Database Management System(UE19CS301)

Assignment - 4

Team Details

Team Number-1

	NAME	SRN
1	**Shatakshi Mohan**	**PES2UG19CS379**
2	**Siddhi Patil**	**PES2UG19CS389**
3	**Sonit Pradhan**	**PES2UG19CS399**

Dependencies installed for the database connectivity:

Psycopg2

The psycopg2 is the PostgreSQL connector commonly used by Python developers to connect to Python. Psycopg2 the driver is designed to be lightweight, fast.

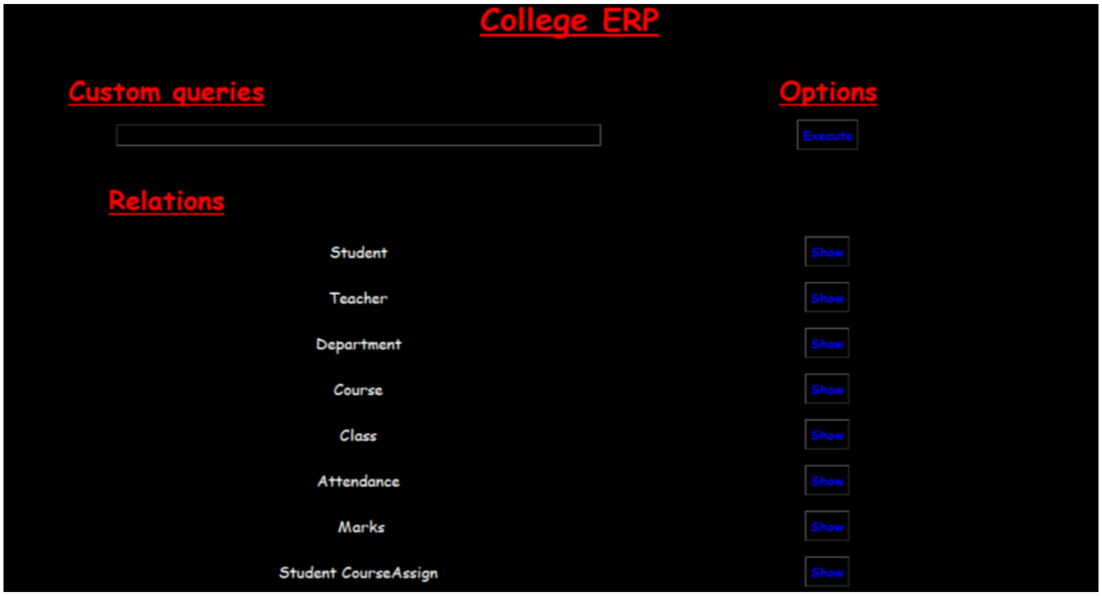
Psycopg2 is a DB API 2.0 compliant PostgreSQL driver that is actively developed. It is designed for multi-threaded applications and manages its own connection pool. Other interesting features of the adapter are that if you are using the PostgreSQL array data type, Psycopg will automatically convert a result using that data type to a Python list.

Since we've used python to create our front end this module appeared to be appropriate.

To install it we used the following command in shell:

```
$pip3 install psycopg2
```

Statements executed from the frontend(Implemented using tkinter):



College ERP

Custom queries

Options

Relations

Student

Teacher

Department

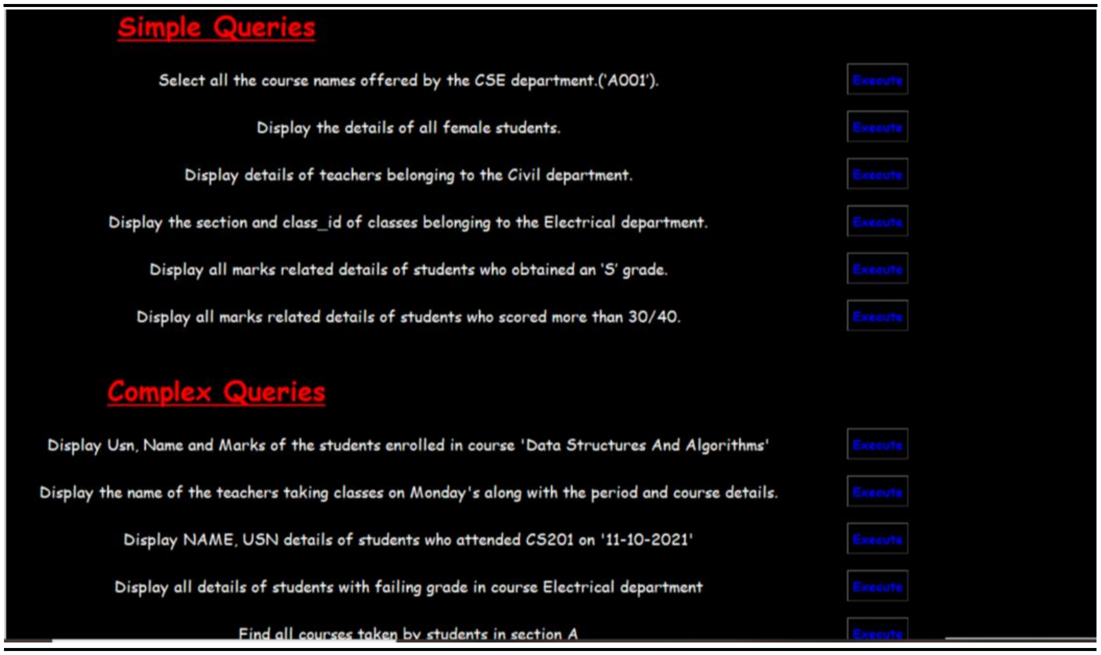
Course

Class

Attendance

Marks

Student CourseAssign



Simple Queries

Select all the course names offered by the CSE department.('A001').

Display the details of all female students.

Display details of teachers belonging to the Civil department.

Display the section and class_id of classes belonging to the Electrical department.

Display all marks related details of students who obtained an 'S' grade.

Display all marks related details of students who scored more than 30/40.

Complex Queries

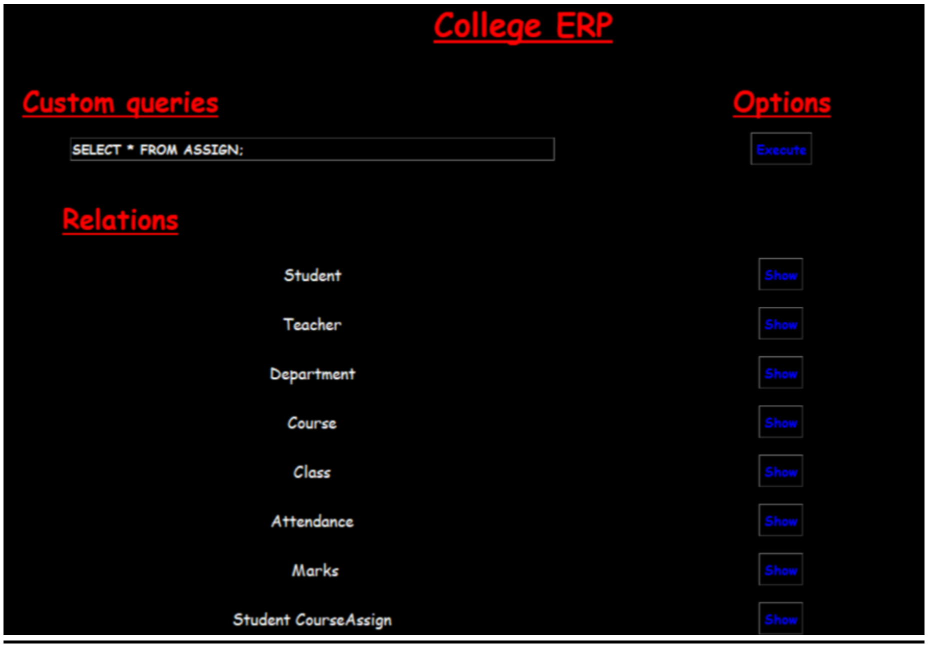
Display Usn, Name and Marks of the students enrolled in course 'Data Structures And Algorithms'

Display the name of the teachers taking classes on Monday's along with the period and course details.

Display NAME, USN details of students who attended CS201 on '11-10-2021'

Display all details of students with failing grade in course Electrical department

Find all courses taken by students in section A



College ERP

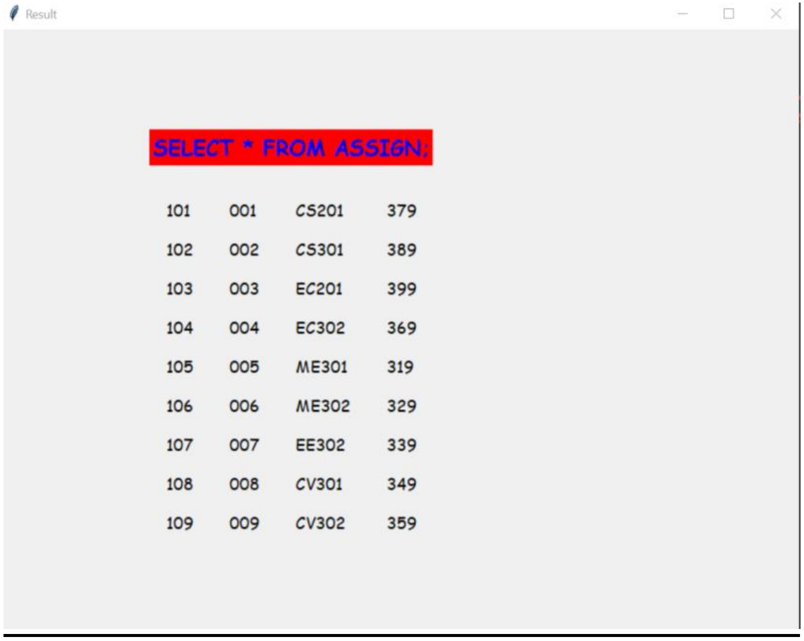
Custom queries

Options

SELECT * FROM ASSIGN;

