REPORT ON STUDENT PERFORMANCE TRACKING SYSTEM

PRESENTED BY AKINDE, ADEDOYIN O

INTRODUCTION

This project contains data presentation insight from a complete dataset located on five different tables.

The title of this project is Students Performance Tracking System. This data sets contains different pieces of information related to students' information, subjects, terms, scores and classes.
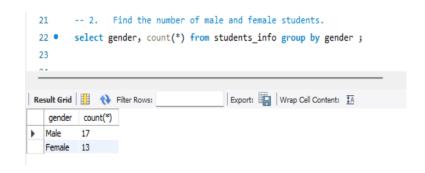
The problem statement of this project states that;

The school management wants to analyse students' academic performance across subjects, terms, and classes. They need insights to identify top students, subjects with poor performance, and track progress over time.
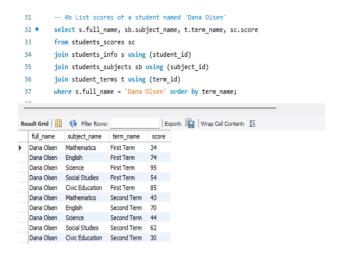
All data were imported into the SQL tool using MySQL and they were queried using different codes to obtain desired results. It was observed from the queries that the general performance of the students increased across terms, and students performed better in civic education than in mathematics based on their average scores per subjects.

There are fifteen (15) questions answered in this project and the queries written for them are attached below;

1)

```
17      -- List all students with their classes
18 •    select student_id, Full_name, class_name from students_info
19      join students_classes using (class_id);
```

student_id	Full_name	class_name
1	Brent Luna	JSS1
2	James Pena	JSS1
3	Cassandra Welch	JSS1
4	Amy Moon	JSS3
5	John Powell	JSS3
6	Sandra Adams	JSS1
7	Jesse Macdonald	JSS1
8	Brittany Mcbride	JSS1
9	Jose Garcia	JSS3
10	Paul Schultz	JSS3

2)

```
21      -- 2.  Find the number of male and female students.
22 •    select gender, count(*) from students_info group by gender ;
23
```

gender	count(*)
Male	17
Female	13

3)

```
25    -- 3.   Show all subjects offered in the school.
26 •  select * from students_subjects;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conter

subject_id	subject_name
1	Mathematics
2	English
3	Science
4	Social Studies
5	Civic Education

4)

```
31    -- 4b List scores of a student named 'Dana Olsen'
32 •  select s.full_name, sb.subject_name, t.term_name, sc.score
33    from students_scores sc
34    join students_info s using (student_id)
35    join students_subjects sb using (subject_id)
36    join student_terms t using (term_id)
37    where s.full_name = 'Dana Olsen' order by term_name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

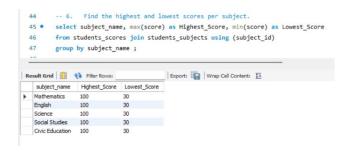| full_name	subject_name	term_name	score
Dana Olsen	Mathematics	First Term	34
Dana Olsen	English	First Term	74
Dana Olsen	Science	First Term	95
Dana Olsen	Social Studies	First Term	54
Dana Olsen	Civic Education	First Term	85
Dana Olsen	Mathematics	Second Term	43
Dana Olsen	English	Second Term	70
Dana Olsen	Science	Second Term	44
Dana Olsen	Social Studies	Second Term	62
Dana Olsen	Civic Education	Second Term	30

5)

```
39    -- 5.   Find the average score per subject.
40 •  select subject_name, avg(score) as AVG_Score from students_scores
41    join students_subjects using (subject_id)
42    group by subject_name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

subject_name	AVG_Score
Mathematics	64.0111
English	64.0889
Science	64.8111
Social Studies	66.0778
Civic Education	67.7556

6)

```
44    -- 6.   Find the highest and lowest scores per subject.
45 •  select subject_name, max(score) as Highest_Score, min(score) as Lowest_Score
46    from students_scores join students_subjects using (subject_id)
47    group by subject_name ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

subject_name	Highest_Score	Lowest_Score
Mathematics	100	30
English	100	30
Science	100	30
Social Studies	100	30
Civic Education	100	30

7)

```sql
50     -- 7.  List top 3 students per subject. (based on total scores)
51 • ⊖ select * from (select full_name, subject_name, sum(score) as Total_score,
52     dense_rank() over (partition by subject_name order by sum(score) desc) as Top_3
53     from students_scores
54     join students_info using(student_id)
```

```sql
55         join students_subjects using (subject_id)
56         group by full_name, subject_name)
57         ranked_scores
58         where top_3 <=3;
59
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

full_name	subject_name	Total_score	Top_3
Anna Hall	Civic Education	256	1
Shannon Owen	Civic Education	249	2
Benjamin Lopez	Civic Education	249	2
Paul Schultz	Civic Education	241	3
Rachel Gregory	English	242	1
Theresa Douglas	English	237	2
Dana Olsen	English	229	3
Theresa Douglas	Mathematics	264	1
Rachel Gregory	Mathematics	261	2
Cassandra Welch	Mathematics	261	2
Paul Schultz	Mathematics	248	3

8)

