PROJECT FOR SQL MODULE

STUDENT DATABASE MANAGEMENT

Institute Name – Itvedant Education Pvt. Ltd

Name – Arun Shankar Ovhal

Email Address- arun77ovhal@gmail.com

Date Of Submission: -16 Nov 2023

AIM OF THE PROJECT

The aim of the project appears to be comprehensive student management within an educational institution. The database structure and its associated tables seem designed to facilitate the organization, tracking, and management of various aspects of student information throughout their academic journey.

Key objectives of the project might include:

Holistic Student Data Management: The project aims to centralize and organize diverse data concerning students, including personal details, academic performance, financial records, certificate status, and pending dues.

Efficient Administrative Operations: Providing a system that aids administrators in overseeing various student-related tasks, such as fee management, academic records, and certification processes.

Insight Generation: Enabling the generation of insights or reports by analyzing the collected data. This could aid in decision-making, identifying trends, and understanding student performance and financial statuses.

Automating Processes: Potentially automating reminders or notifications regarding pending dues, certification status, or outstanding payments.

Enhanced Transparency: Facilitating a transparent view of a student's comprehensive status across different domains, including academic, financial, and administrative aspects.

The ultimate goal is likely to create a robust, centralized system that streamlines student data management, ensuring accuracy, accessibility, and efficiency for educational institutions in handling student-related information.

This database contains 5 tables:

1. Student_Info
2. Fees_Info
3. Academic_Info
4. NDC_Form
5. Student_Status

Description:- The student database management system aims to provide a holistic operation of the students. The management tool efficiently handles student information including personal information, Fees Record, Academic Records, NDC Report, and Student Status.

1.Student_Info :- Contains personal details of students, including their roll numbers, names, dates of birth, genders, and admission dates.

2.Fees_Info:- Stores financial information about students, such as their total fees, discounts, the amount to pay, outstanding dues, and payment status.

3.Academic_Info:- Holds academic records, including roll numbers, courses, percentages, grades, and pass/fail results.

4.NDC Form:- Records whether students have dues in different areas—administration, fees, academics, and the library.

5.Student_Status:- Tracks the status of students' certificate collections, including ASOMs, degree certificates, LC/MC/TC, collection dates, and associated feedback.

ER-Diagram(Entity Relation-Diagram) for Student Database Management.

