Analysing Students' Mental Health in SQL

Creating database:

```
create database mental_health_support_analytics;
use mental_health_support_analytics;
```

Creating Tables:

```
Create table Students(
    student_id int primary key auto_increment,
    name varchar(255);
age int,
gender enum('Male', 'Female', 'Other'),
grade_level varchar(50),
school_id int
);

CREATE TABLE Mental_Health_Survey_Responses (
    response_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT,
    survey_date DATE,
    question_id INT,
    response VARCHAR(255),
    FOREIGN KEY (student_id) REFERENCES
Students(student_id),
    FOREIGN KEY (question_id) REFERENCES
Survey_Questions(question_id)
);

CREATE TABLE Schools (
    school_id INT PRIMARY KEY AUTO_INCREMENT,
    school_name VARCHAR(255),
    location VARCHAR(255)
);
```

```sql
CREATE TABLE Mental_Health_Assessments (
    assessment_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT,
    assessment_date DATE,
    assessment_type VARCHAR(255),
    assessor VARCHAR(255),
    FOREIGN KEY (student_id) REFERENCES
Students(student_id)
);

CREATE TABLE Assessment_Questions (
    question_id INT PRIMARY KEY AUTO_INCREMENT,
    assessment_id INT,
    question_text TEXT,
    response TEXT,
    FOREIGN KEY (assessment_id) REFERENCES
Mental_Health_Assessments(assessment_id)
);

CREATE TABLE Interventions (
    intervention_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT,
    intervention_date DATE,
    intervention_type VARCHAR(255),
    provider VARCHAR(255),
    FOREIGN KEY (student_id) REFERENCES
Students(student_id)
);

CREATE TABLE Intervention_Outcomes (
    outcome_id INT PRIMARY KEY AUTO_INCREMENT,
    intervention_id INT,
    outcome_measure VARCHAR(255),
    outcome_value VARCHAR(255),
    outcome_date DATE,
    FOREIGN KEY (intervention_id) REFERENCES
Interventions(intervention_id)
);
```

```
CREATE TABLE Support_Services (
    service_id INT PRIMARY KEY AUTO_INCREMENT,
    student_id INT,
    service_type VARCHAR(255),
    service_provider VARCHAR(255),
    FOREIGN KEY (student_id) REFERENCES
Students(student_id)
);
```

Inserting values:

```
-- Insert values into the Students table
INSERT INTO Students (name, age, gender, grade_level,
school_id)
VALUES
    ('John Doe', 17, 'Male', '12th Grade', 1),
    ('Jane Smith', 16, 'Female', '11th Grade', 2),
    ('Michael Johnson', 15, 'Male', '10th Grade', 1);

-- Insert values into the Mental_Health_Assessments
table
INSERT INTO Mental_Health_Assessments (student_id,
assessment_date, assessment_type, assessor)
VALUES
    (1, '2024-02-14', 'Questionnaire',
'Psychologist'),
    (2, '2024-02-15', 'Interview', 'Counselor'),
    (3, '2024-02-16', 'Questionnaire', 'Therapist');

-- Insert values into the Assessment_Questions table
INSERT INTO Assessment_Questions (assessment_id,
question_text, response)
VALUES
    (1, 'How often do you feel anxious?', 'Often'),
    (1, 'Do you have trouble sleeping?', 'Yes'),
    (2, 'Describe your mood over the past week.', 'I
have been feeling down lately.'),
```

```
    (3, 'Have you experienced any traumatic events
recently?', 'Yes');

-- Insert values into the Interventions table
INSERT INTO Interventions (student_id,
intervention_date, intervention_type, provider)
VALUES
    (1, '2024-02-16', 'Counseling', 'School
Counselor'),
    (2, '2024-02-17', 'Therapy', 'Licensed
Therapist'),
    (3, '2024-02-18', 'Counseling', 'School
Counselor');

-- Insert values into the Intervention_Outcomes table
INSERT INTO Intervention_Outcomes (intervention_id,
outcome_measure, outcome_value, outcome_date)
VALUES
    (1, 'Reduction in anxiety symptoms', '25%
improvement', '2024-03-01'),
    (2, 'Improvement in mood', 'Significant
improvement', '2024-03-02'),
    (3, 'Increase in coping skills', 'Developed new
coping strategies', '2024-03-03');

-- Insert values into the Support_Services table
INSERT INTO Support_Services (student_id,
service_type, service_provider)
VALUES
    (1, 'Support Group', 'Peer Support Group'),
    (2, 'Peer Counseling', 'Senior Peer Counselor'),
    (3, 'Support Group', 'Mental Health
Organization');

-- Insert values into the Schools table
INSERT INTO Schools (school_name, location)
VALUES
    ('Central High School', 'City A'),
    ('Westside High School', 'City B');
```

```sql
-- Insert values into the
Mental_Health_Survey_Responses table
INSERT INTO Mental_Health_Survey_Responses
(student_id, survey_date, question_id, response)
VALUES
    (1, '2024-02-14', 1, 'Often'),
    (1, '2024-02-14', 2, 'Yes'),
    (2, '2024-02-15', 1, 'Rarely'),
    (2, '2024-02-15', 2, 'No'),
    (3, '2024-02-16', 1, 'Sometimes'),
    (3, '2024-02-16', 2, 'Yes');

-- Insert values into the Intervention_Outcomes table
for Mental Health Survey Outcomes
INSERT INTO Intervention_Outcomes (intervention_id,
outcome_measure, outcome_value, outcome_date)
VALUES
    (NULL, 'Survey Outcome 1', 'Outcome Value 1',
'2024-02-20'),
    (NULL, 'Survey Outcome 2', 'Outcome Value 2',
'2024-02-21');

-- Update the mental health survey responses table to
assign intervention_id
UPDATE Mental_Health_Survey_Responses
SET intervention_id = 1
WHERE student_id IN (1, 2);

-- Insert values into the Intervention_Outcomes table
for Mental Health Survey Outcomes
UPDATE Mental_Health_Survey_Responses
SET intervention_id = 2
WHERE student_id = 3;

-- Insert values into the Support_Services table
INSERT INTO Support_Services (student_id,
service_type, service_provider)
VALUES
```

```
    (1, 'Support Group', 'Peer Support Group'),
    (2, 'Peer Counseling', 'Senior Peer Counselor'),
    (3, 'Support Group', 'Mental Health
Organization');
```