```sql
62     -- 8.  Rank students in a class based on average score.
63 • ⊖ select * from (select s.full_name, c.class_name, round(avg(sc.score),2)as Avg_Score,
64     rank() over (partition by c.class_name order by Avg(sc.score) desc) as ranks
65     from  students_scores sc
66     join students_info s on sc.student_id = s.student_id
67     join students_classes c on s.class_id = c.class_id
68     group by s.full_name, c.class_name) as Ranked_students;
69
70
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

full_name	class_name	Avg_Score	ranks
Theresa Douglas	JSS1	78.67	1
Cassandra Welch	JSS1	72.13	2
Brittany Mcbride	JSS1	71.07	3
Sandra Adams	JSS1	66.87	4
Cory Berg	JSS1	66.67	5
Brent Luna	JSS1	64.67	6
Benjamin Lopez	JSS1	64.20	7
Jesse Macdonald	JSS1	63.87	8
Shannon Owen	JSS1	63.53	9
Mrs. Melissa Lopez MD	JSS1	63.20	10
Seth Gonzalez	JSS1	62.60	11
James Pena	JSS1	61.80	12

9)

```sql
72     -- 9.  Find students who failed any subject (score < 50).
73 •   select distinct  full_name, subject_name, score, term_name from students_scores
74     join students_info using (student_id)
75     join students_subjects using (subject_id)
76     join student_terms using (term_id)
77     where score <50 order by full_name;
78
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

full_name	subject_name	score	term_name
Amy Moon	Civic Education	31	Third Term
Amy Moon	Science	38	Second Term
Amy Moon	English	49	Second Term
Amy Moon	English	44	First Term
Amy Moon	Mathematics	36	Third Term
Amy Moon	Mathematics	41	Second Term
Anna Hall	Science	36	Second Term
Anna Hall	Social Studies	45	First Term
Anna Hall	Social Studies	47	Third Term
Anna Hall	Mathematics	43	Second Term
Anna Hall	Mathematics	30	First Term
Belinda H	Science	46	Third Term

10)

```
86      -- 10.  Show the number of students per class.
87 •    select class_name, count(full_name) from students_classes
88      join students_info using (class_id) group by class_name order by class_name;
89
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

class_name	count(full_name)
JSS1	14
JSS2	5
JSS3	11

11)

```
95      -- 11.  Create a view to show student performance by term--
96 •    Create view Student_Term_Scores as
97      select s.full_name, t.term_name, Avg(sc.score) as Avg_Score
98      from students_scores sc
99      join students_info s using (student_id)
100     join  student_terms t using (term_id)
101     group by s.full_name, t.term_name;
102 •   select * from Student_Term_Scores ;
103
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

full_name	term_name	Avg_Score
Brent Luna	First Term	73.8000
Brent Luna	Second Term	65.8000
Brent Luna	Third Term	54.4000
James Pena	First Term	68.8000
James Pena	Second Term	63.4000
James Pena	Third Term	53.2000
Cassandra Welch	First Term	68.8000
Cassandra Welch	Second Term	73.8000
Cassandra Welch	Third Term	73.8000
Amy Moon	First Term	72.2000
Amy Moon	Second Term	63.4000
Amy Moon	Third Term	58.6000

12)

```
104     -- 12.  Write a subquery to find students who improved across terms.
105 •   select * from students_scores, Students_info, Student_terms;
106 •   select first_term.full_name, first_term.avg_score as First_Term_Score, third_term.avg_score as Third_Term_Score
107     from
108     (select s.full_name, round(avg(sc.score),2) as Avg_Score from students_scores sc
109     join students_info s using (student_id)
110     where sc.term_id = 1
111     group by s.full_name) as First_Term
112
```
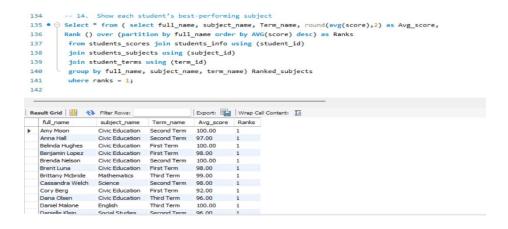
```
113     Join ( select s.full_name, round(avg(sc.score),2) as Avg_score from students_scores sc
114     join students_info s using (student_id)
115     where sc.term_id = 3
116     group by s.full_name) as Third_term
117     On first_term.full_name = third_term.full_name
118     where Third_term.avg_score > first_term.avg_score
119     order by Third_term_score desc;
120
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

full_name	First_Term_Score	Third_Term_Score
Daniel Malone	45.60	82.40
Brittany Mcbride	72.20	82.00
Brenda Nelson	59.40	80.40
Jesse Macdonald	47.00	79.00
Paul Schultz	69.00	76.20
Dana Olsen	68.40	76.00
Shannon Owen	51.00	75.40
Cassandra Welch	68.80	73.80
Cory Berg	67.60	72.00
Anna Hall	62.20	71.60
John Powell	56.80	71.20
Mrs. Melissa Lon	57.80	69.40

