1/25/2025
Prepared by: Fatima Larhrafi

Student Performance Analysis & Automation

SQL & Power BI project

Objective

The objective of this project is to analyse student performance data using **SQL** and **Power BI** to:

- Identify trends in academic performance and attendance.
- Provide actionable insights to improve student outcomes.
- Demonstrate the automation of data updates and reporting.

Key Deliverables

- **Data Analysis**:
  - Cleansed and prepared raw data using SQL.
  - Calculated key metrics such as success rates, attendance rates, and performance trends.
- **Power BI Dashboard**:
  - Created an interactive dashboard to visualize student performance and attendance data. Included KPIs such as:
    - Success Rate by Subject.
    - Attendance Rate by Status.
    - Diversity Metrics
- **Automation suggestions**:
  - Set up a Power Automate flow to monitor changes in the SQL database and send email notifications.
  - Manually refreshed the Power BI dataset to reflect the latest data.

Fatima Larhrafi
FATIMA.LARHRAFI@GMAIL.COM

Tools Used

- **SQL Server**: For data cleansing, preparation, and analysis.
- **Power BI**: For data visualization and dashboard creation.
- **Power Automate**: For automating data updates and notifications.

Key Insights

- **Success Rates**:
  - Cyber Security has the lowest success rate (75%), while Data Science has the highest (90%).
- **Attendance Trends**:
  - 20% of students are frequently absent, impacting their academic performance.
- **Recommendations**:
  - **Implement targeted tutoring programs for Cyber Security.**
  - **Introduce attendance incentives to reduce absenteeism.**

Students Performance Metrics

Average scores by subject, gender, and career aspiration.

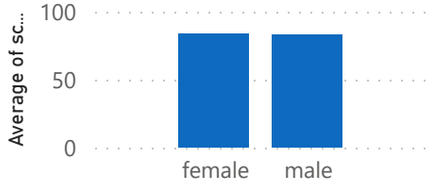
01/10/2023 29/12/2023

subject_name	Average of score
Data Science	88.73
Mathematics for Computing	88.67
Introduction to Databases	86.71
Artificial Intelligence	86.13
Research Methods	83.60
Programming Fundamentals	83.43
Software Engineering	79.40
Advanced Algorithms	79.27
Database Systems	78.53
Cyber Security	74.93
Total	**83.05**

career_aspiration	Average of score
Software Engineer	86.90
Engineer	86.47
Scientist	86.06
Doctor	84.87
Accountant	84.70
Business Owner	83.40
Teacher	82.43
Writer	81.40
Lawyer	81.13
Total	**83.05**