Queries:

1. Retrieve mental health assessments along with student information.

```
select A.*, s.name, s.age, s.gender, s.grade_level,
sc.school_name from mental_health_assessments A join
Students S on A.student_id = s.student_id join schools
sc on s.school_id = sc.school_id;
```

assessment_id	student_id	assessment_date	assessment_type	assessor	name	age	gender	grade_level	school_name
1	1	2024-02-14	Questionnaire	Psychologist	John Doe	17	male	12th Grade	Central High School
2	2	2024-02-15	Interview	Counselor	Jane Smith	16	female	11th Grade	Westside High School
3	3	2024-02-16	Questionnaire	Therapist	Michael Johnson	15	male	10th Grade	Central High School

3 rows in set (0.007 sec)

2. Retrieve intervention outcomes along with intervention details.

```
select O.*, I.intervention_date, I.intervention_type,
I.provider, S.name as Student_name from
intervention_outcomes O join interventions I on
O.intervention_id = I.intervention_id join students S
on I.student_id = S.student_id;
```

```
MariaDB [mental_health_support_analytics]> select O.*, I.intervention_date, I.intervention_type, I.provider, S.name as Student_name from intervention_outcomes O
    -> join interventions I on O.intervention_id = I.intervention_id
    -> join students S on I.student_id = S.student_id;
```

outcome_id	intervention_id	outcome_measure	outcome_value	outcome_date	intervention_date	intervention_type	provider	Student_name
1	1	Reduction in anxiety symptoms	25% improvement	2024-03-01	2024-02-16	Counseling	School Counselor	John Doe
2	2	Improvement in mood	Significant improvement	2024-03-02	2024-02-17	Therapy	Licensed Therapist	Jane Smith
3	3	Increase in coping skills	Developed new coping strategies	2024-03-03	2024-02-18	Counseling	School Counselor	Michael Johnson

3 rows in set (0.006 sec)

3. Retrieve support service provided to each student.

```
select ss.*, s.name as student_name from
support_services ss join students s on ss.student_id =
s.student_id;
```

```
+------------+------------+-----------------+-----------------------------+-----------------+
| service_id | student_id | service_type    | service_provider            | student_name    |
+------------+------------+-----------------+-----------------------------+-----------------+
|          1 |          1 | Support Group   | Peer Support Group          | John Doe        |
|          2 |          2 | Peer Counseling | Senior Peer Counselor       | Jane Smith      |
|          3 |          3 | Support Group   | Mental Health Organization | Michael Johnson |
+------------+------------+-----------------+-----------------------------+-----------------+
3 rows in set (0.001 sec)
```

