Design DB model for Guvi Zen class:

Entities:

1. users
2. courses
3. modules
4. tasks
5. queries
6. leaveApplication

Attributes:

1. users
    - user_id
    - first_name
    - last_name
    - email
    - password
    - role

2. courses
    - course_id
    - course_name
    - course_description
    - start_date
    - end_date
    - instructor_id

3. modules
    - module_id
    - module_name
    - module_description
    - course_id

4. tasks
    - task_id
    - module_id
    - task_description
    - due_date

5. queries
    - query_id
    - module_id
    - user_id
    - question
    - answer
    - status

6. leaveApplication
    - leave_id
    - user_id
    - start_date
    - end_date
    - reason
    - status

Relationships:

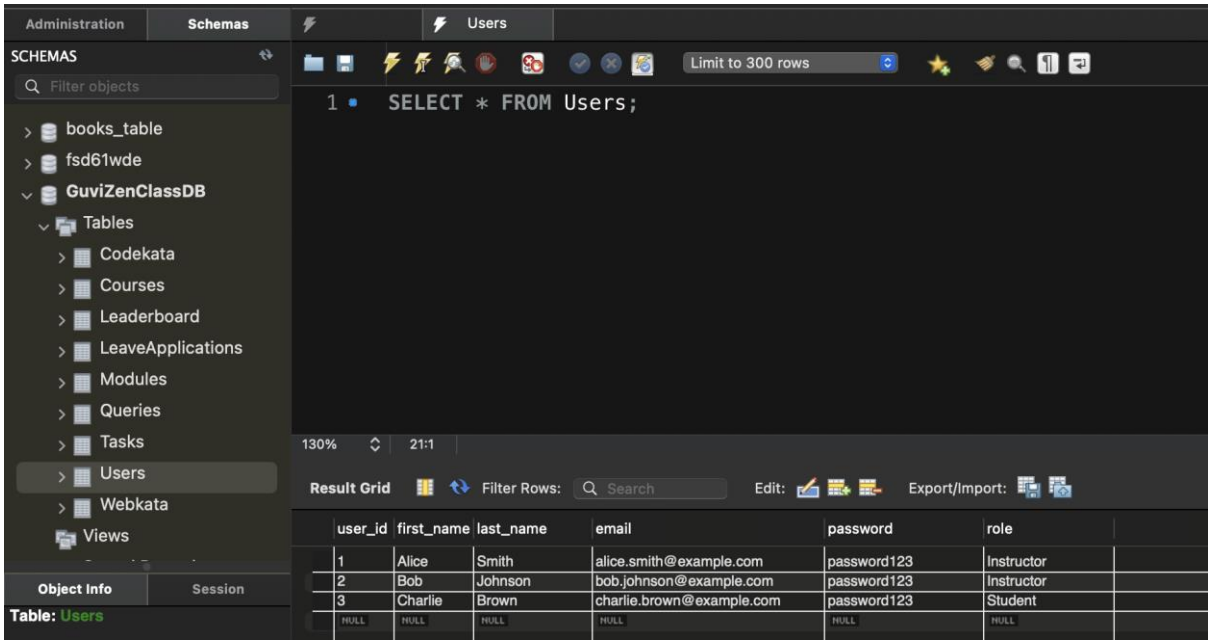
1. Users - Courses (One-to-Many)
2. Courses - Modules (One-to-Many)
3. Modules - Tasks (One-to-Many)
4. Users - Queries (One-to-Many)
5. Modules - Queries (One-to-Many)
6. Users - LeaveApplications (One-to-Many)

1.Users:

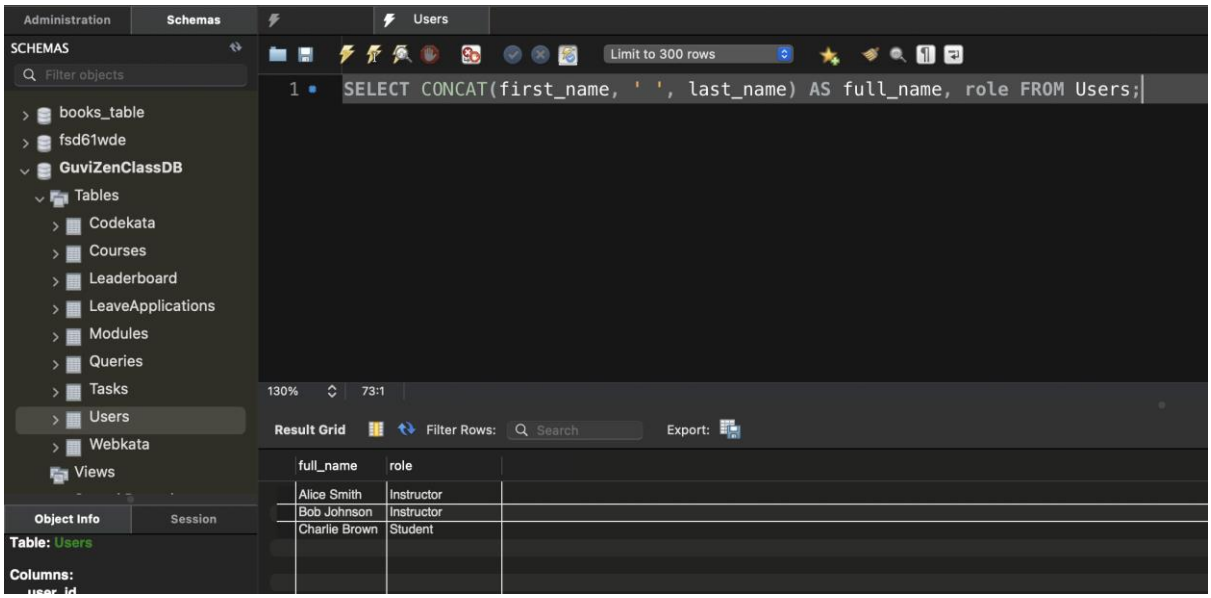
```sql
create table users (
  user_id int not null auto_increment,
  first_name varchar(50) not null,
  last_name varchar (50) not null,
  email varchar (100) not null unique,
  password varchar(255) not null,
  role varchar(20) not null,
  primary key (user_id)
);

insert into users (first_name, last_name, email, password, role)
values
('Alice', 'Smith', 'alice.smith@example.com', 'password123', 'Instructor'),
('Bob', 'Johnson', 'bob.johnson@example.com', 'password123', 'Instructor'),
('Charlie', 'Brown', 'charlie.brown@example.com', 'password123', 'Student');
```

Output:



List all users, showing their full name and role from users table ?

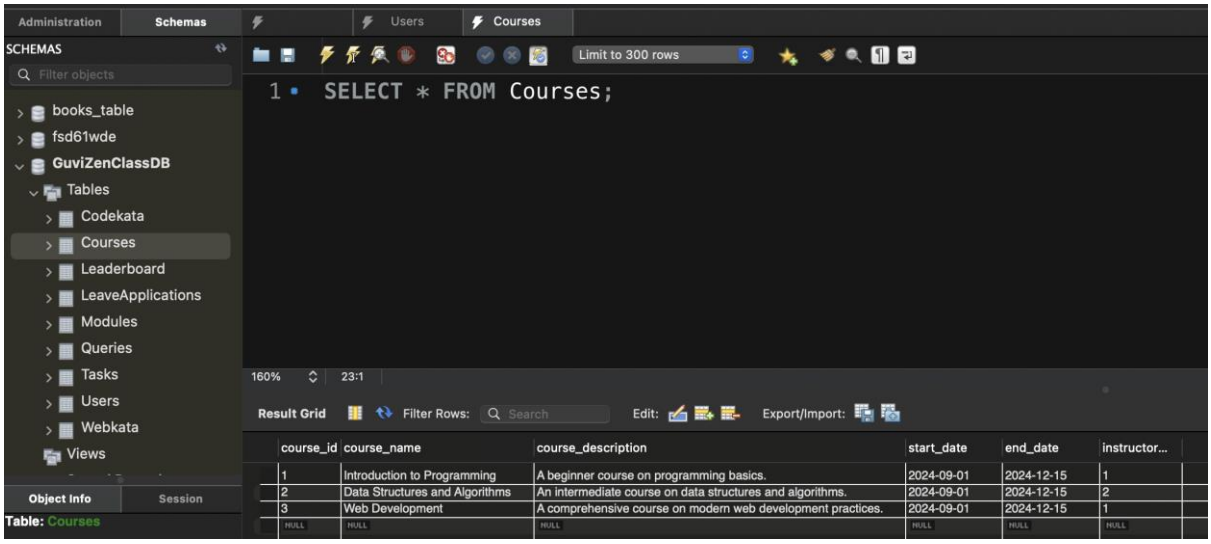


2.Courses:

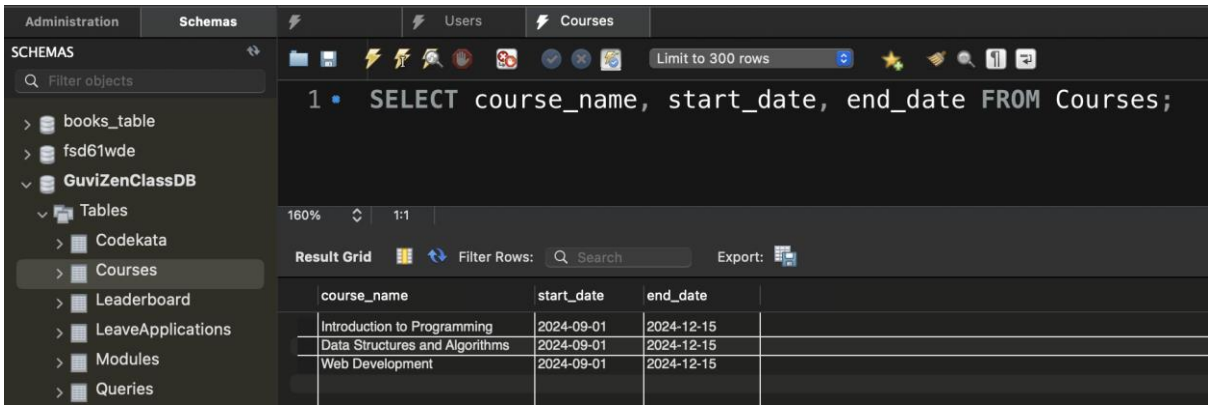
```sql
create table courses (
  course_id int not null auto_increment,
  course_name varchar(100) not null,
  course_description text,
  start_date date not null,
  end_date date not null,
  instructor_id int not null,
  primary key (course_id),
  foreign key (instructor_id) references users(user_id)
);

insert into courses (course_name, course_description, start_date, end_date, instructor_id)
values
('Introduction to Programming', 'A beginner course on programming basics.', '2024-09-01', '2024-12-15', 1),
('Data Structures and Algorithms', 'An intermediate course on data structures and algorithms.', '2024-09-01', '2024-12-15', 2),
('Web Development', 'A comprehensive course on modern web development practices.', '2024-09-01', '2024-12-15', 1);
```

Output:



List all courses along with start and end date ?

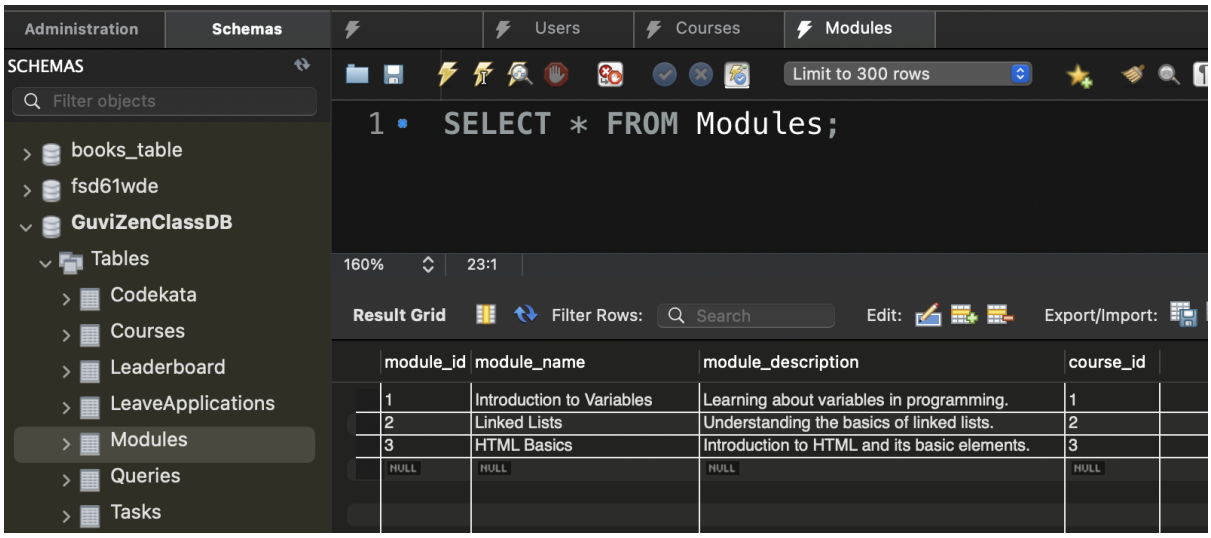


3.Modules:

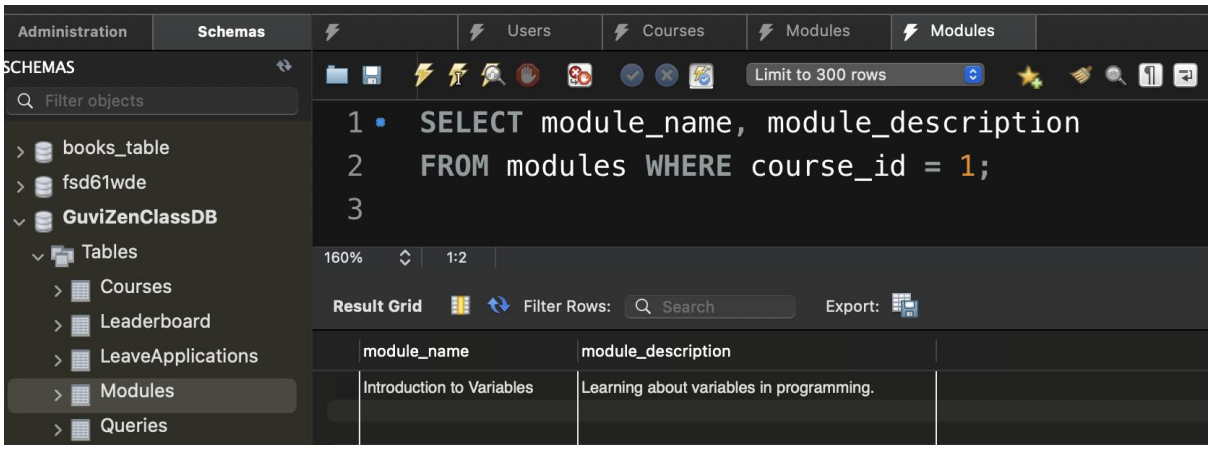
```sql
create table modules (
  module_id int not null auto_increment,
  module_name varchar(100) not null,
  module_description text,
  course_id int not null,
  primary key (module_id),
  foreign key (course_id) references courses(course_id)
);

insert into modules (module_name, module_description, course_id)
values
('Introduction to Variables', 'Learning about variables in programming.', 1),
('Linked Lists', 'Understanding the basics of linked lists.', 2),
('HTML Basics', 'Introduction to HTML and its basic elements.', 3);
```

Output:



Module along with description for specific course identify by the course_id:

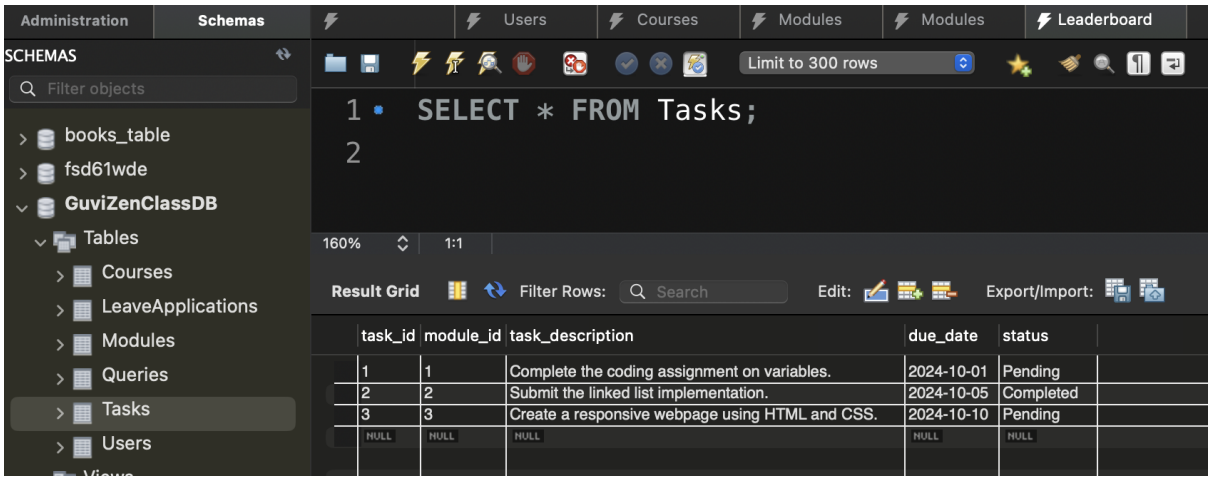


4.Tasks:

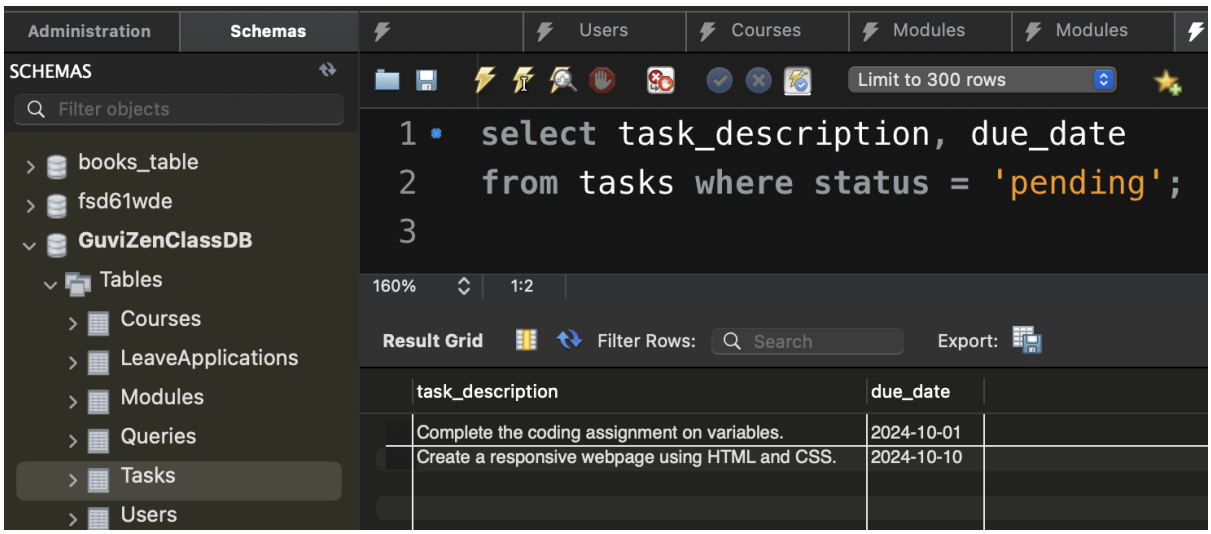
```sql
create table tasks (
  task_id int not null auto_increment,
  module_id int not null,
  task_description text,
  due_date date,
  status varchar(20),
  primary key (task_id),
  foreign key (module_id) references modules(module_id)
);

insert into tasks (module_id, task_description, due_date, status) values
(1, 'complete the coding assignment on variables.', '2024-10-01', 'pending'),
(2, 'submit the linked list implementation.', '2024-10-05', 'completed'),
(3, 'create a responsive webpage using html and css.', '2024-10-10', 'pending');
```

Output:



List all tasks that are currently pending along with their due dates:

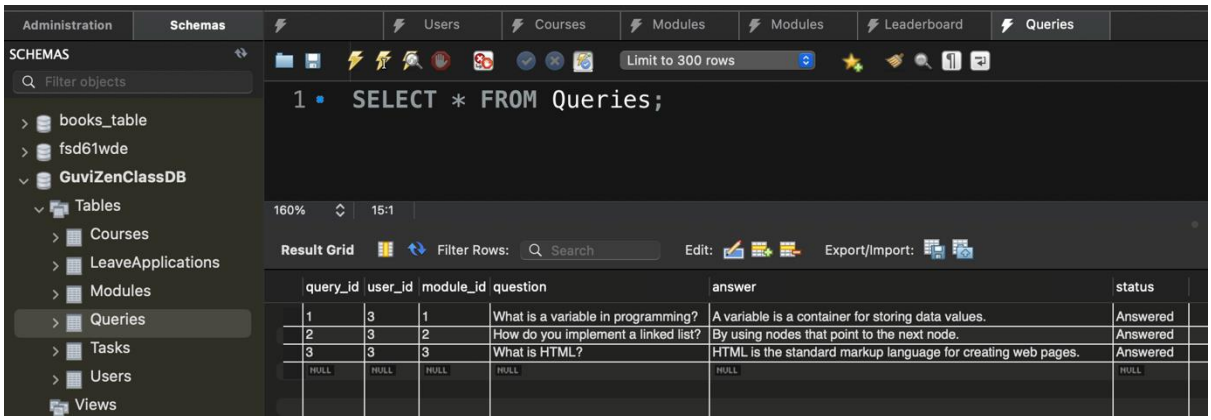


5.Queries:

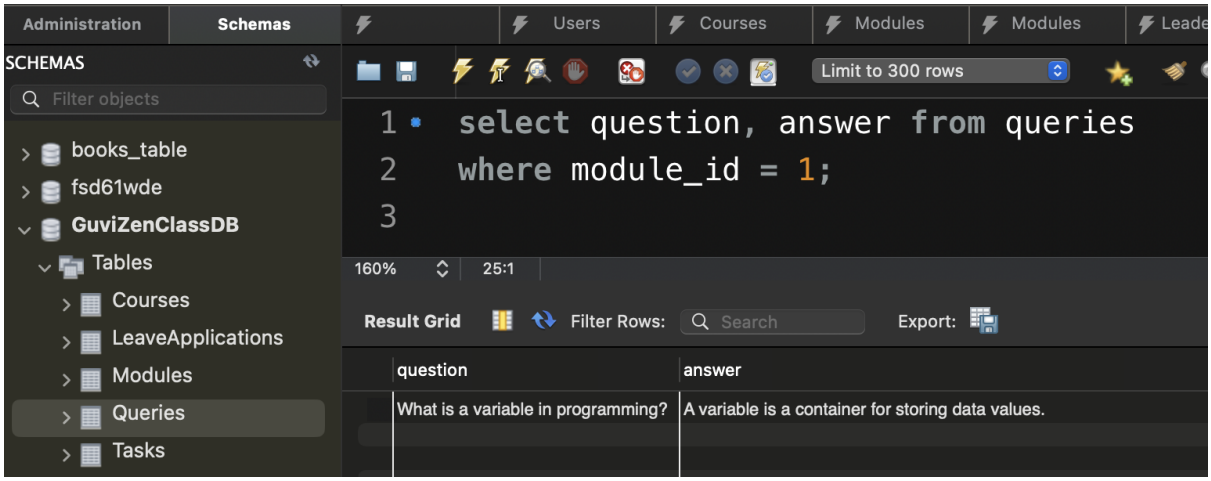
```sql
create table queries (
  query_id int not null auto_increment,
  user_id int not null,
  module_id int not null,
  question text,
  answer text,
  status varchar(20),
  primary key (query_id),
  foreign key (user_id) references users(user_id),
  foreign key (module_id) references modules(module_id)
);

insert into queries (user_id, module_id, question, answer, status) values
(3, 1, 'what is a variable in programming?', 'a variable is a container for storing data values.', 'answered'),
(3, 2, 'how do you implement a linked list?', 'by using nodes that point to the next node.', 'answered'),
(3, 3, 'what is html?', 'html is the standard markup language for creating web pages.', 'answered');
```

Output:



Retrieve all the questions and their corresponding answers for a specific module:

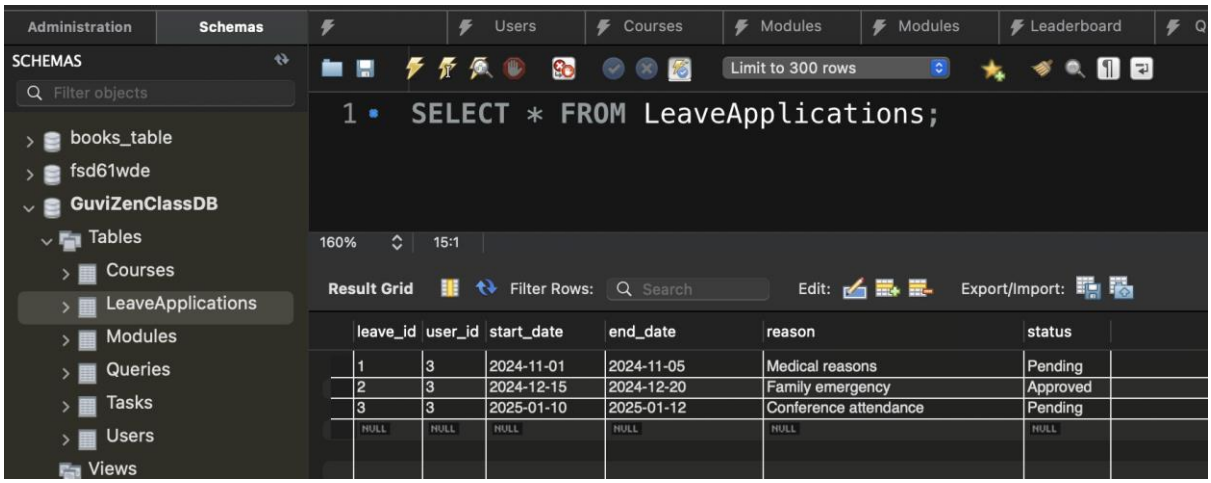


6.LeaveApplication:

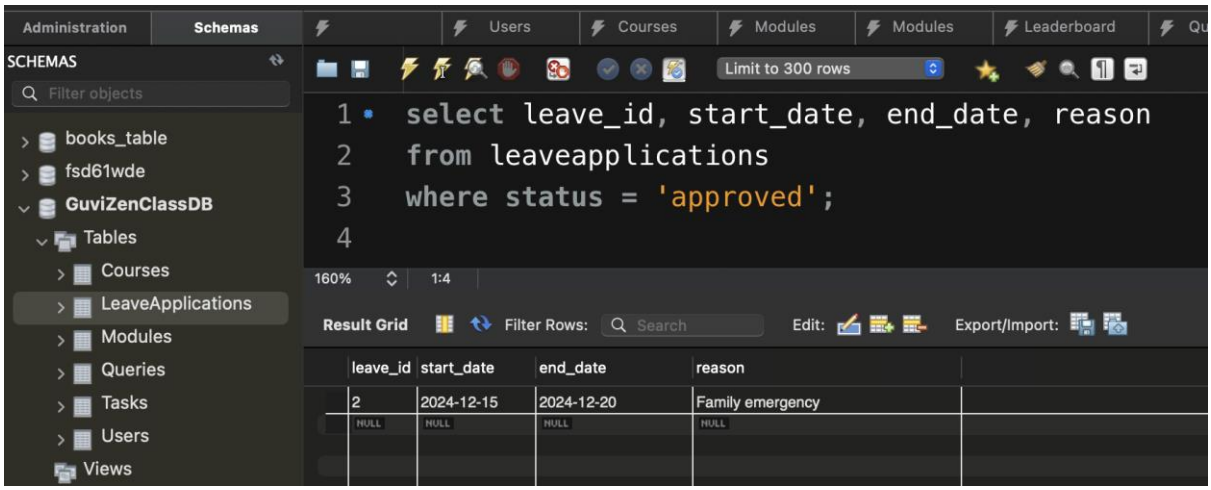
```sql
create table leaveapplications (
  leave_id int not null auto_increment,
  user_id int not null,
  start_date date not null,
  end_date date not null,
  reason text,
  status varchar(20) default 'pending',
  primary key (leave_id),
  foreign key (user_id) references users(user_id)
);

insert into leaveapplications (user_id, start_date, end_date, reason, status)
values
(3, '2024-11-01', '2024-11-05', 'medical reasons', 'pending'),
(3, '2024-12-15', '2024-12-20', 'family emergency', 'approved'),
(3, '2025-01-10', '2025-01-12', 'conference attendance', 'pending');
```

Output:



List all leave applications that have been approved:



Join Clause:

List all approved leave applications along with the full name of the user who applied for the leave?

```
use GuviZenClassDB;

select
  u.first_name,
  u.last_name,
  l.start_date,
  l.end_date
from
  leaveapplications l
join
  users u
on
  l.user_id = u.user_id
where
  l.status = 'approved';
```

first_name	last_name	start_date	end_date
Charlie	Brown	2024-12-15	2024-12-20

List all modules for each course, showing the course name and module name ?