gender	Average of score
female	83.44
male	82.64
Total	**83.05**

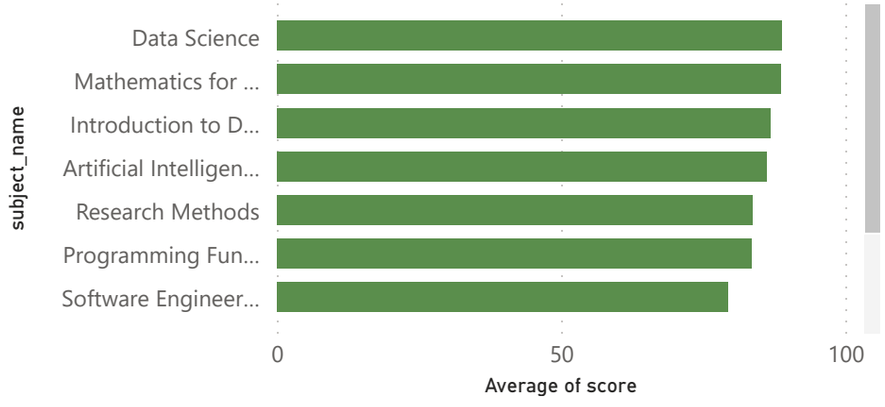
Average of score by gender



Total_Students

30

Average of score by subject_name



Average of score by career_aspiration

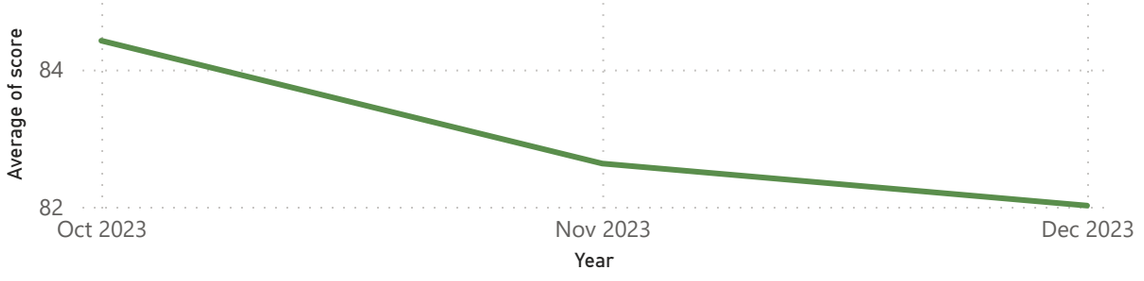
Pass/Fail Rates

By Subject, Students Pass/Fail Status

01/10/2023 29/12/2023

first_name	last_name	subject_name	Academic...	score	Pass_Fail_Status
Eva	Wilson	Programming Fundamentals	2023/24	100.00	Pass
William	King	Programming Fundamentals	2023/24	100.00	Pass
Ava	Wright	Artificial Intelligence	2023/24	99.00	Pass
Frank	Moore	Artificial Intelligence	2023/24	99.00	Pass
Ava	Wright	Mathematics for Computing	2023/24	99.00	Pass
Frank	Moore	Mathematics for Computing	2023/24	99.00	Pass
Ethan	Scott	Artificial Intelligence	2023/24	98.00	Pass
Ivy	Thomas	Artificial Intelligence	2023/24	98.00	Pass
Ethan	Scott	Mathematics for Computing	2023/24	98.00	Pass
Ivy	Thomas	Mathematics for Computing	2023/24	98.00	Pass
Ava	Wright	Introduction to Databases	2023/24	97.00	Pass
Frank	Moore	Introduction to Databases	2023/24	97.00	Pass
Ava	Wright	Programming Fundamentals	2023/24	96.00	Pass
Total					

Average of score by Year and Month



Pass Rate

88.76%

Fail Rate

11.24%

Subject_name

- [ ] Advanced Algorithms
- [ ] Artificial Intelligence
- [ ] Cyber Security
- [ ] Data Science
- [ ] Database Systems
- [ ] Dissertation
- [ ] Introduction to Databases
- [ ] Mathematics for Computing
- [ ] Programming Fundament...
- [ ] Research Methods
- [ ] Software Engineering

Total_Students_Scori...

