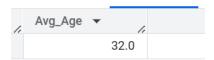
Determine the average age (rounded to the nearest whole number) of female students who are either Undergraduate or Postgraduate, have enrolled in the *Cybersecurity* course, and meet at least one of the following criteria:

- Have a high engagement level
- Prefer a reading/writing learning style
- Have participated in at least 20 forums

SQL Query

```
select round(avg(Age)) as Avg_Age from `student-learning-case-study.students_dataset.students` where Education_Level in('Undergraduate','Postgraduate') and Gender in ('Female') and Course_Name = 'Cybersecurity' and (Engagement_Level = 'High' or (Learning_Style ='Reading/Writing' or Forum_Participation >= 20));
```

Sample Output



Explanation



student-learning-case-study.students_dataset.students

This dataset contains individual records with demographic, course-related, and behavioral information, including age, gender, education level, course name, learning preferences, and engagement metrics.

- ROUND(AVG(Age)): Computes the average age and rounds the result to the nearest whole number.
- WHERE Education_Level IN (...): Filters for students who are either Undergraduate or Postgraduate.
- AND Gender = 'Female': Restricts the selection to female students.
- AND Course_Name = 'Cybersecurity': Includes only those who enrolled in the Cybersecurity course.
- AND (...): Ensures students meet at least one of the following:
 - High engagement level
 - o Reading/Writing as learning style
 - o Forum participation of 20 or more

This query offers insights into:

- The age demographic of highly engaged female learners in technical fields like cybersecurity
- Patterns in learning preferences and social engagement within targeted groups
- Supporting user-centric course design based on age and engagement tendencies

Problem Statement

Identify all the male postgraduate student or female undergraduate student with final score over 90. Output should contains 2 columns, 1 with the student id and second column should indicate male postgraduate student as "Master degree" and female undergraduate student as "Bachelor degree"

SQL Query

select Student ID,

Case

When Education_Level = 'Postgraduate' then 'Master degree'

When Education_Level = 'Undergraduate' then 'Bachelor degree'

else Education_Level

End as Education_Level_category

from `student-learning-case-study.students_dataset.students`

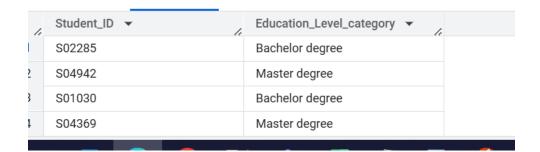
where

(Gender = 'Male' and Education_Level = 'Postgraduate' and Final_Exam_Score>90)

or

(Gender = 'Female' and Education_Level = 'Undergraduate' and Final_Exam_Score>90);

Sample Output



Explanation

Used Dataset Table

student-learning-case-study.students_dataset.students

This dataset includes student demographics, educational levels, course data, and academic performance indicators such as final exam scores.

- CASE WHEN: Used to assign a human-readable label based on education level.
 - o 'Master degree' is assigned to postgraduate students.
 - o 'Bachelor degree' is assigned to undergraduate students.
- WHERE clause: Filters records based on:
 - Male + Postgraduate + Final_Exam_Score > 90
 - Female + Undergraduate + Final Exam Score > 90

This query allows us to:

- Spot top-performing students from distinct gender and education-level groups
- Derive categorized outputs for clearer academic analysis
- Provide personalized academic labels for advanced reporting and segmentation

Problem Statement

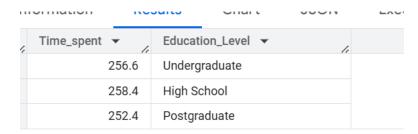
Calculate the average time (in minutes) spent watching videos, grouped by education level.

Only include students whose age is an even number, and round the average value to one decimal place.

SQL Query

select round(avg(Time_Spent_on_Videos),1) as Time_spent,Education_Level from `students_dataset.students` where mod(Age,2)=0 group by Education_Level;

Sample Output



Explanation



students_dataset.students

This dataset tracks student-level metrics such as age, education background, and time spent engaging with video content on the platform.