13)

```
-- 13.  Use CASE to grade students: A (≥ 70), B (60-69), C (50-59), F (< 50)
select full_name, subject_name, score, term_name,
Case
When score >= 70 Then 'A'
When score >= 60 Then 'B'
When score >= 50 Then 'C'
Else 'F'
End as Grade
from students_scores
```

```
127          Else 'F'
128          End as Grade
129          from students_scores
130          join students_info using (student_id)
131          join students_subjects using (subject_id)
132          join student_terms using (term_id);
133
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

full_name	subject_name	score	term_name	Grade
Brent Luna	Civic Education	50	Third Term	C
Brent Luna	Civic Education	61	Second Term	B
Brent Luna	Civic Education	98	First Term	A
Brent Luna	Social Studies	51	Third Term	C
Brent Luna	Social Studies	39	Second Term	F
Brent Luna	Social Studies	64	First Term	B
Brent Luna	Science	56	Third Term	C
Brent Luna	Science	75	Second Term	A
Brent Luna	Science	77	First Term	A
Brent Luna	English	50	Third Term	C
Brent Luna	English	76	Second Term	A
Brent Luna	English	88	First Term	A

14)

```
134      -- 14.  Show each student's best-performing subject
135 ● ⊖ Select * from ( select full_name, subject_name, Term_name, round(avg(score),2) as Avg_score,
136          Rank () over (partition by full_name order by AVG(score) desc) as Ranks
137          from students_scores join students_info using (student_id)
138          join students_subjects using (subject_id)
139          join student_terms using (term_id)
140          group by full_name, subject_name, term_name) Ranked_subjects
141      where ranks = 1;
142
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

full_name	subject_name	Term_name	Avg_score	Ranks
Amy Moon	Civic Education	Second Term	100.00	1
Anna Hall	Civic Education	Second Term	97.00	1
Belinda Hughes	Civic Education	First Term	100.00	1
Benjamin Lopez	Civic Education	First Term	98.00	1
Brenda Nelson	Civic Education	Second Term	100.00	1
Brent Luna	Civic Education	First Term	98.00	1
Brittany Mcbride	Mathematics	Third Term	99.00	1
Cassandra Welch	Science	Second Term	98.00	1
Cory Berg	Civic Education	First Term	92.00	1
Dana Olsen	Civic Education	Third Term	96.00	1
Daniel Malone	English	Third Term	100.00	1
Danielle Klein	Social Studies	Second Term	96.00	1

15)

```sql
-- 15.  Generate a report showing student names, class, subject, term, score, and grade
Select full_name, subject_name, term_name, score, class_name,
Case
when sc.score >= 70 Then 'A'
When sc.score >= 60 Then 'B'
When sc.score >= 50 Then 'C'
When sc.score >= 40 Then 'D'
ELse 'F'
End as Grade
```

```
152        End as Grade
153        from Students_scores sc
154        join students_info using (student_id)
155        join students_subjects using (subject_id)
156        join student_terms using (term_id)
157        join students_classes using (class_id);
```

Result Grid | 🔲 | ⟷ | Filter Rows: [   ] | Export: 🔲 | Wrap Cell Content: 🛦

full_name	subject_name	term_name	score	class_name	Grade
Theresa Douglas	Mathematics	Third Term	84	JSS1	A
Theresa Douglas	Mathematics	Second Term	94	JSS1	A
Theresa Douglas	Mathematics	First Term	86	JSS1	A
Anna Hall	Mathematics	Third Term	84	JSS2	A
Anna Hall	Mathematics	Second Term	43	JSS2	D
Anna Hall	Mathematics	First Term	30	JSS2	F
Mrs. Melissa Lopez MD	Mathematics	Third Term	33	JSS1	F
Mrs. Melissa Lopez MD	Mathematics	Second Term	48	JSS1	D
Mrs. Melissa Lopez MD	Mathematics	First Term	55	JSS1	C
Rachel Gregory	Mathematics	Third Term	86	JSS3	A
Rachel Gregory	Mathematics	Second Term	86	JSS3	A
Rachel Gregory	Mathematics	First Term	89	JSS3	A

Finally, all the queries were visualized on Power Bi.

RECOMMENDATIONS

The performance of the students varies across all subjects, but it was found out that the students perform better in civic education and other subjects more than mathematics. This could mean that the students have more interest in these other subjects than in mathematics. With mathematics being an important subject, it would be best if the school looks into the mathematic department of the school and deduce what challenges the teachers are facing.

The school could either change the mode of teaching, and learning for teachers and students in other to improve their performances.

In the case where the school lacks competent hands, more qualified teachers who understands how to deliver the lectures that would meet the need of the students should be employed.

Finally, all students should be encouraged to put in their best effort in understanding all the subjects taught and there should be opportunities for students to ask questions wherever any subjects seem difficult to understand.