4. Retrieve assessment questions along with assessment details

```
select Q.*, A.assessment_date, A.assessment_type,
A.assessor, S.name as student_name from
assessment_questions Q join mental_health_assessments
A on Q.assessment_id  = A.assessment_id join students
s on A.student_id = S.student_id;
```

```
+-------------+-----------------------------------------+---------------+-----------------------------+-----------------+-----------------+-------------+-----------------+
| question_id | question_text                           | assessment_id | response                    | assessment_date | assessment_type | assessor    | student_name    |
+-------------+-----------------------------------------+---------------+-----------------------------+-----------------+-----------------+-------------+-----------------+
|           1 | How often do you feel anxious?          |             1 | Often                       | 2024-02-14      | Questionnaire   | Psychologist | John Doe        |
|           2 | Do you have trouble sleeping?           |             1 | Yes                         | 2024-02-14      | Questionnaire   | Psychologist | John Doe        |
|           3 | Describe your mood over the past week.  |             2 | I have been feeling down lately. | 2024-02-15      | Interview       | Counselor   | Jane Smith      |
|           4 | Have you experienced any traumatic events recently? |             3 | Yes                         | 2024-02-16      | Questionnaire   | Therapist   | Michael Johnson |
+-------------+-----------------------------------------+---------------+-----------------------------+-----------------+-----------------+-------------+-----------------+
4 rows in set (0.001 sec)

MariaDB [mental_health_support_analytics]>
```

5. Retrieve student names and their corresponding mental health assessment dates [inner join].

```
select students.name,
mental_health_assessments.assessment_date from
students inner join mental_health_assessments on
students.student_id =
mental_health_assessments.student_id;
```

```
+-----------------+-----------------+
| name            | assessment_date |
+-----------------+-----------------+
| John Doe        | 2024-02-14      |
| Jane Smith      | 2024-02-15      |
| Michael Johnson | 2024-02-16      |
+-----------------+-----------------+
3 rows in set (0.001 sec)
```

6. Retrieve student names and the count of their mental health assessment (inner join with group by and count)

```
select students.name,
count(mental_health_assessments.assessment_id) as
```

```
assessment_count from students inner join
mental_health_assessments on students.student_id =
mental_health_assessments.student_id group by
students.name;
```

```
+-----------------+------------------+
| name            | assessment_count |
+-----------------+------------------+
| Jane Smith      |                1 |
| John Doe        |                1 |
| Michael Johnson |                1 |
+-----------------+------------------+
3 rows in set (0.001 sec)
```

7. Retrieve all students and their mental health assessments, including those who haven't had any assessments (left join)

```
select students.name,
mental_health_assessments.assessment_date from
students left join mental_health_assessments on
students.student_id =
mental_health_assessments.student_id;
```

```
+-----------------+-----------------+
| name            | assessment_date |
+-----------------+-----------------+
| John Doe        | 2024-02-14      |
| Jane Smith      | 2024-02-15      |
| Michael Johnson | 2024-02-16      |
+-----------------+-----------------+
3 rows in set (0.001 sec)
```

8. Retrieve student name and the count of their interventions, ordered by the intervention count in descending order (inner join with group by, count, and order by)

```
select s.name, count(i.intervention_id) as
intervention_count from students s inner join
interventions i on s.student_id = i.student_id group
by s.name order by intervention_count DESC;
```

```
+-----------------+--------------------+
| name            | intervention_count |
+-----------------+--------------------+
| John Doe        |                  1 |
| Jane Smith      |                  1 |
| Michael Johnson |                  1 |
+-----------------+--------------------+
3 rows in set (0.002 sec)
```

9. Retrieve the total number of interventions provided to students (count with subquery)

```
select count(*) as total_interventions from
interventions;
```

```
+---------------------+
| total_interventions |
+---------------------+
|                   3 |
+---------------------+
1 row in set (0.009 sec)
```

10. Retrieve the sum of outcome values for each intervention type(sum with group by)

```
select i.intervention_type, sum(ir.outcome_value) as
total_outcome_value from interventions i join
intervention_outcomes ir on i.intervention_id =
ir.intervention_id group by i.intervention_type;
```

```
+-------------------+---------------------+
| intervention_type | total_outcome_value |
+-------------------+---------------------+
| Counseling        |                  25 |
| Therapy           |                   0 |
+-------------------+---------------------+
2 rows in set, 3 warnings (0.005 sec)
```