Relations

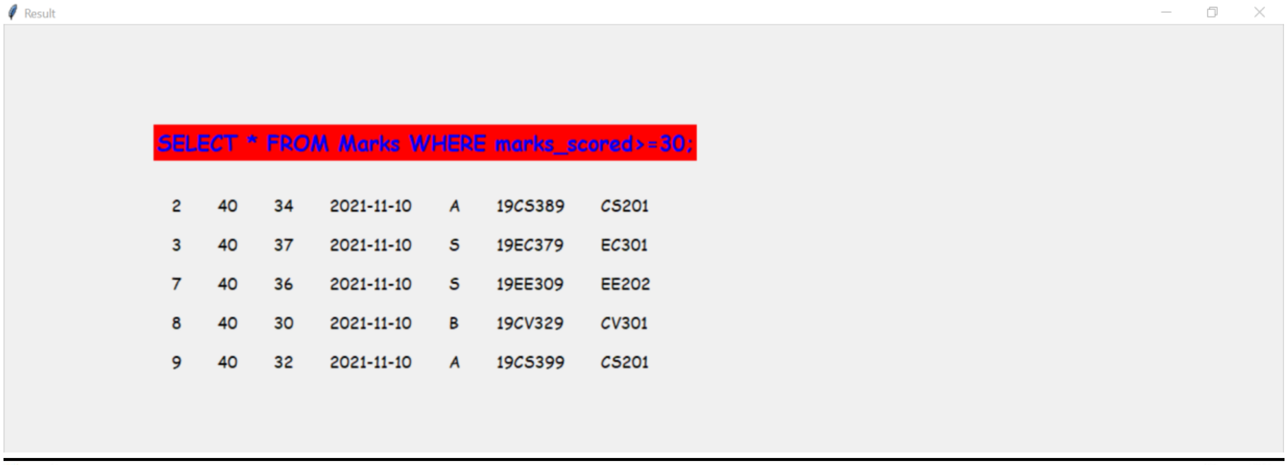
Student	Show
Teacher	Show
Department	Show
Course	Show
Class	Show
Attendance	Show
Marks	Show
Student CourseAssign	Show



SELECT * FROM ASSIGN;

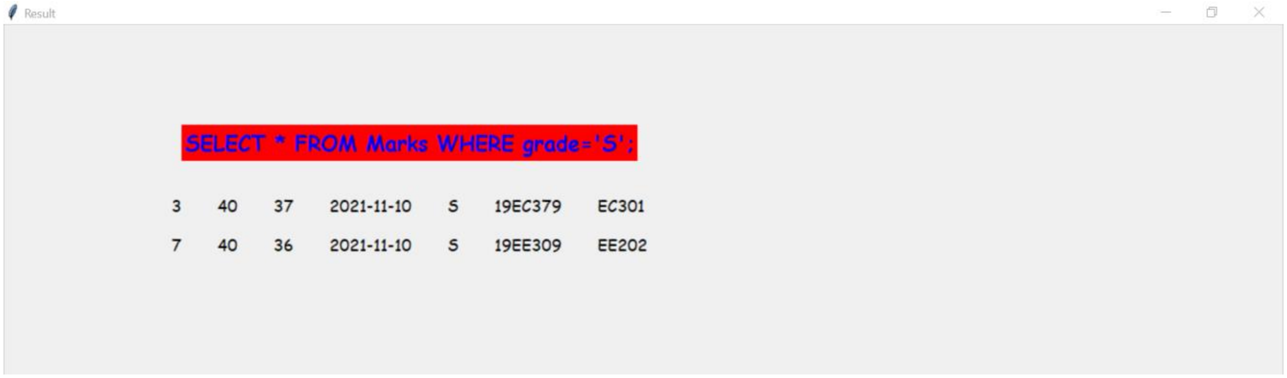
```
101   001   CS201   379
102   002   CS301   389
103   003   EC201   399
104   004   EC302   369
105   005   ME301   319
106   006   ME302   329
107   007   EE302   339
108   008   CV301   349
109   009   CV302   359
```

SIMPLE QUERIES:



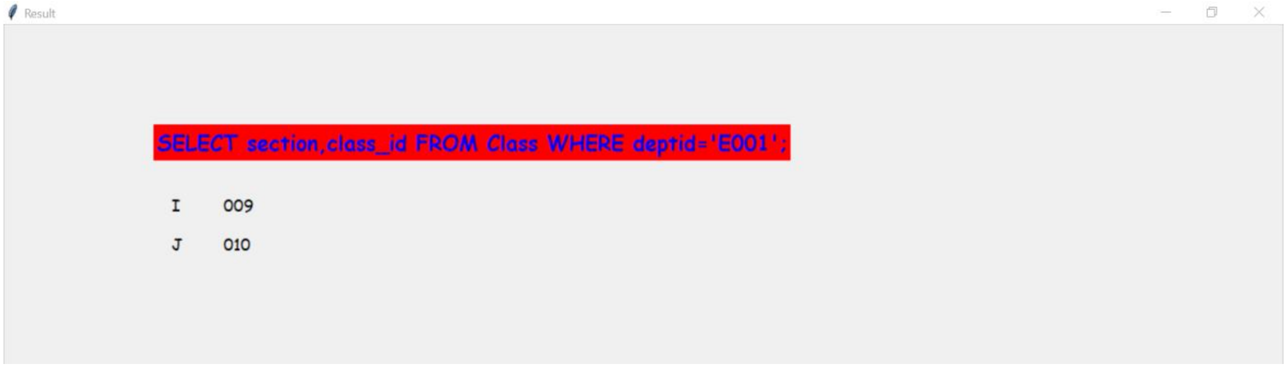
SELECT * FROM Marks WHERE marks_scored>=30;

```
2    40    34    2021-11-10    A    19CS389    CS201
3    40    37    2021-11-10    S    19EC379    EC301
7    40    36    2021-11-10    S    19EE309    EE202
8    40    30    2021-11-10    B    19CV329    CV301
9    40    32    2021-11-10    A    19CS399    CS201
```



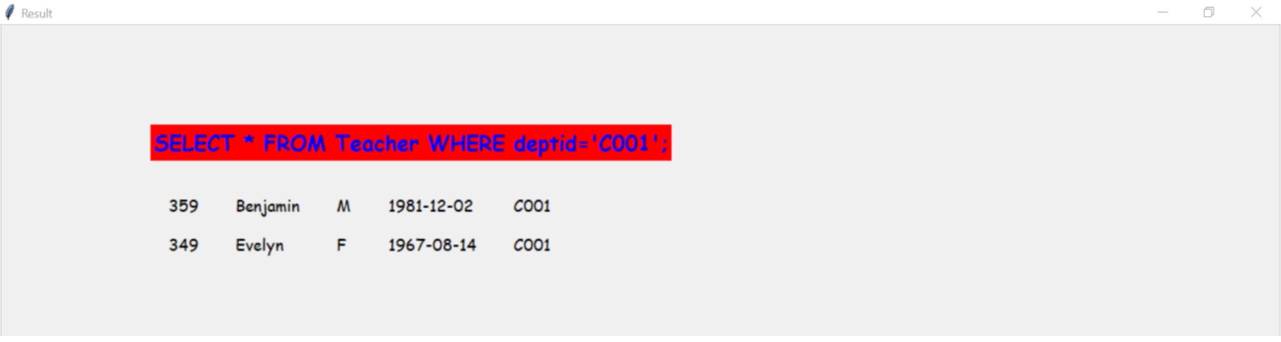
SELECT * FROM Marks WHERE grade='S';

```
3    40    37    2021-11-10    S    19EC379    EC301
7    40    36    2021-11-10    S    19EE309    EE202
```



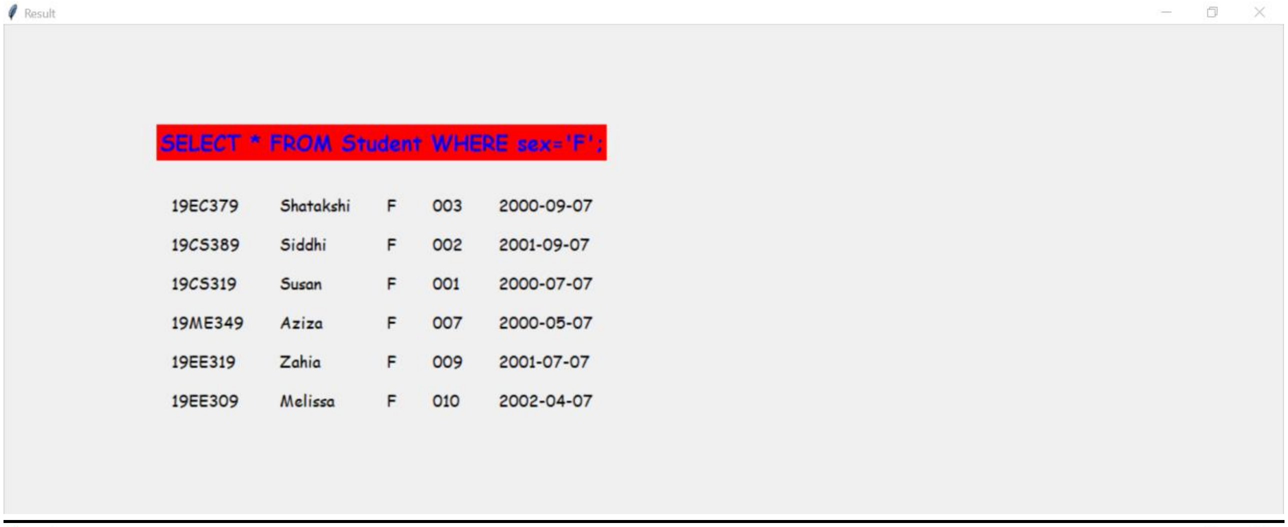
SELECT section,class_id FROM Class WHERE deptid='E001';

```
I    009
J    010
```



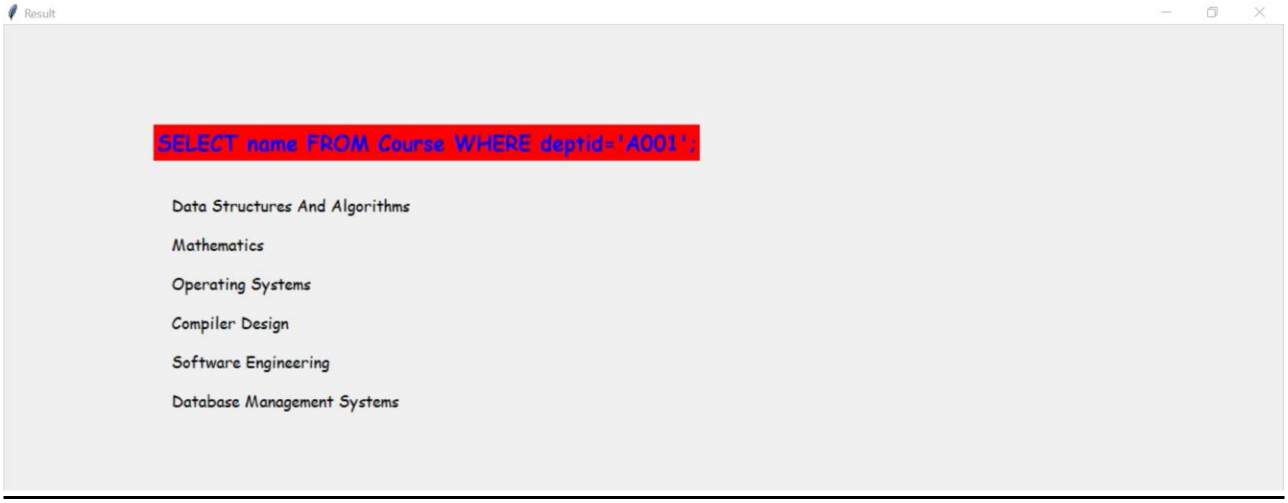
SELECT * FROM Teacher WHERE deptid='C001';

```
359    Benjamin    M    1981-12-02    C001
349    Evelyn      F    1967-08-14    C001
```



```
SELECT * FROM Student WHERE sex='F';
```