178

Total_Passes

158

Total_Failes

20

Attendance Rates

01/10/2023 05/11/2023

Absence Rate

20%

Presence Rate

63%

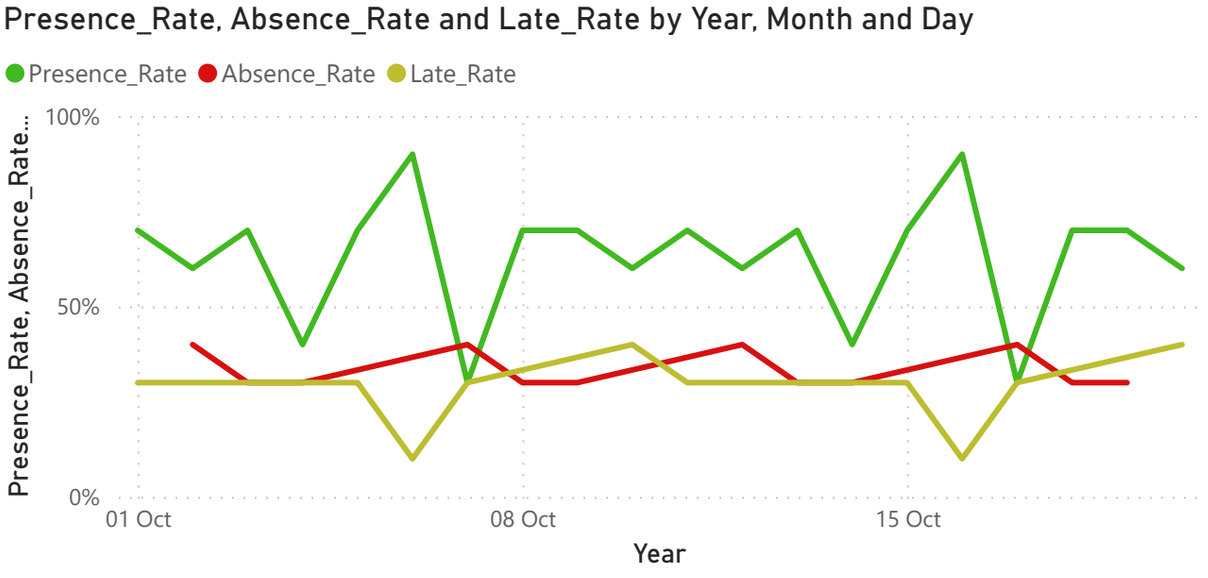
Late Rate

17%

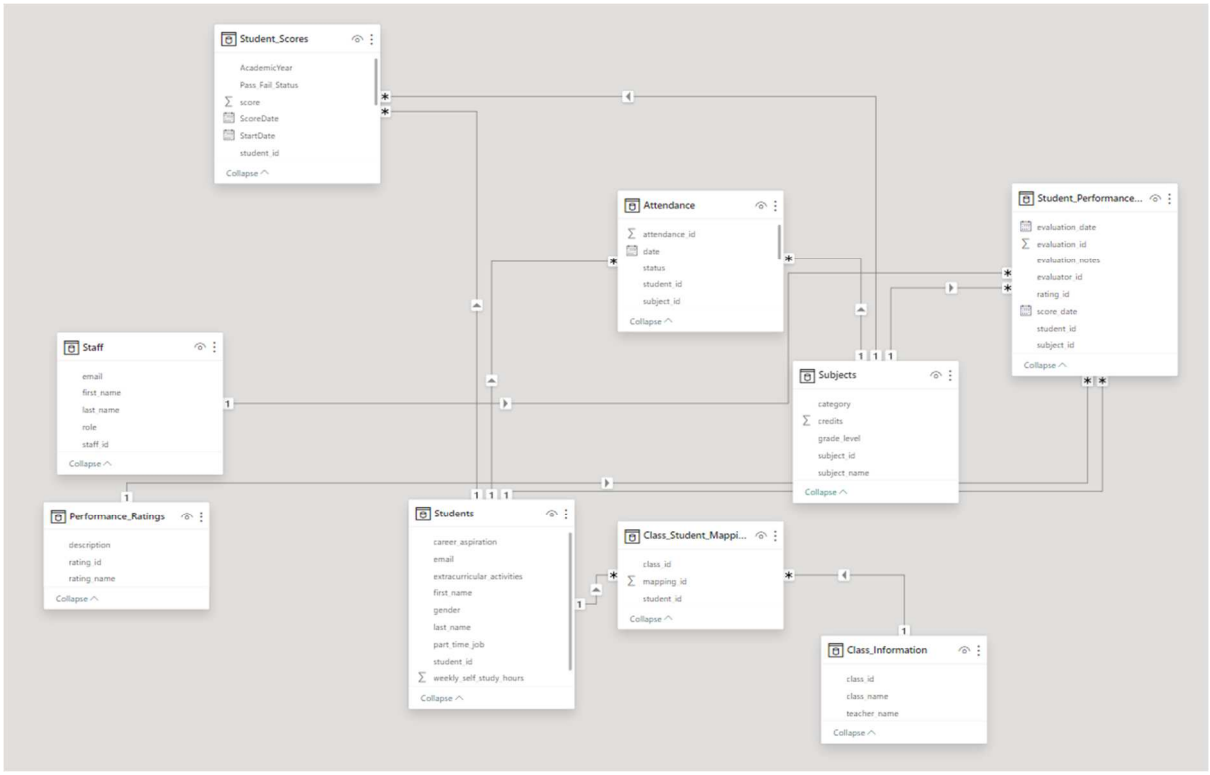
Subject_name

- [ ] Advanced Algorithms
- [ ] Artificial Intelligence
- [ ] Cyber Security
- [ ] Data Science
- [ ] Database Systems
- [ ] Dissertation
- [ ] Introduction to Databases

subject_name	Absence_Rate	Presence_Rate	Late_Rate	Average of score
Cyber Security		90%	10%	74.67
Artificial Intelligence	30%	70%		81.11
Data Science	30%	70%		90.80
Introduction to Databases	30%	70%		90.78
Mathematics for Computing		70%	30%	87.32
Software Engineering		70%	30%	77.50
Programming Fundamentals	40%	60%		88.38
Research Methods		60%	40%	81.67
Database Systems	30%	40%	30%	78.67
Total	**20%**	**63%**	**17%**	**84.20**



Data Model and relationships



SQL Server

1. Student Success Rates by Subject.

Results Messages

	subject_name	number_pass	number_fail	overall_pass_rate	performance_category
1	Mathematics for Computing	19	2	86.36 %	Moderate Performance
2	Introduction to Databases	19	2	86.36 %	Moderate Performance
3	Software Engineering	18	2	90.00 %	High Performance
4	Programming Fundamentals	17	4	77.27 %	Moderate Performance
5	Database Systems	16	4	80.00 %	Moderate Performance
6	Cyber Security	14	1	93.33 %	High Performance
7	Research Methods	14	1	93.33 %	High Performance
8	Advanced Algorithms	13	2	86.67 %	Moderate Performance
9	Artificial Intelligence	13	2	86.67 %	Moderate Performance

```sql