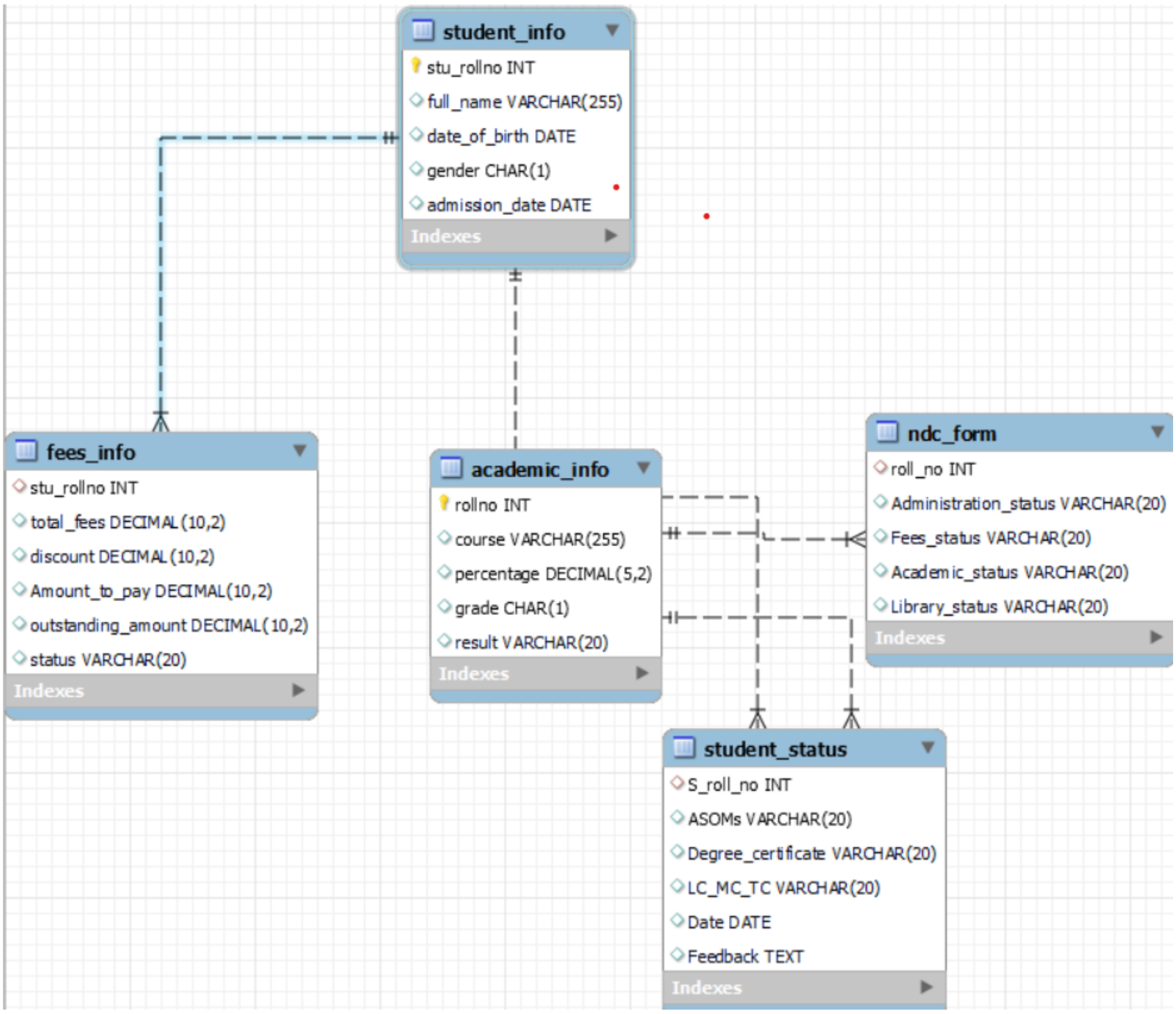

Table Description:

1.Student_info:

Field	Type	Null	Key	Default	Extra
stu_rollno	Int	NO	PRI	NULL	
full_name	varchar(255)	YES		NULL	
date_of_birth	Date	YES		NULL	
Gender	char(1)	YES		NULL	
admission_date	Date	YES		NULL	

2.Fees_Info:

Field	Type	Null	Key	Default	Extra
stu_rollno	Int	YES	MUL	NULL	
total_fees	decimal(10,2)	YES		NULL	
Discount	decimal(10,2)	YES		NULL	
Amount_to_pay	decimal(10,2)	YES		NULL	
outstanding_amount	decimal(10,2)	YES		NULL	
Status	varchar(20)	YES		NULL	

3.Academic_Info

Field	Type	Null	Key	Default	Extra
Rollno	Int	NO	PRI	NULL	
Course	varchar(255)	YES		NULL	
Percentage	decimal(5,2)	YES		NULL	
Grade	char(1)	YES		NULL	
Result	varchar(20)	YES		NULL	

4.NDC_Info:

Field	Type	Null	Key	Default	Extra
roll_no	Int	YES	MUL	NULL	
Administration_status	varchar(20)	YES		NULL	
Fees_status	varchar(20)	YES		NULL	
Academic_status	varchar(20)	YES		NULL	
Library_status	varchar(20)	YES		NULL	

5.Student_Info:

Field	Type	Null	Key	Default	Extra
S_roll_no	Int	YES	MUL	NULL	
ASOMs	varchar(20)	YES		NULL	
Degree_certificate	varchar(20)	YES		NULL	
LC_MC_TC	varchar(20)	YES		NULL	
Date	Date	YES		NULL	
Feedback	Text	YES		NULL	

Commands

Create Database:

```
create database student_management;

use student_management;
```

Create table named student_ info:

```
CREATE TABLE student_info (

stu_rollno int primary key,

full_name varchar(255),

date_of_birth date,

gender char(1),

admission_date date);
```

Create table named fees_ info:

```
stu_rollno INT,

total_fees decimal(10, 2),

discount decimal(10, 2),

Amount_to_pay decimal(10, 2),

outstanding_amount decimal(10,2),

status varchar(20),

foreign key (stu_rollno) references student_info (stu_rollno));
```

Create table named Academic_info:

```
CREATE TABLE academic_info (

rollno int primary key,

course varchar(255),

percentage decimal(5, 2),

grade char(1),

result varchar(20));
```

Create table named NDC_Form:

```
CREATE TABLE NDC_form (

roll_no int,

Administration_status varchar(20),

Fees_status varchar (20),

Academic_status varchar (20),

Library_status varchar (20),

foreign key (roll_no) references academic_info (rollno));
```

Create table named Student_status:

```
S_roll_no int,

ASOMs varchar(20),

Degree_certificate varchar(20),

LC_MC_TC varchar(20),

Date date,

Feedback text,

foreign key (S_roll_no) references academic_info (rollno),

foreign key (S_roll_no) references student_info (stu_rollno));
```