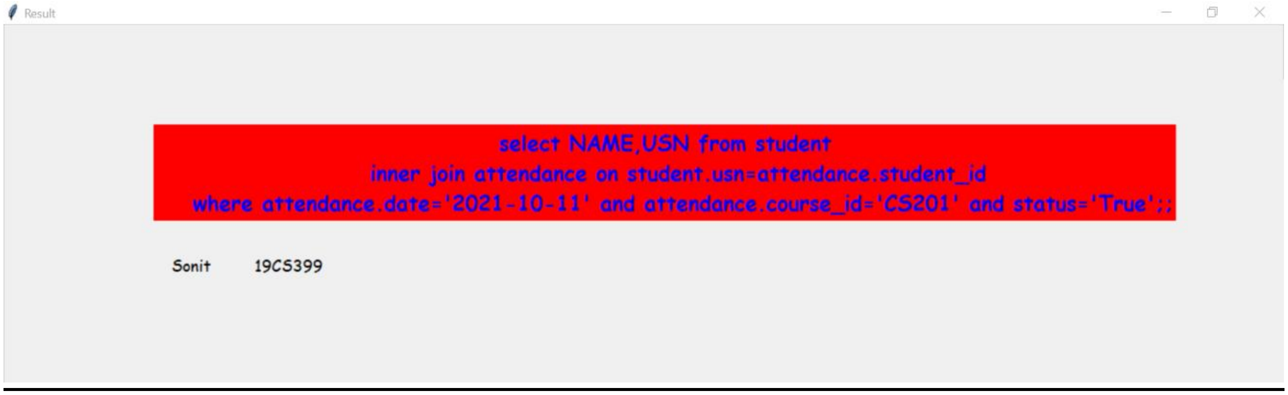
```
19EC379    Shatakshi    F    003    2000-09-07
19CS389    Siddhi       F    002    2001-09-07
19CS319    Susan        F    001    2000-07-07
19ME349    Aziza        F    007    2000-05-07
19EE319    Zahia        F    009    2001-07-07
19EE309    Melissa      F    010    2002-04-07
```



```
SELECT name FROM Course WHERE deptid='A001';
```

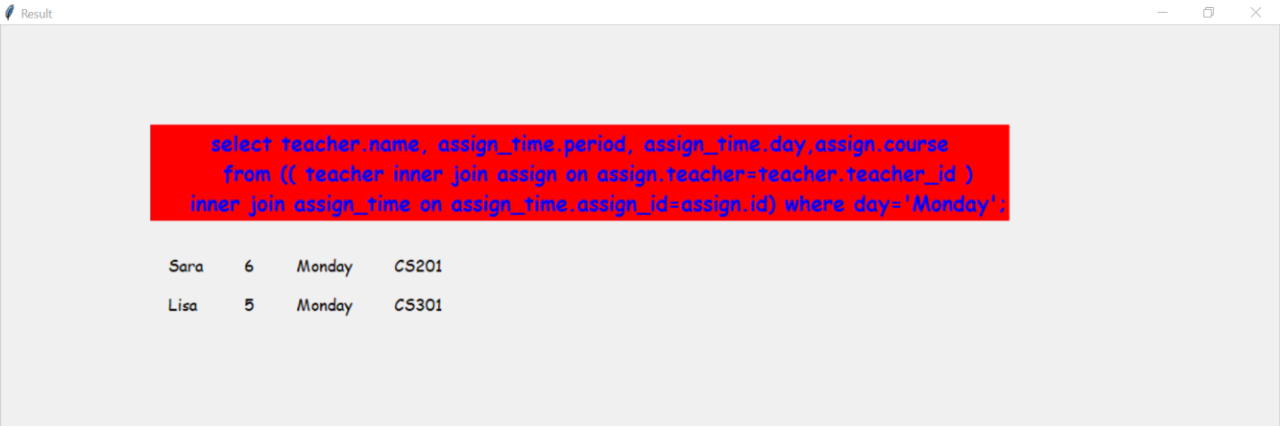
```
Data Structures And Algorithms
Mathematics
Operating Systems
Compiler Design
Software Engineering
Database Management Systems
```

COMPLEX QUERIES:



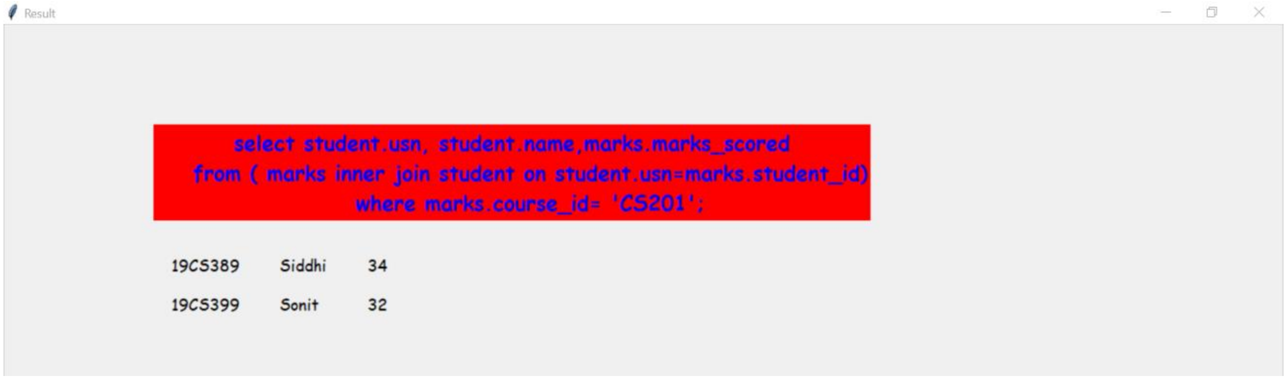
```
select NAME,USN from student
inner join attendance on student.usn=attendance.student_id
where attendance.date='2021-10-11' and attendance.course_id='CS201' and status='True';;
```

```
Sonit    19CS399
```



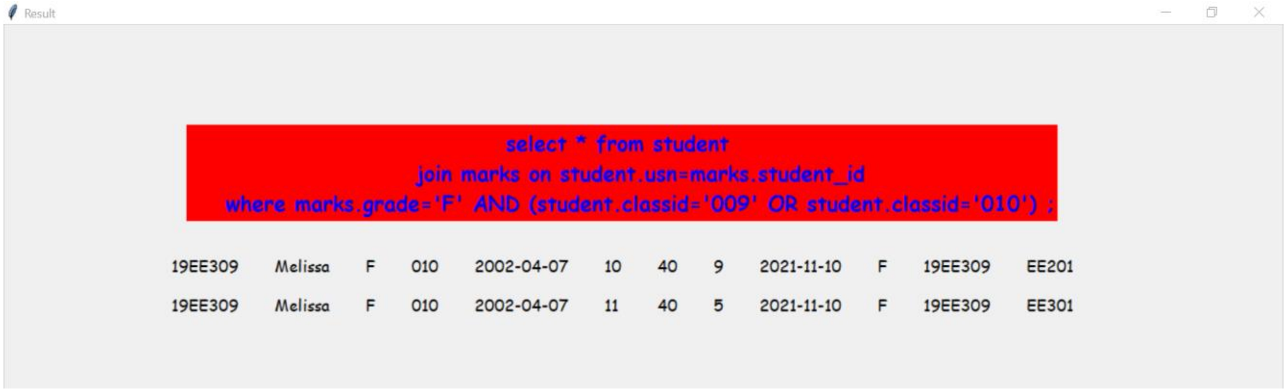
```
select teacher.name, assign_time.period, assign_time.day,assign.course
 from (( teacher inner join assign on assign.teacher=teacher.teacher_id )
inner join assign_time on assign_time.assign_id=assign.id) where day='Monday';
```

```
Sara    6    Monday    CS201
Lisa    5    Monday    CS301
```



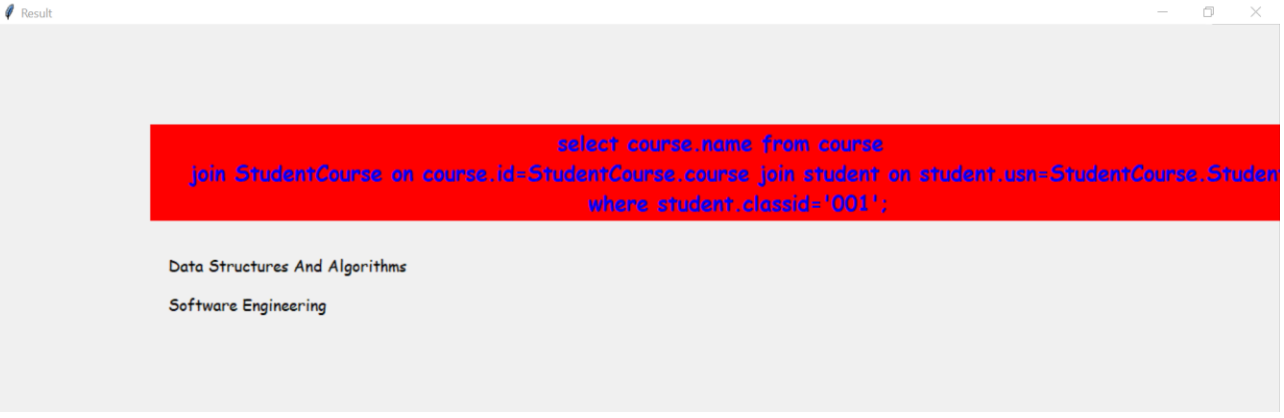
```
select student.usn, student.name,marks.marks_scored
from ( marks inner join student on student.usn=marks.student_id)
where marks.course_id= 'CS201';
```

```
19CS389    Siddhi    34
19CS399    Sonit     32
```



```
select * from student
join marks on student.usn=marks.student_id
where marks.grade='F' AND (student.classid='009' OR student.classid='010') ;
```