- ROUND(AVG(...), 1): Calculates the average video watch time for each education level, rounded to one decimal place.
- MOD(Age, 2) = 0: Filters for students with even-numbered ages using the modulus function.
- GROUP BY Education_Level: Aggregates the average by each distinct level of education.

This query reveals:

- How students across different educational backgrounds engage with video content
- Whether age (specifically even-numbered) correlates with longer video consumption
- Useful inputs for tailoring video-based learning content according to education segments

Problem Statement

Determine the most preferred course and learning style among students who are over 40 years of age.

Analyze the count of course enrollments and preferred learning styles for this age group to identify trends.

SQL Query

select * from `student-learning-case-study.students_dataset.students`;

select Course_Name,Learning_Style, count(Course_Name) as count_of_course_name_prefered,count(Learning_Style)as count_of_learning_style_prefered from `student-learning-case-study.students_dataset.students` where Age > 40 group by Course_Name,Learning_Style order by Course_name,count(Course_Name),count(Learning_Style) asc;

Sample Output

Course_Name ▼	Learning_Style ▼	count_of_course_nar	count_of_learning_st	
Cybersecurity	Reading/Writing	127	127	
Cybersecurity	Auditory	128	128	
Cybersecurity	Visual	129	129	
Cybersecurity	Kinesthetic	139	139	

Explanation



student-learning-case-study.students_dataset.students

This dataset contains student demographic and academic data, including age, course enrollment, and preferred learning styles.

- WHERE Age > 40: Filters data to include only students older than 40.
- GROUP BY Course_Name, Learning_Style: Aggregates results based on course and learning style combinations.
- COUNT(Course_Name), COUNT(Learning_Style): Computes how many students over 40 are enrolled in each course and prefer each learning style.
- ORDER BY Course_Name, COUNT(...) ASC: Orders results alphabetically by course and in ascending order
 of counts.

This query reveals:

- Course and learning style preferences of older students
- Which subjects attract more engagement among students over 40
- Valuable trends for designing age-appropriate learning paths and resources

Problem Statement

Identify the courses that are enrolled by more than 2000 students.

This analysis helps in determining the most popular courses based on enrollment volume.

SQL Query

SELECT Course_Name, COUNT(Student_ID) AS Total_Enrolled_Students FROM `student-learning-case-study.students_dataset.students` GROUP BY Course_Name HAVING COUNT(Student_ID) > 2000;

Sample Output

Course_Name ▼	Total_Enrolled_Stude
Machine Learning	2043
Cybersecurity	2026

Explanation



student-learning-case-study.students_dataset.students

This table stores each student's data, including their enrolled course and unique identifier.

Query Logic

- COUNT(Student_ID): Counts the number of students enrolled in each course.
- **GROUP BY Course_Name**: Aggregates the counts by course.
- HAVING COUNT(...) > 2000: Filters to include only courses with more than 2000 students enrolled.

Insight

This query helps in:

- Identifying high-demand courses
- Making data-driven decisions for scaling popular content
- Prioritizing resource allocation to meet learner demand in top-enrolled courses

Conclusion

This project demonstrates how structured query language (SQL) can be effectively used to analyze student performance and engagement data within an online learning platform. Through a series of focused queries, we extracted meaningful insights on academic behaviors, course preferences, learning styles, and dropout risks. Key findings include:

- Popular Courses: Courses like Data Science and Machine Learning attract the highest enrollment, indicating strong learner interest in data-driven fields.
- Engagement Trends: Students with high engagement and assignment completion rates tend to perform better in final exams.
- Dropout Risk Indicators: High video watch time combined with low performance metrics often correlates with increased dropout likelihood, especially among female students.
- Demographic Patterns: Preferences in courses and learning styles vary by age and education level, which can help in customizing learning paths.

These insights enable educators and platform developers to optimize learning content, personalize student experiences, and improve retention strategies. SQL serves as a powerful tool in transforming raw educational data into actionable intelligence.