Part 1:

1. Consider the scenario when school management wants to evaluate which employee will get the next promotion based their time in school, their designation along with their performance (which they evaluate based on feedback from student or their parents.)

    1.1. Identify the age of employee when they joined the school within Employee table. [Hint: Use Employee_Birthdate and Employee_since columns].

    Calculate the total employees belonging to each age/age-group.

```
SELECT Employee_since - YEAR(Employee_Birthdate) AS
JoiningAge,

COUNT(*) AS Total_Employees

FROM employee

GROUP BY Employee_since - YEAR(Employee_Birthdate);
```

JoiningAge	Total_Employees
28	4
29	1
42	1
22	4
23	1

24	3
20	1
36	1
15	1
10	1
13	1
5	1
31	1

1.2. Calculate the min and max of age of employees.

```
SELECT MIN(2023 - YEAR(Employee_Birthdate)) AS MinAge,
MAX(2023 - YEAR(Employee_Birthdate)) AS MaxAge
FROM employee;
```

MinAge	MaxAge
23	58

1.3 Identify the time spent by employees in school grouped by

1.3.1 Their designation

```
SELECT Employee_designation,
SUM(2023 - Employee_since) AS TotalTimeSpent
FROM employee
GROUP BY Employee_designation;
```

Employee_designation	TotalTimeSpent
Professor	103
Principal	11
VicePrincipal	13
Peon	24
Clerk	5
Assistant	23

1.4. Calculate following feedback statistics:

1.4.1. Total number of feedbacks for an employee on employee id

```
 SELECT Employee_Id, COUNT(*) AS Total_Feedbacks
FROM ratings
GROUP BY Employee_Id;
```

Employee_Id	Total_Feedbacks
2	4
9	3
12	3
19	3
21	4

1.4.2. Average rating of an employee having at least 3 feedbacks.

```
SELECT employee_id, COUNT(*) AS Total_feedbacks,
AVG(Rating) AS Avg_Ratings
FROM ratings
GROUP BY employee_id
HAVING COUNT(*) >= 3;
```

employee_id	Total_feedbacks	Avg_Ratings
2	4	4.0000
9	3	5.0000
12	3	3.6667

19	3	4.3333
21	4	3.7500

1. School wants to determine if they have to divide a class to create more sections or they have to merge multiple sections into one based on their strength. (If there are too many students in class then it would create chaos in class while if there are less students then it will be wastage of resources for school).

    1.1. Identify the total number of students by

        1.1.1. Class_Id

```
SELECT Class_Id, COUNT(*) AS Total_Students

FROM student

GROUP BY Class_Id;
```

Class_Id	Total_Students
8	10
9	10
10	10

        1.1.2. Class_Id and Student_Class

```
SELECT Class_Id, Student_Class,

COUNT(*) AS Total_Students
```

```
FROM student

GROUP BY Class_Id, Student_Class;
```

Class_Id	Student_Class	Total_Students
8	A	5
8	B	5
9	A	5
9	B	5
10	A	5
10	B	5

2. School provides bus services to difference locations of Delhi-NCR for employees and students. School management wants to optimize the bus services in a way that we get no seat left in buses and everybody gets picked up. Provide following stats for internal evaluation of school:

    2.1. Total number of employees by Employee designation

```
SELECT Employee_designation, COUNT(*) Total_Employees

FROM employee

GROUP BY Employee_designation;
```

Employee_designation	Total_Employees
Professor	14
Principal	1
VicePrincipal	1
Peon	2
Clerk	1
Assistant	2

2.2. Total number of students by each city

```
SELECT Student_City, COUNT(*) Total_Students

FROM student

GROUP BY Student_City;
```

Student_City	Total_Students
Gurgaon	10
Delhi	13

Noida	7

Consider the scenario that school is hosting an annual event where they will be distributing the prizes to students for various things. For instance, best performer in studies or sports or arts or some extra - curricular activities. Now answer the following:

1. Class 8A student from Gurgaon has been a stellar performer whole year. Get the name of the student.

```
SELECT Student_Name

FROM student

WHERE Class_Id = 8 AND Student_Class = 'A' AND Student_City =
'Gurgaon';
```

Student_Name
Abhimanyu

2. Class 9A and 10B students from Delhi are fantastic musicians and just gave an outstanding performance in a national level event. Get the name of the students. (Try to solve this query using only AND, OR operation as well. Try to explain where it could fail by applying IN operation)

```
SELECT Student_Name

FROM student
```

```
WHERE Class_Id IN (9,10) AND Student_Class IN ('A', 'B') AND
Student_City = 'Delhi';
```

OR

```
SELECT Student_Name

FROM student

WHERE (Class_Id = 9 AND Student_Class = 'A' AND Student_City
= 'Delhi')

OR (Class_Id = 10 AND Student_Class = 'B' AND Student_City =
'Delhi');
```

Student_Name
Abhilasha
Anushka
Sanyam
Sanya

3. Professor from gurgaon who are with us since 2006 and 2020 has been a fantastic duo to carry out the science projects on state level with school students and got prize from state CM. Get the name of professors.

```
SELECT Employee_Name

FROM employee

WHERE Employee_designation = 'Professor'

AND Employee_City = 'Gurgaon'

AND Employee_since IN (2006, 2020);
```

Employee_Name
Priya
Rajesh

1. School management wants to identify all the professor names and their Ids that are not mapped with any courses as of now. Also, identify professor info who are currently mapped to a course. [Hint: Use DISTINCT with JOINs and IS NULL in WHERE for matching up null values]

```
SELECT DISTINCT e.Employee_Id, e.Employee_Name

FROM employee AS e

JOIN course AS c

ON e.Employee_Id = c.Professor_Id;
```