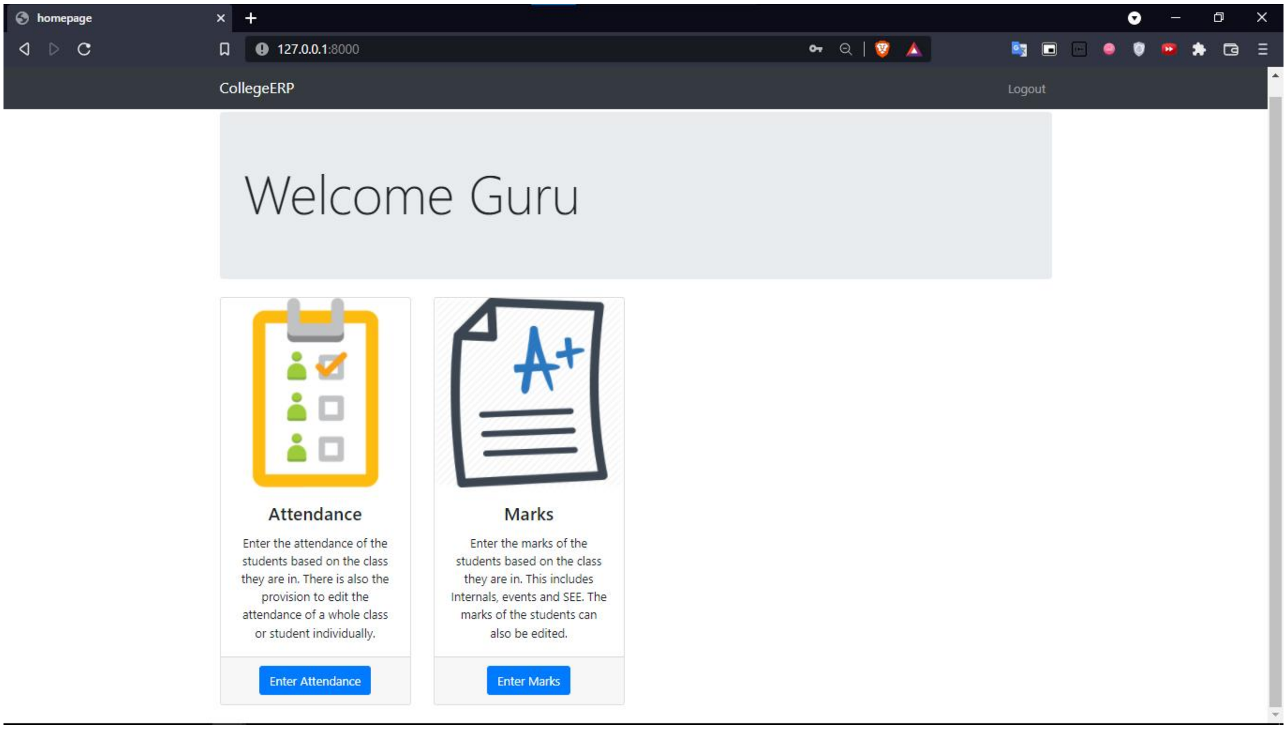
```
19EE309    Melissa    F    010    2002-04-07    10    40    9    2021-11-10    F    19EE309    EE201
19EE309    Melissa    F    010    2002-04-07    11    40    5    2021-11-10    F    19EE309    EE301
```



```
select course.name from course
join StudentCourse on course.id=StudentCourse.course join student on student.usn=StudentCourse.Studen
where student.classid='001';
```

```
Data Structures And Algorithms
Software Engineering
```

UI/frontend(implemented through django framework):



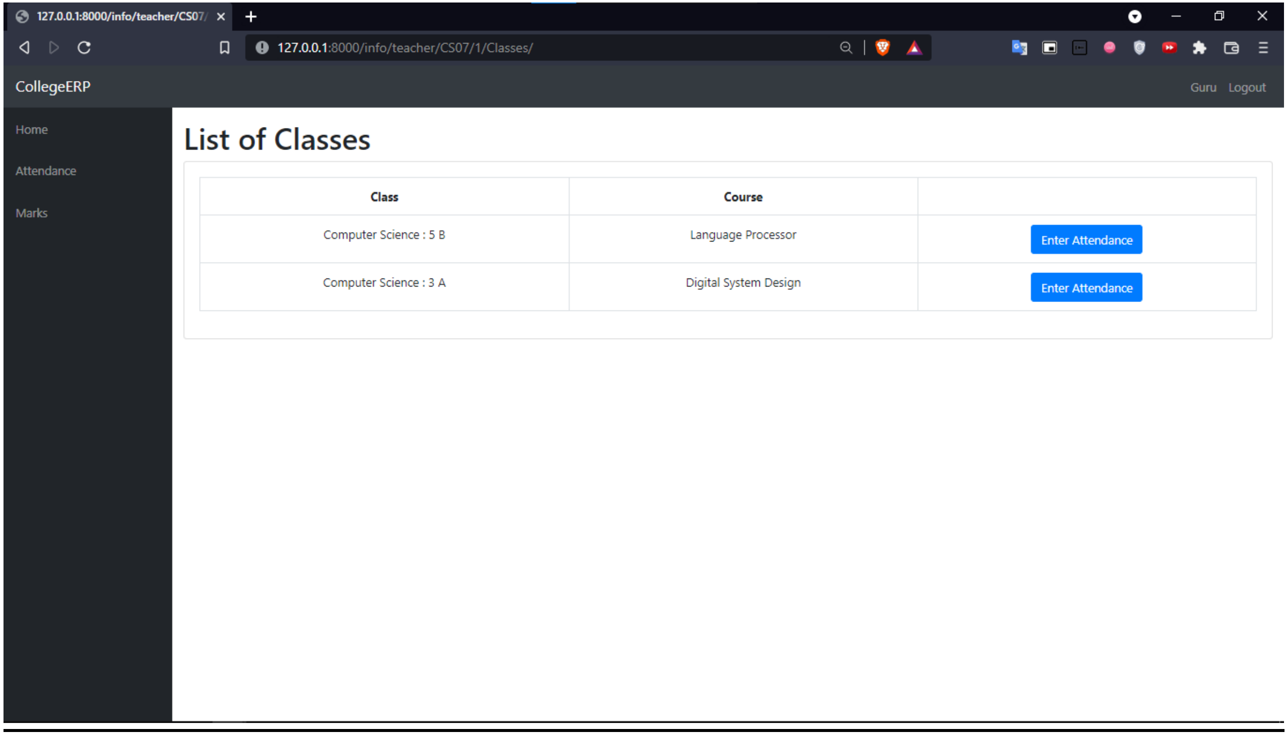
Welcome Guru

Attendance

Enter the attendance of the students based on the class they are in. There is also the provision to edit the attendance of a whole class or student individually.

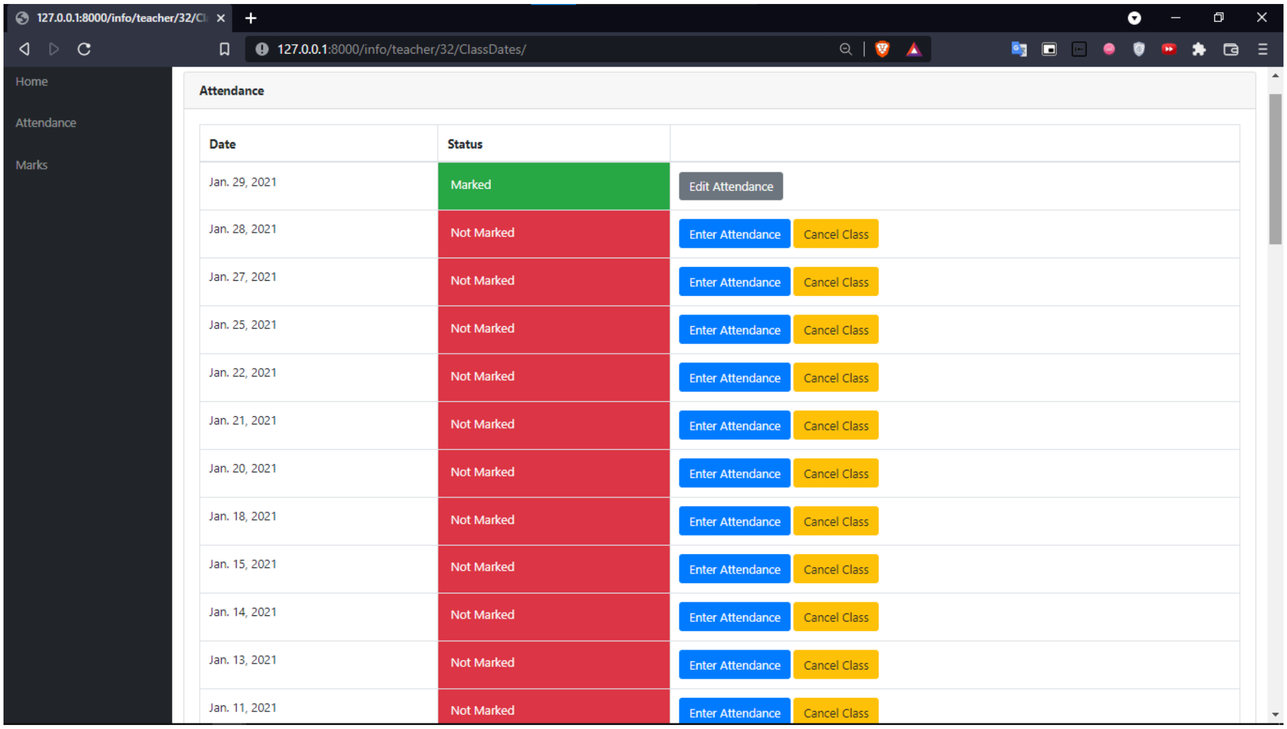
Marks

Enter the marks of the students based on the class they are in. This includes Internals, events and SEE. The marks of the students can also be edited.


127.0.0.1:8000/info/teacher/CS07/
127.0.0.1:8000/info/teacher/CS07/1/Classes/
CollegeERP
Guru
Logout
Home
Attendance
Marks