                        /******* 1.Student Success Rates by Subject ********/
--- cte_count_pass: Counts the number of students who passed (score >= 70) for each subject
With cte_count_pass as (
    Select
          subject_id
         ,COUNT(*) as number_pass
    From [dbo].[Student_Scores]
    Where score >= 70
    Group by subject_id
),
--- cte_count_fail: Counts the number of students who failed (score < 70) for each subject
cte_count_fail as (
    Select
          subject_id
         ,COUNT(*) as number_fail
    From [dbo].[Student_Scores]
    Where score < 70
    Group by subject_id

),
--- cte_count_all: Counts the total number of students for each subject
cte_count_all as (
    Select
         subject_id
         ,COUNT(*) as all_count
    From [dbo].[Student_Scores]
    Group by subject_id
)
--- Final Calculation and formatting
Select
       s.subject_name
      ,ss.number_pass
      ,cf.number_fail
     ,Format(ROUND(100.0 * ss.number_pass / cc.all_count, 2) , '0.00') +' %' as overall_pass_rate
     , CASE
        WHEN ROUND(100.0 * ss.number_pass / cc.all_count, 2) >= 90 THEN 'High Performance'
        WHEN ROUND(100.0 * ss.number_pass / cc.all_count, 2) BETWEEN 70 AND 89 THEN 'Moderate Performance'
        ELSE 'Low Performance'
    END AS performance_category
From cte_count_pass ss
Left join cte_count_all cc
On ss.subject_id = cc.subject_id
Left join cte_count_fail cf
On cf.subject_id = cc.subject_id
Left join [dbo].[Subjects] s
On s.subject_id = cf.subject_id
Where s.subject_name is not null
Order by  ss.number_pass desc
         ,cf.number_fail