* Populate “student_info” table:

```
INSERT INTO student_info (stu_rollno, full_name, date_of_birth, gender,
admission_date)

VALUES

   (1, 'John Smith', '2000-03-15', 'M', '2022-06-17'),

   (2, 'Jane Doe', '2001-07-20', 'F', '2022-08-09'),

   (3, 'Michael Johnson', '2002-01-10', 'M', '2021-08-28'),

   (4, 'Emily Davis', '2000-11-05', 'F', '2021-09-10'),

   (5, 'Robert Wilson', '2003-04-25', 'M', '2021-08-30'),

   (6, 'Maria Garcia', '2001-09-12', 'F', '2022-09-03'),

   (7, 'William Brown', '2002-12-30', 'M', '2021-08-27'),

   (8, 'Sophia Lee', '2001-06-08', 'F', '2021-09-02'),

   (9, 'James Martinez', '2000-05-18', 'M', '2021-09-07'),

   (10, 'Olivia Taylor', '2003-02-03', 'F', '2020-09-08'),

   (11, 'Daniel Anderson', '2001-11-22', 'M', '2021-09-04'),

   (12, 'Chloe White', '2002-08-14', 'F', '2021-09-06'),

   (13, 'Matthew Thomas', '2000-10-02', 'M', '2021-08-29'),

   (14, 'Ava Harris', '2003-05-30', 'F', '2021-08-31'),

   (15, 'David Johnson', '2001-04-19', 'M', '2022-09-09'),

   (16, 'Emma Miller', '2002-07-07', 'F', '2021-08-26'),

   (17, 'Andrew Wilson', '2000-09-26', 'M', '2021-09-11'),

   (18, 'Isabella Clark', '2003-01-08', 'F', '2021-08-25'),

   (19, 'Joseph Walker', '2001-03-17', 'M', '2020-08-24'),

   (20, 'Sophia Robinson', '2002-06-23', 'F', '2021-09-12');
```

* Populate “fees_info” table:

```
INSERT INTO fees_info (stu_rollno, total_fees, discount, Amount_to_pay,
outstanding_amount, status)

VALUES

   (1, 5000.00, 500.00, 4500.00, 0.00, 'Paid'),

   (2, 5500.00, 550.00, 4950.00, 0.00, 'Paid'),

   (3, 6000.00, 600.00, 5400.00, 1200.00, 'balance'),

   (4, 5200.00, 520.00, 4680.00, 0.00, 'Paid'),

   (5, 4800.00, 480.00, 4320.00, 0.00, 'Paid'),

   (6, 5600.00, 560.00, 5040.00, 500.00, 'balance'),

   (7, 5800.00, 580.00, 5220.00, 0.00, 'Paid'),

   (8, 5100.00, 510.00, 4590.00, 1700.00, 'balance'),

   (9, 5250.00, 525.00, 4725.00, 0.00, 'Paid'),

   (10, 5050.00, 505.00, 4545.00, 1765.00, 'balance'),

   (11, 5300.00, 530.00, 4770.00, 0.00, 'Paid'),

   (12, 5400.00, 540.00, 4860.00, 0.00, 'Paid'),

   (13, 5050.00, 505.00, 4545.00, 1500.00, 'balance'),

   (14, 5500.00, 550.00, 4950.00, 0.00, 'Paid'),

   (15, 5250.00, 525.00, 4725.00, 0.00, 'Paid'),

   (16, 5200.00, 520.00, 4680.00, 0.00, 'Paid'),

   (17, 5000.00, 500.00, 4500.00, 3000.00, 'balance'),

   (18, 5600.00, 560.00, 5040.00, 0.00, 'Paid'),

   (19, 5300.00, 530.00, 4770.00, 0.00, 'Paid'),

   (20, 5800.00, 580.00, 5220.00, 1000.00, 'balance');
```

*** Populate “Academic_info” table:**

```
INSERT INTO academic_info (rollno, course, percentage, grade, result)
VALUES
    (1, 'Mathematics', 95.50, 'A', 'Pass'),
    (2, 'Science', 30.72, 'E', 'Fail'),
    (3, 'History', 75.25, 'C', 'Pass'),
    (4, 'English', 34.00, 'E', 'Fail'),
    (5, 'Computer Science', 33.33, 'E', 'Fail'),
    (6, 'Physics', 76.50, 'C', 'Pass'),
    (7, 'Chemistry', 91.25, 'A', 'Pass'),
    (8, 'Geography', 82.00, 'B', 'Pass'),
    (9, 'Biology', 74.75, 'C', 'Pass'),
    (10, 'Art', 87.00, 'B', 'Pass'),
    (11, 'Economics', 60.45, 'D', 'Pass'),
    (12, 'Physical Education', 70.50, 'C', 'Pass'),
    (13, 'Music', 85.75, 'B', 'Pass'),
    (14, 'Psychology', 77.00, 'C', 'Pass'),
    (15, 'Sociology', 94.50, 'A', 'Pass'),
    (16, 'Political Science', 58.25, 'D', 'Pass'),
    (17, 'Business Studies', 73.75, 'C', 'Pass'),
    (18, 'Foreign Language', 86.00, 'B', 'Pass'),
    (19, 'Physical Science', 80.25, 'B', 'Pass'),
    (20, 'Environmental Science', 27.00, 'E', 'Fail');
```