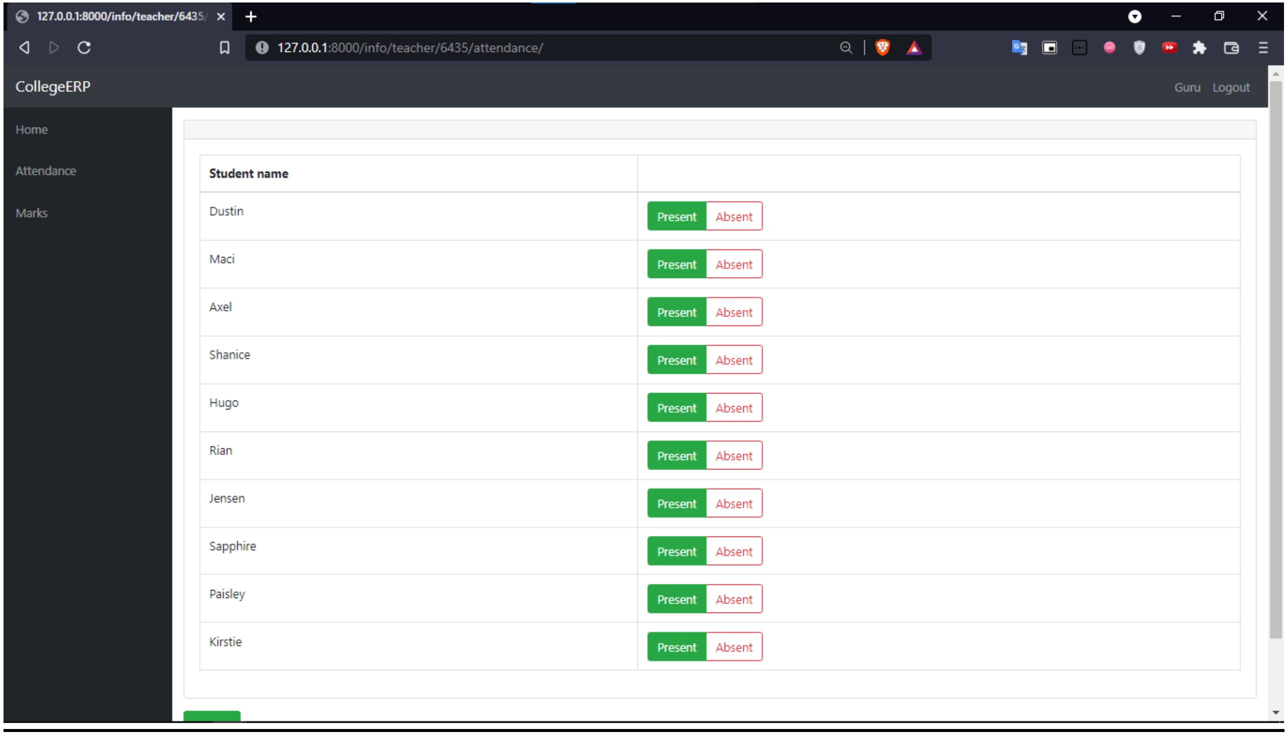
List of Classes

Class	Course	
Computer Science : 5 B	Language Processor	Enter Attendance
Computer Science : 3 A	Digital System Design	Enter Attendance

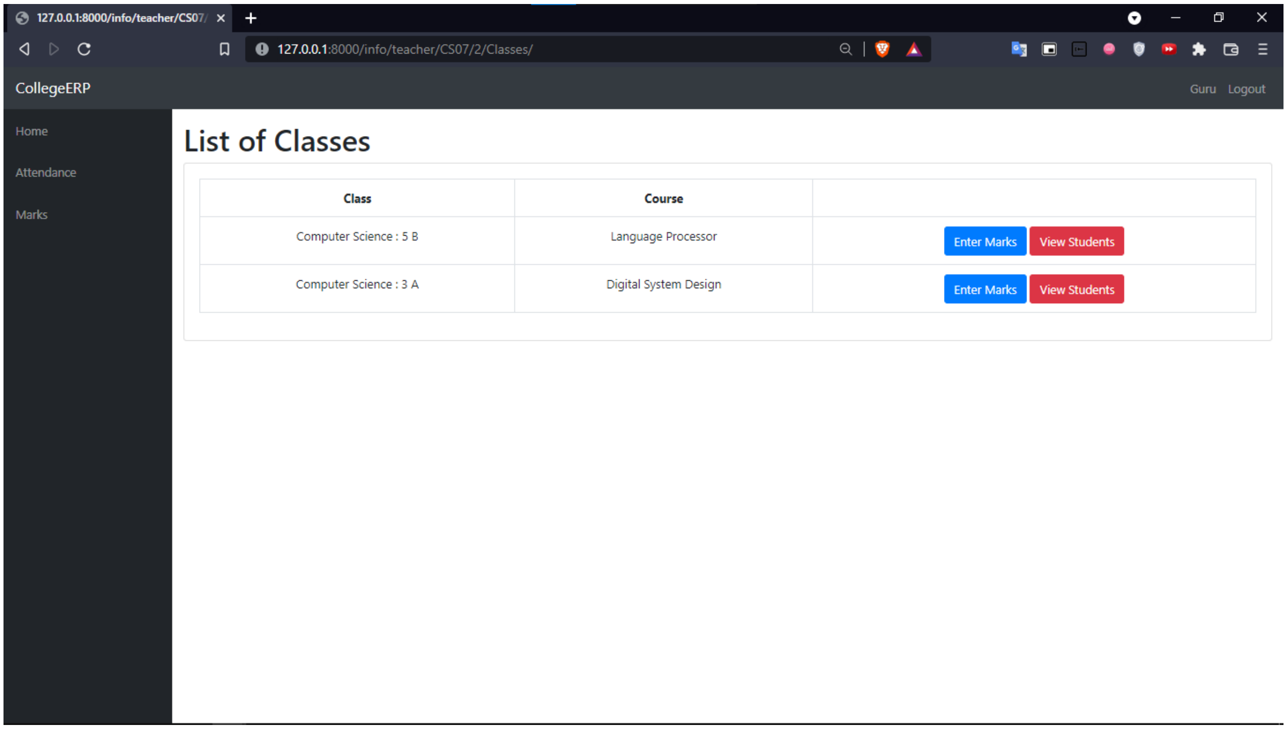

127.0.0.1:8000/info/teacher/32/Cl
127.0.0.1:8000/info/teacher/32/ClassDates/
Home
Attendance
Marks

Attendance

Date	Status	
Jan. 29, 2021	Marked	Edit Attendance
Jan. 28, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 27, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 25, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 22, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 21, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 20, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 18, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 15, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 14, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 13, 2021	Not Marked	Enter Attendance Cancel Class
Jan. 11, 2021	Not Marked	Enter Attendance Cancel Class


127.0.0.1:8000/info/teacher/6435/
127.0.0.1:8000/info/teacher/6435/attendance/
CollegeERP
Guru
Logout
Home
Attendance
Marks

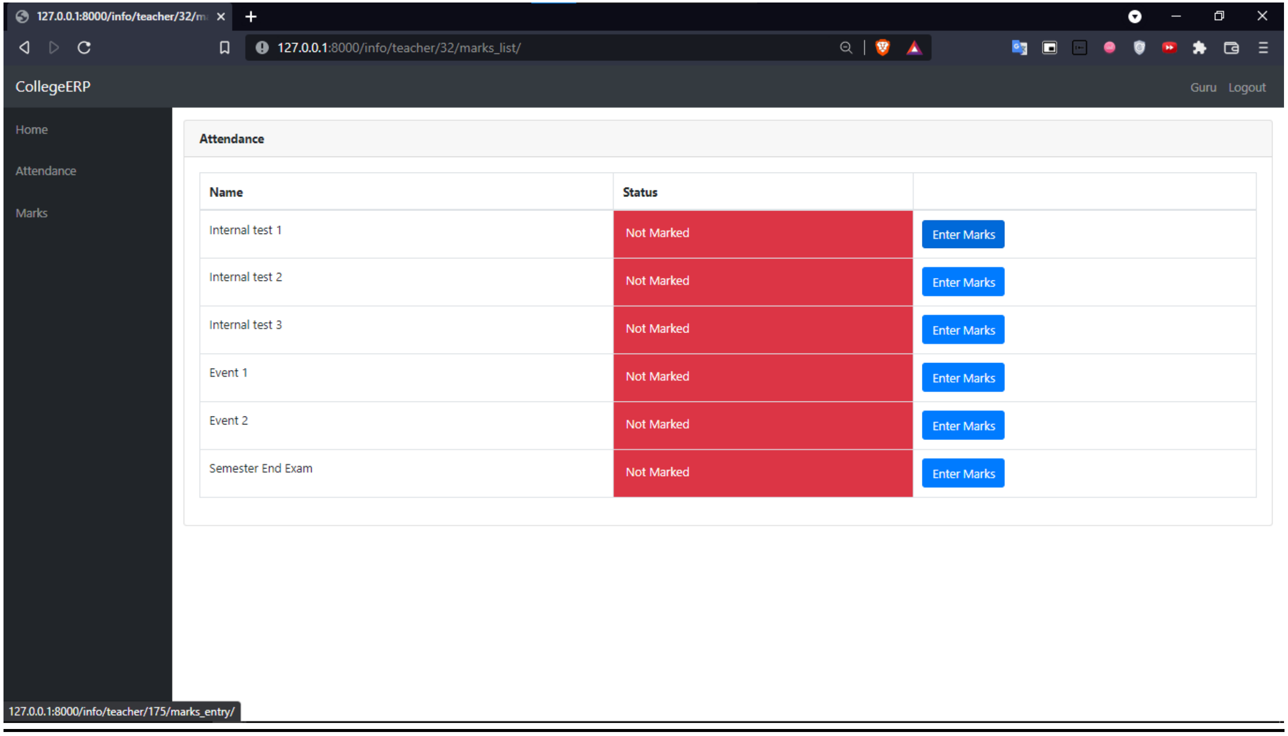
Student name	
Dustin	Present Absent
Maci	Present Absent
Axel	Present Absent
Shanice	Present Absent
Hugo	Present Absent
Rian	Present Absent
Jensen	Present Absent
Sapphire	Present Absent
Paisley	Present Absent
Kirstie	Present Absent



127.0.0.1:8000/info/teacher/CS07/
127.0.0.1:8000/info/teacher/CS07/2/Classes/
CollegeERP
Guru
Logout
Home
Attendance
Marks

List of Classes

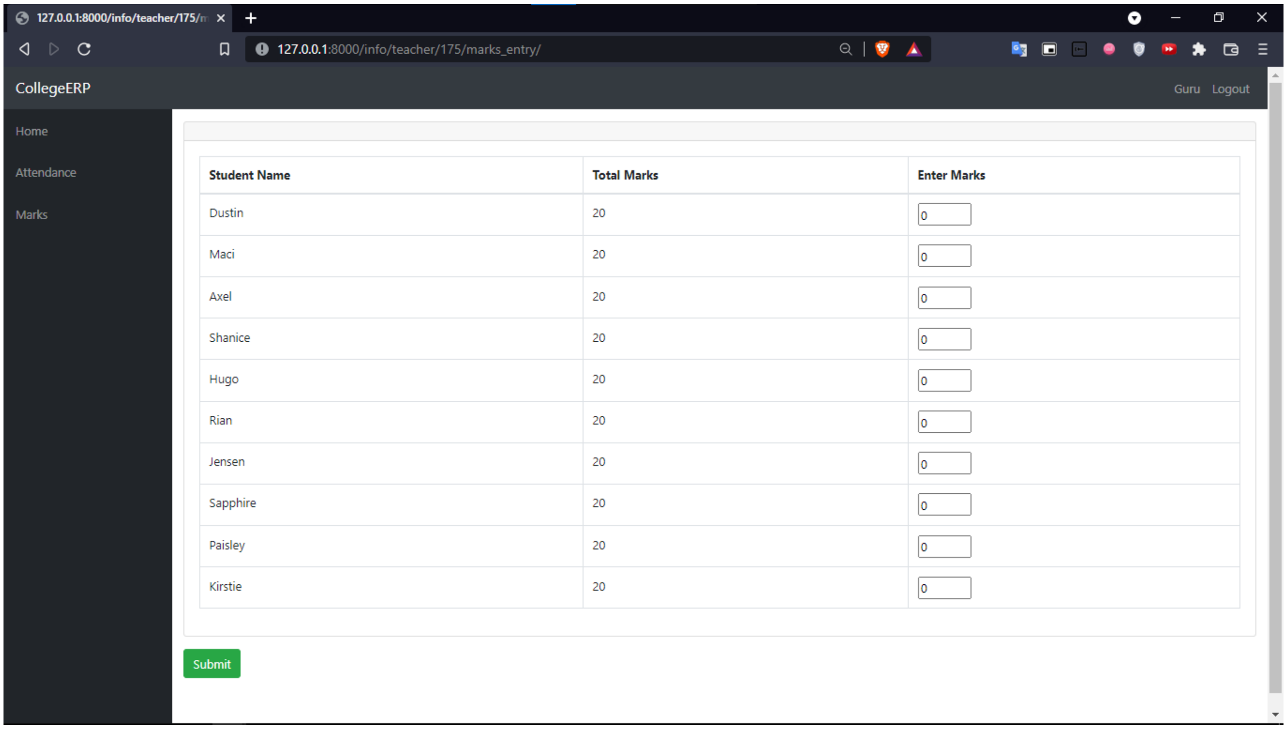
Class	Course	
Computer Science : 5 B	Language Processor	Enter Marks View Students
Computer Science : 3 A	Digital System Design	Enter Marks View Students