Employee_Id	Employee_Name
2	Komal
9	Gyanesh
12	Javed
13	Suman

14	Rajesh
19	Amrinder
21	Uday

```
SELECT e.Employee_Id, e.Employee_Name

FROM employee AS e

LEFT JOIN course AS c

ON e.Employee_Id = c.Professor_Id

WHERE c.Professor_Id IS NULL;
```

Employee_Id	Employee_Name
1	**Priya**
3	**Sharda**
4	**Lokesh**
5	**Anagh**
6	**Pravesh**
7	**Samyara**

8	Somesh
10	Mahesh
11	Saniya
15	Abhishek
16	Muskan
17	Shashank
18	Nikita
20	Sukhvinder

2. School management wants to identify the class teachers to give them proper resources.

   a. Get the total professors that are currently a class teacher.

```
SELECT COUNT(*) AS TotalClassTeachers

FROM employee AS e

JOIN class AS c

ON e.Employee_Id = c.ClassTeacher;
```

TotalClassTeachers
3

b. Get the id and name of professors who are currently a class teacher.

```
SELECT e.Employee_Name, e.Employee_Id
FROM employee AS e
JOIN class AS c
ON e.Employee_Id = c.ClassTeacher;
```

Employee_Name	Employee_Id
Javed	12
Amrinder	19
Uday	21

c. Find the total Assignments and paper by each class teacher in a class.

```
SELECT c.ClassTeacher,
SUM(co.Course_Assignments),
SUM(co.Course_Paper)
FROM
course as co
JOIN class AS c
ON c.Class_Id = co.Class_Id
GROUP BY c.ClassTeacher;
```

ClassTeacher	Total_Assignments	Total_Papers
12	119	12
19	132	18
21	120	12

3. School management wants to know the employees having the birth date on the same day to plan for the leaves they provide to employees. Please check which of the 2 employees in employee table have birth date on the same day. [Hint: Use Self JOIN]

```
SELECT E1.Employee_Name, E2.Employee_Name

FROM employee AS E1, employee AS E2

WHERE E1.Employee_Id != E2.Employee_Id

AND E1.Employee_Birthdate = E2.Employee_Birthdate;
```

Employee_Name	Employee_Name
Pravesh	Priya
Priya	Pravesh

4. Get the TOP 2 employees name and their ratings who got the best overall ratings from students.

```
SELECT Employee_Name, AVG(Rating) AS Avg_Rating
                FROM ratings AS r
                JOIN employee AS e
                ON r.Employee_Id = e.Employee_Id
                GROUP BY r.Employee_Id, Employee_Name
                ORDER BY Avg_Rating DESC LIMIT 2
```

Employee_Name	Avg_Rating
Gyanesh	5.0000
Amrinder	4.3333

Part 2:

Let's suppose school wants to open up career counselling sessions for students. For that they will not have professionals from industry guiding the students but also professors having good bonding with students along with professors for administrative work. Please identify the following:

1. Get the professors that aren't involved in any courses as of now.

```
SELECT *

FROM employee as e

WHERE Employee_designation = 'Professor'

AND Employee_Id NOT IN (SELECT

                        DISTINCT Professor_Id FROM course);
```

Employee_Id	Employee_Name	Employee_City	Employee_Birthdate	Employee_designation	Employee_since
1	Priya	Gurgaon	1992-06-06	Professor	2020
5	Anagh	Delhi	2000-09-29	Professor	2022
6	Pravesh	Alwar	1992-06-06	Professor	2015
7	Samyara	Delhi	1994-10-15	Professor	2022
17	Shashank	Bhiwadi	1998-08-26	Professor	2022
18	Nikita	Delhi	1991-07-30	Professor	2015
20	Sukhvinder	Jaipur	1988-03-17	Professor	2019

2. Get the employee name along where the average paper >= 3 and assignments >20.

```
SELECT *

FROM employee as e

WHERE EXISTS (

SELECT Professor_Id, AVG(Course_Assignments) AS CA,
AVG(Course_Paper) AS CP

FROM course as c

  WHERE e.Employee_Id = c.Professor_Id

GROUP BY Professor_Id

HAVING CA > 20 and CP >= 3);
```

Employee_Id	Employee_Name	Employee_City	Employee_Birthdate	Employee_designation	Employee_since
19	Amrinder	Hisar	1985-10-28	Professor	2013

3. Get the employees that are rated by students. [Solve below 2 with IN and EXIST both operators]
    1. Get the employees with more than 2 students ratings. [Note: Identify number of students]

```
SELECT *
```

```
FROM employee as e

WHERE EXISTS

(SELECT r.Employee_Id FROM ratings AS r

WHERE e.Employee_Id = r.Employee_Id

GROUP BY r.Employee_Id

HAVING count(r.Student_Id) >= 2

);
```

Employee_Id	Employee_Name	Employee_City	Employee_Birthdate	Employee_designation	Employee_since
2	Komal	Gurgaon	1992-04-29	Professor	2021
9	Gyanesh	Noida	1992-11-02	Professor	2016
19	Amrinder	Hisar	1985-10-28	Professor	2013

2. Get the employees with an average rating of more than 4 and are rated by more than 3 students.

```
SELECT *

FROM employee as e

WHERE EXISTS

(SELECT r.Employee_Id

 FROM ratings AS r

 WHERE e.Employee_Id = r.Employee_Id

GROUP BY r.Employee_Id

 HAVING COUNT(Student_Id) > 3 AND AVG(Rating) >= 4

);
```

Employee_Id	Employee_Name	Employee_City	Employee_Birthdate	Employee_designation	Employee_since
2	Komal	Gurgaon	1992-04-29	Professor	2021