* Populate “NDC_Form” table:

```
INSERT INTO NDC_form (roll_no, Administration_status, Fees_status, Academic_status,
Library_status)

VALUES

   (1, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (2, 'No Dues', 'No Dues', 'Dues', 'No Dues'),

   (3, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (4, 'No Dues', 'NO Dues', 'Dues', 'No Dues'),

   (5, 'No Dues', 'No Dues', 'Dues', 'Dues'),

   (6, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (7, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (8, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (9, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (10, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (11, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (12, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (13, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (14, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (15, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (16, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (17, 'No Dues', 'Dues', 'No Dues', 'No Dues'),

   (18, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (19, 'No Dues', 'No Dues', 'No Dues', 'No Dues'),

   (20, 'No Dues', 'Dues', 'Dues', 'No Dues');
```

* Populate “Student_Status” table:

```
INSERT INTO student_status (S_roll_no, ASOMs, Degree_certificate, LC_MC_TC, Date,
Feedback)

VALUES

   (1, 'Received', 'Received', 'Received', '2023-10-17', 'Top student.'),

   (2, 'Pending', 'Pending', 'Pending',null, 'Need Attention.'),

   (3, 'Pending', 'Pending', 'Pending',null, 'Well done..'),

   (4, 'Pending', 'Pending', 'Pending',null, 'Need Attention.'),

   (5, 'Pending', 'Pending', 'Pending',null, 'Need Attention.'),

   (6, 'Pending', 'Pending', 'Pending',null, 'Well done.'),

   (7, 'Received', 'Received', 'Received', '2023-10-07', 'Excellent performance.'),

   (8, 'Pending', 'Pending', 'Pending',null, 'Well Done.'),

   (9, 'Received', 'Received', 'Received', '2023-10-09', 'Well Done.'),

   (10, 'Pending', 'Pending', 'Pending',null, 'Highly recommended.'),

   (11, 'Received', 'Received', 'Received', '2023-10-11', 'Satisfactory.'),

   (12, 'Received', 'Received', 'Received', '2023-10-08', 'Well done.'),

   (13, 'Pending', 'Pending', 'Pending',null, 'Well done.'),

   (14, 'Received', 'Received', 'Received', '2023-10-14', 'Well Done.'),

   (15, 'Received', 'Received', 'Received', '2023-10-15', 'Excellent performance.'),

   (16, 'Received', 'Received', 'Received', '2023-10-16', 'Satisfactory.'),

   (17, 'Pending', 'Pending', 'Pending',null, 'Well done.'),

   (18,'Received', 'Received', 'Received', '2023-10-18', 'Excellent performance.'),

   (19, 'Received', 'Received', 'Received', '2023-10-19', 'Well done.'),

   (20, 'Pending', 'Pending', 'Pending',null, 'Need Attention.');
```

SUBQUERIES

1.Query to find the number of male students who were admitted to the school in each year.

```
Select year(admission_date) AS year, count(*) as num_male_students
from student_info where gender = 'M' group by year(admission_date)
order by year asc;
```

Year	num_male_students
2020	1
2021	7
2022	2

2. Query to find the no dues clear student (rollno) from table ndc_form.

```
Select roll_no from ndc_form where administration_status='No Dues' and
Fees_status='No Dues'and Academic_status='No Dues' and Library_status='No
Dues';
```

roll_no
1
7
9
11
12
14
15
16
18
19

3.Query to find the student who has collected there certificate.

```
select * from student_status where date is not null;
```

S_roll_no	ASOMs	Degree_certificate	LC_MC_TC	Date	Feedback
1	Received	Received	Received	17-10-2023	Top student.
7	Received	Received	Received	07-10-2023	Excellent performance.
9	Received	Received	Received	09-10-2023	Well Done.
11	Received	Received	Received	11-10-2023	Satisfactory.
12	Received	Received	Received	08-10-2023	Well done.
14	Received	Received	Received	14-10-2023	Well Done.
15	Received	Received	Received	15-10-2023	Excellent performance.
16	Received	Received	Received	16-10-2023	Satisfactory.
18	Received	Received	Received	18-10-2023	Excellent performance.
19	Received	Received	Received	19-10-2023	Well done.