127.0.0.1:8000/info/teacher/32/m
127.0.0.1:8000/info/teacher/32/marks_list/
CollegeERP
Guru
Logout
Home
Attendance
Marks
127.0.0.1:8000/info/teacher/175/marks_entry/

Attendance

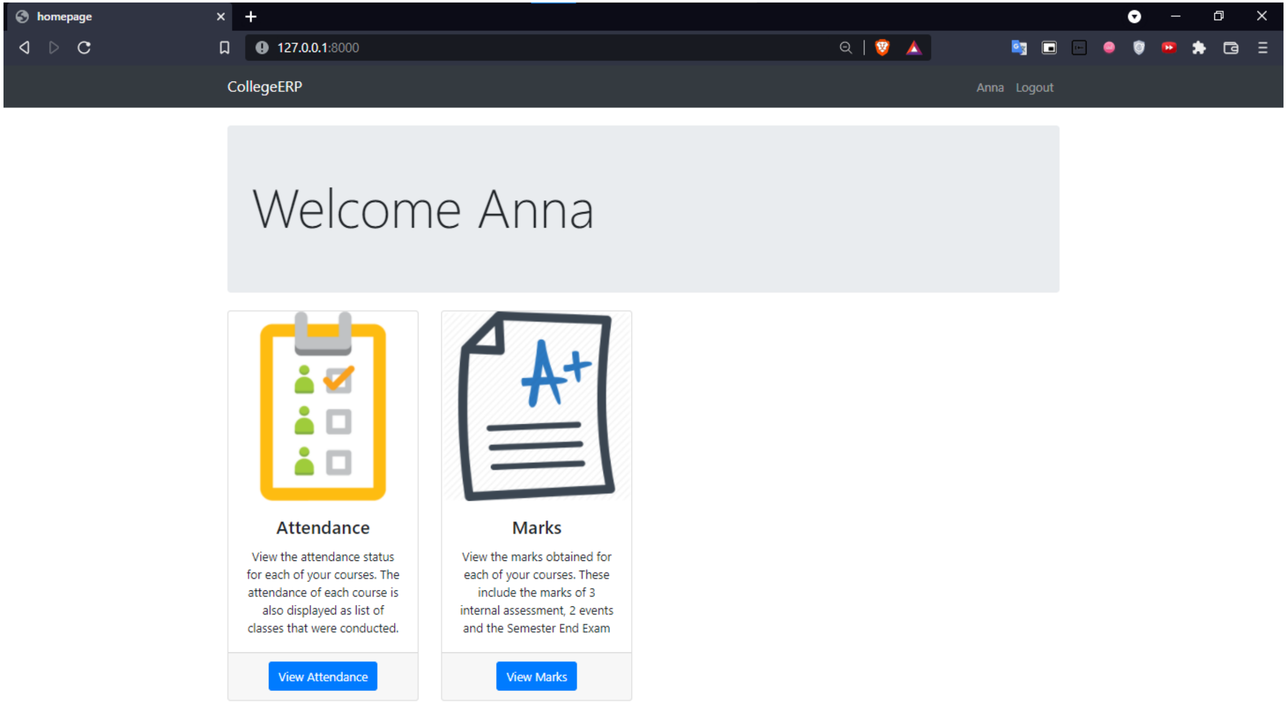
Name	Status	
Internal test 1	Not Marked	Enter Marks
Internal test 2	Not Marked	Enter Marks
Internal test 3	Not Marked	Enter Marks
Event 1	Not Marked	Enter Marks
Event 2	Not Marked	Enter Marks
Semester End Exam	Not Marked	Enter Marks



127.0.0.1:8000/info/teacher/175/marks_entry/

CollegeERP

Student Name	Total Marks	Enter Marks
Dustin	20	0
Maci	20	0
Axel	20	0
Shanice	20	0
Hugo	20	0
Rian	20	0
Jensen	20	0
Sapphire	20	0
Paisley	20	0
Kirstie	20	0



127.0.0.1:8000

CollegeERP

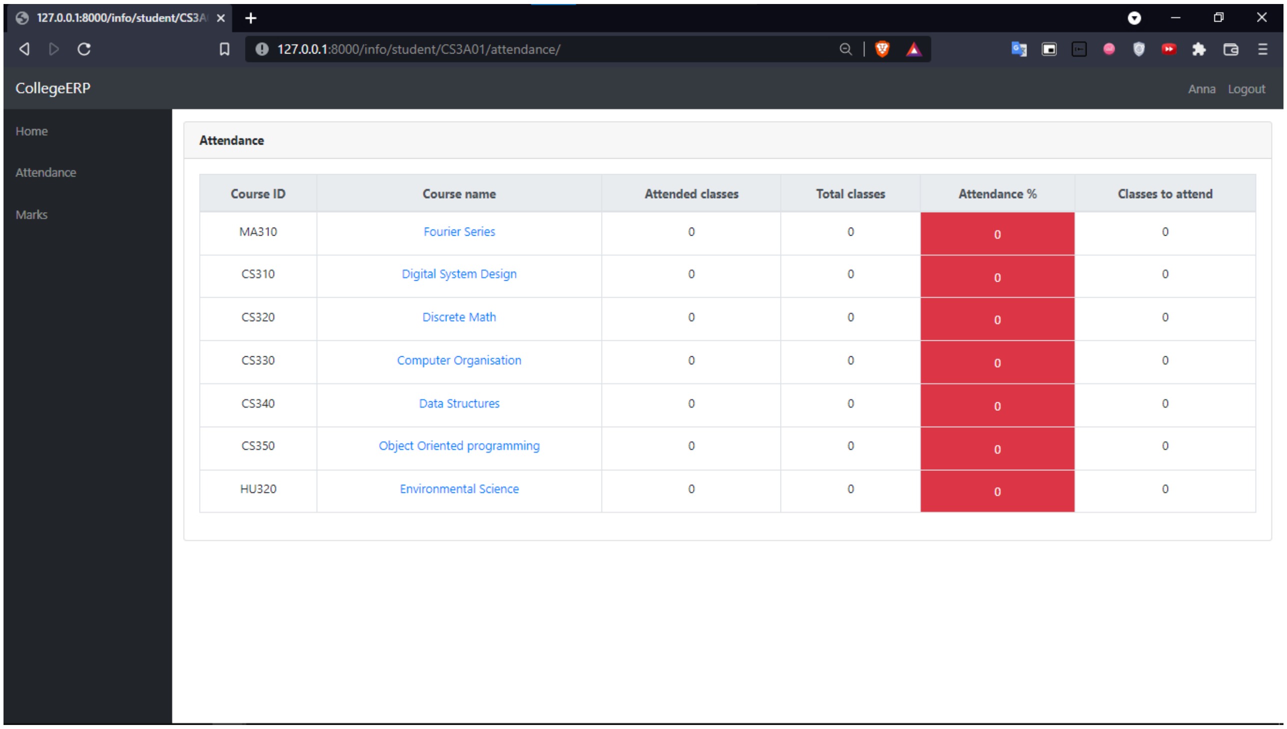
Welcome Anna

Attendance

View the attendance status for each of your courses. The attendance of each course is also displayed as list of classes that were conducted.

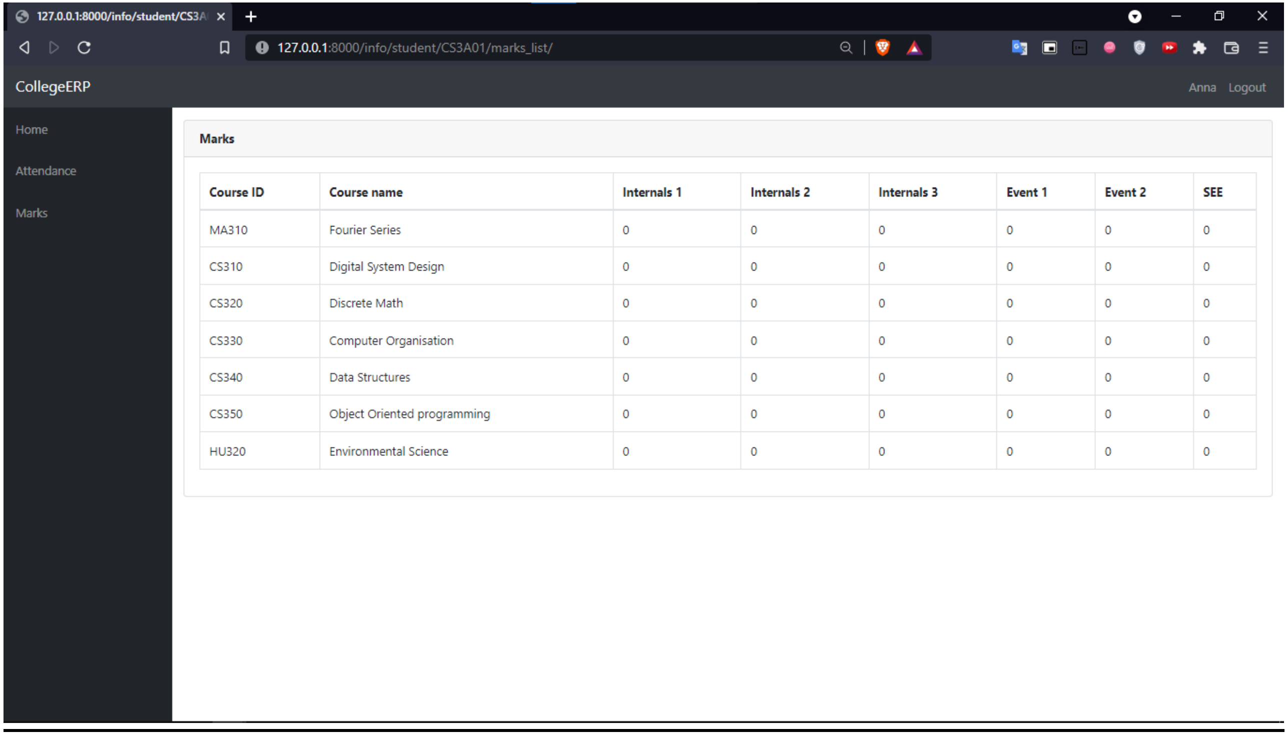
Marks

View the marks obtained for each of your courses. These include the marks of 3 internal assessment, 2 events and the Semester End Exam