```

2. Underperforming students (Score <70)

Results | Messages

	first_name	last_name	subject_name	score
1	Ava	Wright	Advanced Algorithms	64
2	Bob	Brown	Mathematics for Computing	65
3	Bob	Brown	Database Systems	68
4	Ethan	Scott	Software Engineering	67
5	Ethan	Scott	Programming Fundamentals	69
6	Eva	Wilson	Introduction to Databases	67
7	Frank	Moore	Advanced Algorithms	64
8	Grace	Taylor	Database Systems	63
9	Henry	Anderson	Programming Fundamentals	68
10	Isabella	Hill	Programming Fundamentals	68
11	Ivy	Thomas	Software Engineering	67
12	Ivy	Thomas	Programming Fundamentals	69
13	James	Hall	Cyber Security	63
14	James	Hall	Mathematics for Computing	65
15	James	Hall	Research Methods	66
16	James	Hall	Database Systems	68
17	Mason	Lopez	Database Systems	63
18	Olivia	Young	Artificial Intelligence	65
19	Sophia	Walker	Artificial Intelligence	65
20	William	King	Introduction to Databases	67

```
1
2                       /********************  5. Gap Analysis  *********************************/
3
4   ---- Identify underperforming students (score < 70)
5   Select
6              s.first_name
7             ,s.last_name
8             ,sb.subject_name
9             ,ss.score
10
11  From       [dbo].[Student_Scores] ss
12  Left join  [dbo].[Students] s
13  On         s.student_id = ss.student_id
14  Left join  [dbo].[Subjects] sb
15  On         ss.subject_id = sb.subject_id
16
17  Where      ss.score <  70
18  Order by   s.first_name
19            ,s.last_name
20            ,ss.score asc
21
```

3. Retention Rate

Results | Messages

	subject_name	number_of_students	retained_students	retention rates
1	Advanced Algorithms	15	15	100.00 %
2	Artificial Intelligence	15	15	100.00 %
3	Cyber Security	15	15	100.00 %
4	Data Science	15	15	100.00 %
5	Database Systems	20	20	100.00 %
6	Introduction to Dat...	22	21	95.45 %
7	Mathematics for Co...	22	21	95.45 %
8	Programming Fund...	22	21	95.45 %
9	Research Methods	15	15	100.00 %
10	Software Engineering	20	20	100.00 %

```sql

                    /******************************** 3. Retention Analysis ************************************/

-- Analyze retention rates to see how many students are retained in each subject
-- Retention rate is a key metric in education that measures the percentage of students who continue their studies in a particular subject or program.

-- CTE to count retained students (students with non-null scores)
With cte_retained_students as (
Select
          subject_id
          ,COUNT(*)  as 'retained_students'
          From   [dbo].[Student_Scores]
          where score IS NOT NULL
          Group by subject_id
),
-- CTE to count total students per subject
cte_all_students_count as (
Select
          subject_id
          ,COUNT(*) as 'number_of_students'
          From   [dbo].[Student_Scores]
          Group by subject_id
)
-- Main query to calculate retention rates
Select
          ss.subject_name
          ,s.number_of_students
          ,r.retained_students
          , CASE    -- If subject has no students avoid division by zero
            When s.number_of_students = 0 then '0.00 %'
            Else FORMAT(ROUND(100.0 * r.retained_students / s.number_of_students , 2) , '0.00') + ' %'
            End  as 'retention rates'

From      [dbo].[Subjects] ss
Left join cte_all_students_count s
On        s.subject_id = ss.subject_id
Left join cte_retained_students r
On        r.subject_id = s.subject_id
Where     s.number_of_students IS NOT NULL  -- Ensure subjects with no students are excluded
Order by  ss.subject_name
          ,[retention rates]


```