*JOINS

1.Query to find the 3 tables data in one table which will help to find the academic result, fees and final status of the student from the table.

```
select a.rollno,a.result,

ndc.Administration_status, ndc.Fees_status, ndc.Academic_status, ndc.Library_status,

ss.ASOMs, ss.Degree_certificate, ss.LC_MC_TC, ss.Date, ss.Feedback

from academic_info a left join (

select roll_no, Administration_status, Fees_status, Academic_status, Library_status from ndc_form)ndc on a.rollno = ndc.roll_no

left join (select s_roll_no, ASOMs, Degree_certificate, LC_MC_TC, Date, Feedback from student_status) ss ON a.rollno = ss.S_roll_no;
```

Roll no	Result	Admin_status	Fees_status	Academic -status	Library_status	ASOMs	Degree	Lc/mc/tc	Date	Feedback
1	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	17-10-2023	Top student.
2	Fail	No Dues	No Dues	Dues	No Dues	Pending	Pending	Pending	NULL	Need Attention.
3	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Well done..

4	Fail	No Dues	No Dues	Dues	No Dues	Pending	Pending	Pending	NULL	Need Attention.
5	Fail	No Dues	No Dues	Dues	Dues	Pending	Pending	Pending	NULL	Need Attention.
6	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Well done.
7	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	07-10-2023	Excellent performance.
8	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Well Done.
9	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	09-10-2023	Well Done.
10	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Highly recommended.
11	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	11-10-2023	Satisfactory.
12	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	08-10-2023	Well done.
13	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Well done.
14	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	14-10-2023	Well Done.
15	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	15-10-2023	Excellent performance.
16	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	16-10-2023	Satisfactory.
17	Pass	No Dues	Dues	No Dues	No Dues	Pending	Pending	Pending	NULL	Well done.
18	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	18-10-2023	Excellent performance.
19	Pass	No Dues	No Dues	No Dues	No Dues	Received	Received	Received	19-10-2023	Well done.
20	Fail	No Dues	Dues	Dues	No Dues	Pending	Pending	Pending	NULL	Need Attention.

2. Query to find student who has fail, having a dues and certificate pending.

```
select ai.rollno from academic_info ai

left join Ndc_form nf ON ai.rollno = nf.roll_no

left join student_status ss ON ai.rollno = ss.S_roll_no where ai.result = 'Fail'

and nf.Fees_status='Dues'

or nf.Administration_status ='Dues'

or nf.library_status='Dues'

or ss.ASOMs = 'Pending'

or ss.Degree_certificate = 'Pending';
```

Rollno
2
3
4
5
6
8
10
13
17
20

3.Query to find where the student has pass the exam but his fees is pending with outstanding amount.

```
Select ai.rollno, ai.course, ai.result,fi.outstanding_amount,fi.status

from academic_info ai

left join fees_info fi ON ai.rollno = fi.stu_rollno

where ai.result = 'Pass' and  fi.status = 'balance' and fi.outstanding_amount > 0;
```

Rollno	course	result	outstanding_amount	status
3	History	Pass	1200	balance
6	Physics	Pass	500	balance
8	Geography	Pass	1700	balance
10	Art	Pass	1765	balance
13	Music	Pass	1500	balance
17	Business Studies	Pass	3000	balance

CONCLUSION

The database structure covers a wide spectrum of student-related information, ranging from personal and academic details to financial records and certificate collection status. Such a system is invaluable for comprehensive student management.

Record Consolidation: The integration of these tables provides a holistic view of each student's journey. This allows for streamlined record-keeping and easy access to diverse student information.

Efficient Management: Administrators and educators can effectively monitor and manage students' academic, financial, and administrative aspects through this system. They can promptly address issues related to fees, academic performance, and outstanding obligations.

Certificate Tracking: The inclusion of a 'Student_status' table dedicated to certificates allows for efficient tracking of certificate collection, ensuring students meet requirements and collect their certificates in a timely manner.

Insights and Decision Making: The data stored in these tables could be analyzed to derive insights into student performance, financial trends, and overall institutional management, aiding in decision-making processes.

The success of this system hinges on its usability, security, and accuracy of data entry. Designing an intuitive interface for users to interact with this data and ensuring robust security measures will be crucial for the effectiveness and reliability of the system. Additionally, automating certain processes, such as notifications for dues or certificate collection, could significantly enhance its functionality.

Thank You!