127.0.0.1:8000/info/student/CS3A
127.0.0.1:8000/info/student/CS3A01/attendance/
CollegeERP
Anna
Logout
Home
Attendance
Marks

Attendance

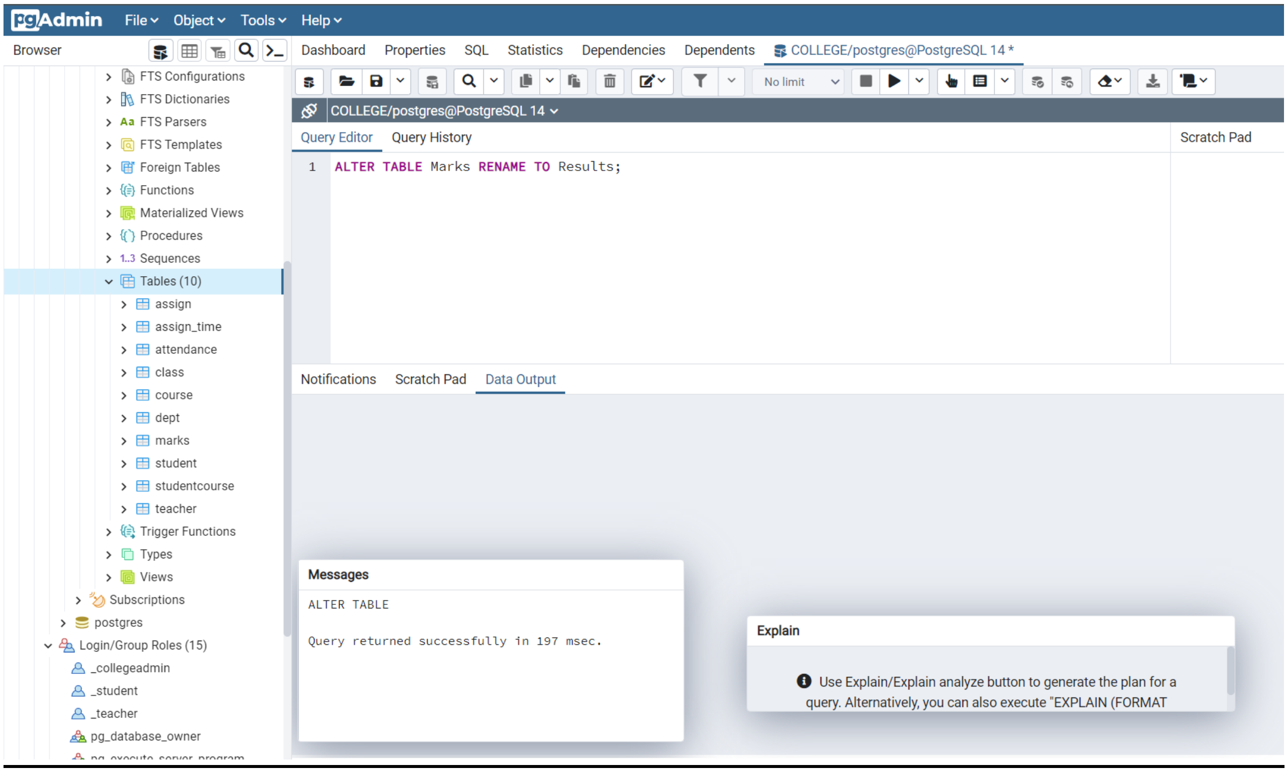
Course ID	Course name	Attended classes	Total classes	Attendance %	Classes to attend
MA310	Fourier Series	0	0	0	0
CS310	Digital System Design	0	0	0	0
CS320	Discrete Math	0	0	0	0
CS330	Computer Organisation	0	0	0	0
CS340	Data Structures	0	0	0	0
CS350	Object Oriented programming	0	0	0	0
HU320	Environmental Science	0	0	0	0


127.0.0.1:8000/info/student/CS3A
127.0.0.1:8000/info/student/CS3A01/marks_list/
CollegeERP
Anna
Logout
Home
Attendance
Marks

Marks

Course ID	Course name	Internals 1	Internals 2	Internals 3	Event 1	Event 2	SEE
MA310	Fourier Series	0	0	0	0	0	0
CS310	Digital System Design	0	0	0	0	0	0
CS320	Discrete Math	0	0	0	0	0	0
CS330	Computer Organisation	0	0	0	0	0	0
CS340	Data Structures	0	0	0	0	0	0
CS350	Object Oriented programming	0	0	0	0	0	0
HU320	Environmental Science	0	0	0	0	0	0

Screenshots for schema change statements:



```
ALTER TABLE Marks RENAME TO Results;
```

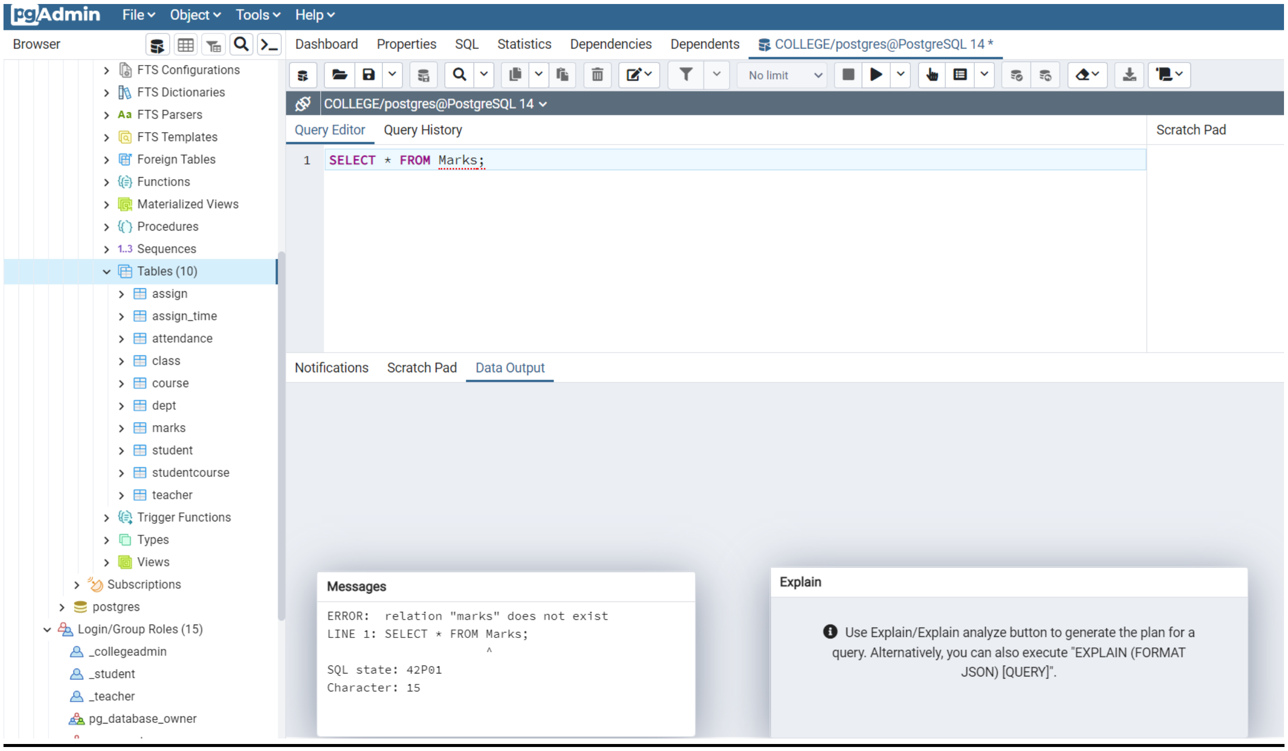
Messages

```
ALTER TABLE

Query returned successfully in 197 msec.
```

Explain

Use Explain/Explain analyze button to generate the plan for a query. Alternatively, you can also execute "EXPLAIN (FORMAT



```
SELECT * FROM Marks;
```

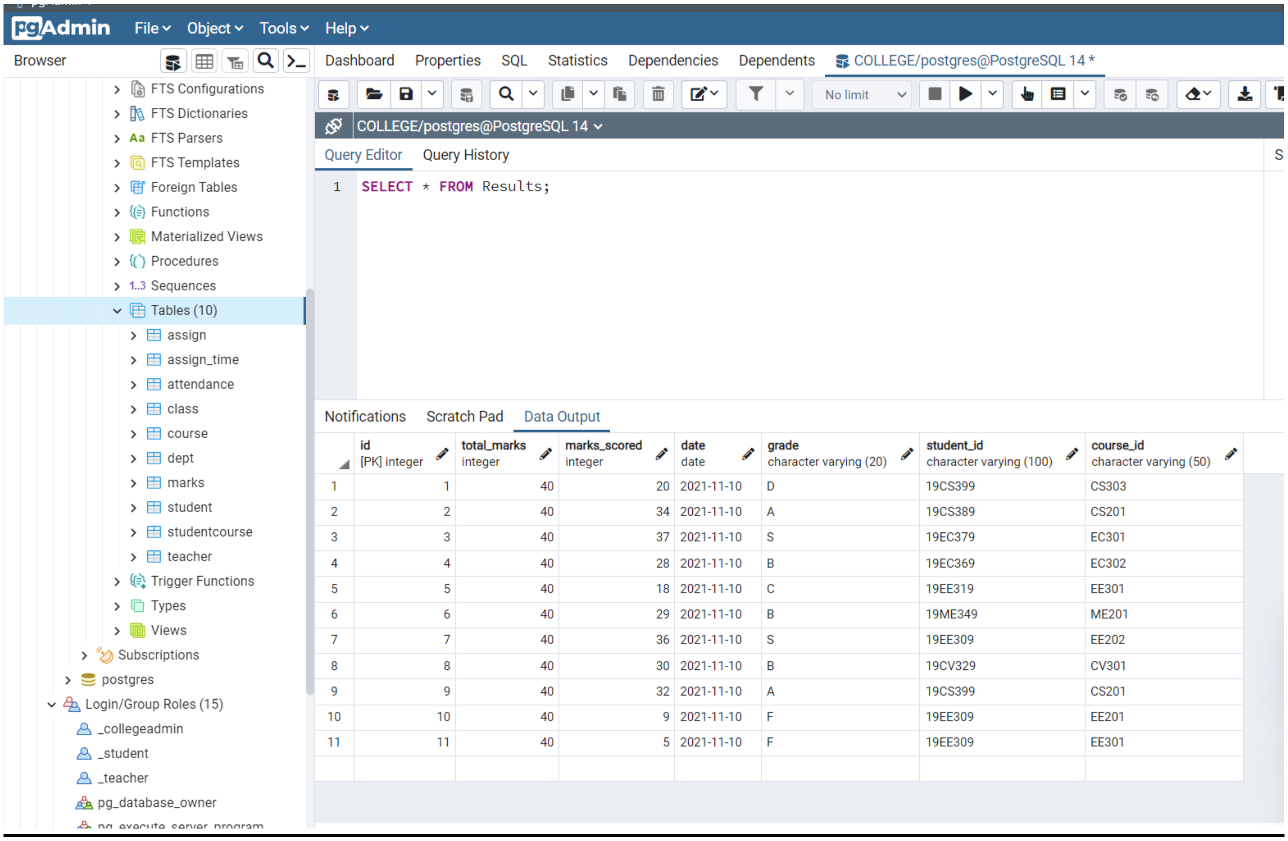
Messages

```
ERROR:  relation "marks" does not exist
LINE 1: SELECT * FROM Marks;
                      ^

SQL state: 42P01
Character: 15
```