11. Retrieve student name and the count of support service they have received including students who haven't received any services (right join with coalesce to handle null values).

```
select s.name, coalesce(count(ss.service_id), 0) as
service_count from students s right join
support_services ss on s.student_id = ss.student_id
group by s.name;
```

```
+-----------------+---------------+
| name            | service_count |
+-----------------+---------------+
| Jane Smith      |             1 |
| John Doe        |             1 |
| Michael Johnson |             1 |
+-----------------+---------------+
3 rows in set (0.007 sec)
```

12. Retrieve student names and the count of their mental health assessments where the count is greater than 2 (having clause with group by and count)

```
select s.name, count(m.assessment_id) as
assessment_count from students s inner join
mental_health_assessments m on s.student_id =
m.student_id group by s.name having
count(m.assessment_id) > 2;
```

```
Empty set (0.002 sec)
```

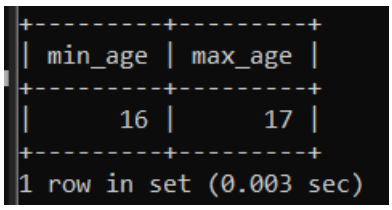
13. Retrieve the names of students who have received interventions from Licensed therapist(subquery with exists and inner join)

```
select name from students where student_id = (select
student_id from interventions i where provider =
'Licensed Therapist');
```

```
+------------+
| name       |
+------------+
| Jane Smith |
+------------+
1 row in set (0.001 sec)
```

14. Retrieve the minimum and maximum ages of students whose names start name with ‘J’ (MIN, MAX, LIKE , SUBSTRING)

```
select min(age) as min_age, max(age) as max_age from
students where name like 'J%';
```



```
+---------+---------+
| min_age | max_age |
+---------+---------+
|      16 |      17 |
+---------+---------+
1 row in set (0.003 sec)
```

15. Creating a trigger to update grade level based on their age updation on the students table.

```
create trigger update_grade_level
after insert on students
for each row
begin
    if new.age >= 18 then
        update students
        set grade_level = 'College'
        where student_id = new.student_id;
    end if;
end;
//
delimiter ;
```

```
MariaDB [mental_health_support_analytics]> delimiter //
MariaDB [mental_health_support_analytics]> create trigger update_grade_level
    -> after insert on students
    -> for each row
    -> begin
    -> if new.age >= 18 then
    -> update students
    -> set grade_level = 'College'
    -> where student_id = new.student_id;
    -> end if;
    -> end;
    -> //
Query OK, 0 rows affected (0.024 sec)

MariaDB [mental_health_support_analytics]> delimiter ;
MariaDB [mental_health_support_analytics]>
```

16. Create a stored procedure which retrieves a summary of mental health support serviced provided to a specific student, including the count of assessments, interventions and support services. (passing student id as argument)

```
delimiter //

 create procedure GetStudentSummary(IN student_id int)
    -> begin
    -> select s.name as student_name,
    -> count(distinct mha.assessment_id) as assessment_count,
    -> count(distinct i.intervention_id) as intervention_count,
    -> count(distinct ss.service_id) as service_count
    -> from students s
    -> left join mental_health_assessments mha on s.student_id =
mha.student_id
    -> left join interventions i on s.student_id = i.student_id
    -> left join support_services ss on s.student_id = ss.student_id
    -> where s.student_id  = student_id;
    -> end //

delimiter ;
```

```
MariaDB [mental_health_support_analytics]> create procedure GetStudentSummary(IN student_id int)
    -> begin
    -> select s.name as student_name,
    -> count(distinct mha.assessment_id) as assessment_count,
    -> count(distinct i.intervention_id) as intervention_count,
    -> count(distinct ss.service_id) as service_count
    -> from students s
    -> left join mental_health_assessments mha on s.student_id = mha.student_id
    -> left join interventions i on s.student_id = i.student_id
    -> left join support_services ss on s.student_id = ss.student_id
    -> where s.student_id  = student_id;
    -> end //
Query OK, 0 rows affected (0.055 sec)

MariaDB [mental_health_support_analytics]> delimiter ;
```

```
MariaDB [mental_health_support_analytics]> call GetStudentSummary(1);
+--------------+------------------+--------------------+---------------+
| student_name | assessment_count | intervention_count | service_count |
+--------------+------------------+--------------------+---------------+
| John Doe     |                1 |                  1 |             1 |
+--------------+------------------+--------------------+---------------+
1 row in set (0.013 sec)

Query OK, 0 rows affected (0.032 sec)

MariaDB [mental_health_support_analytics]>
```

Here.