Explain

Use Explain/Explain analyze button to generate the plan for a query. Alternatively, you can also execute "EXPLAIN (FORMAT JSON) [QUERY]".



```
SELECT * FROM Results;
```

	id [PK] integer	total_marks integer	marks_scored integer	date date	grade character varying (20)	student_id character varying (100)	course_id character varying (50)
1	1	40	20	2021-11-10	D	19CS399	CS303
2	2	40	34	2021-11-10	A	19CS389	CS201
3	3	40	37	2021-11-10	S	19EC379	EC301
4	4	40	28	2021-11-10	B	19EC369	EC302
5	5	40	18	2021-11-10	C	19EE319	EE301
6	6	40	29	2021-11-10	B	19ME349	ME201
7	7	40	36	2021-11-10	S	19EE309	EE202
8	8	40	30	2021-11-10	B	19CV329	CV301
9	9	40	32	2021-11-10	A	19CS399	CS201
10	10	40	9	2021-11-10	F	19EE309	EE201
11	11	40	5	2021-11-10	F	19EE309	EE301

Since 'Marks' contain the score as well as the grade, we can change 'Marks' relation name to 'Results'.

2. Choice of NoSQL database for migration:

If we have to migrate to a No-SQL database, we would choose to go with MongoDB as it is a scalable and versatile NoSQL document database. Rather than tables, MongoDB stores data in collections of Binary JSON documents. Both organised and unstructured data can be handled by MongoDB. This allows us to start constructing your project without having to define the schema first.

MongoDB, with its flexible schema, is the right answer if our data model and schema change regularly in an Agile setting. With MongoDB, we can instantly change the format of documents without having to go through application code to update queries and table references. It also allows us to quickly establish and scale MongoDB clusters.

Although flexible data models and schemas are important, scalability should also be considered. As a result, MongoDB is the best solution for scalability and efficient querying. Both vertically and horizontally, MongoDB scales effectively.

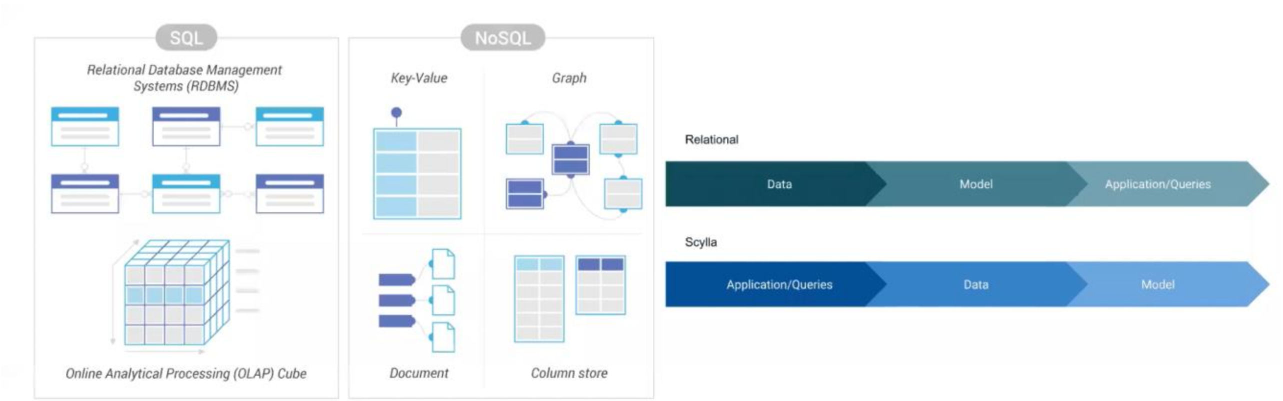
3.Migrating to NoSql database:

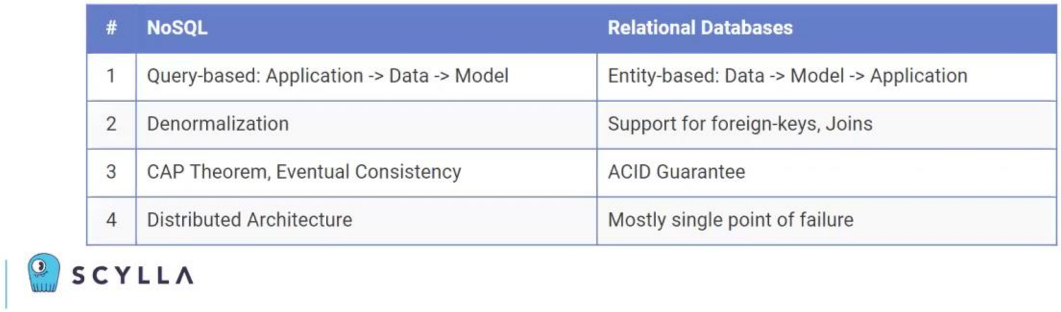
SQL is a common query language used when dealing with an RDBMS. Using an RDBMS is a choice for storing transactional data or records where the ACID (Atomicity, Consistency, Isolation, Durability) properties of transactions must be provided by an underlying database. An RDBMS is also a choice where the security and accessibility of data are of utmost importance. Typical use cases are financial records, financial transactions, OLTP, ERP, CRM systems, e-commerce applications, etc.

Since we had an ERP system our first choice for a database was a RDBMS. However NoSql too has its benefits.

NoSQL (sometimes referred to as Not only SQL, non-SQL or non-relational) is a database that is suitable for managing data that is non-relational, i.e. not structured in tabular format or have fixed data type formats and variables that do not possess tabular relationships. There are various types of NoSQL databases that exist, like key-value, document-based, column-based, and graph-based. When it comes to scalability and performance of unstructured data, NoSQL is the obvious choice.

A few notable factors important to note before migration:

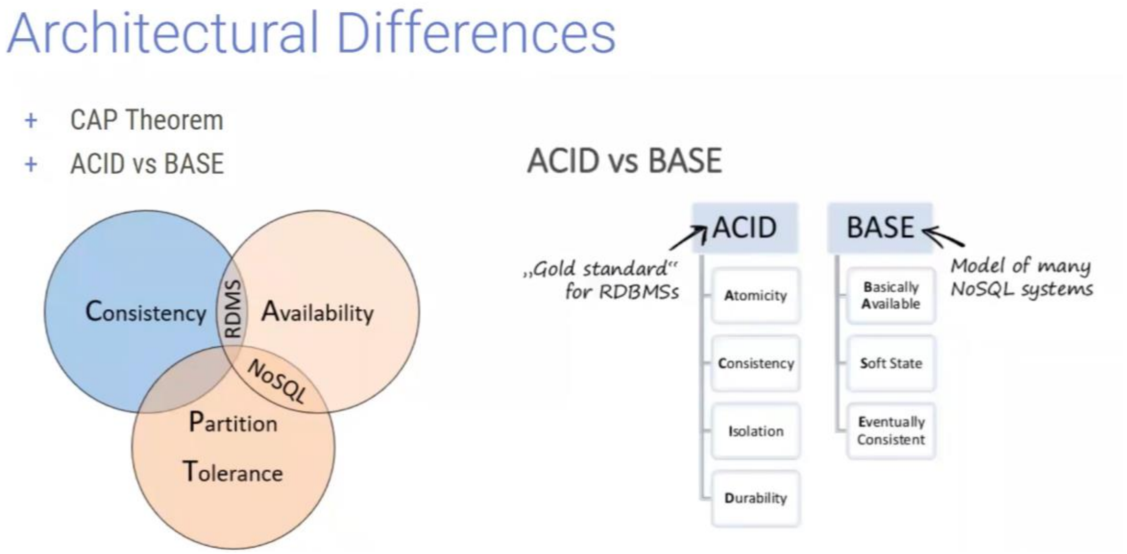




#	NoSQL	Relational Databases
1	Query-based: Application -> Data -> Model	Entity-based: Data -> Model -> Application
2	Denormalization	Support for foreign-keys, Joins
3	CAP Theorem, Eventual Consistency	ACID Guarantee
4	Distributed Architecture	Mostly single point of failure

SCYLLA

referenced from https://www.youtube.com/watch?v=i26RSbTyXE4&ab_channel=ScyllaDB



Architectural Differences

+ CAP Theorem
+ ACID vs BASE

Model transformation from relational databases to NoSQL databases or data modeling of NoSQL, in general, has become an important research topic with the growing adoption of NoSQL databases. Due to the lack of migration tools and the differences in the design principles and features of different NoSQL databases, model transformation and data migration are often done manually and left to the expertise of the database administrators. Generally rebuilding the model in NoSql would be preferred, to ensure long term reliability as that of Postgres.

4.Business/Application changes/expansion -that might

lead to:

- Schema changes

By using Existing System accessing information from files is a difficult task and there is no quick and easy way to keep the records of students and staff. Lack of automation is also there in the Existing System. The aim of Our System is to reduce the workload and to save significant staff time. Our implemented ERP system solves the issue of attendance , marks scored, along with its visibility and maintenance. But there is further room for improvement. Significant improvement could include :

- A Timetable relation which ensures that there are no clashes amongst sections, courses and as such it is agreeable to the various teacher's schedules.
- Detailed report cards, could further expand the marks scored relation to include internal assessments, project marks,percentage/gpa calculation etc.
- Event notifications.
- Implementing cloud based architecture

Contributions

Shatakshi Mohan- Report writing and schema changes/alterations.
Siddhi Patil- Frontend designing and database connectivity
Sonit Pradhan- Frontend designing and database connectivity

Time Spent- 12 hrs