**Design DB model for Guvi Zen class:**

**Users:**

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)

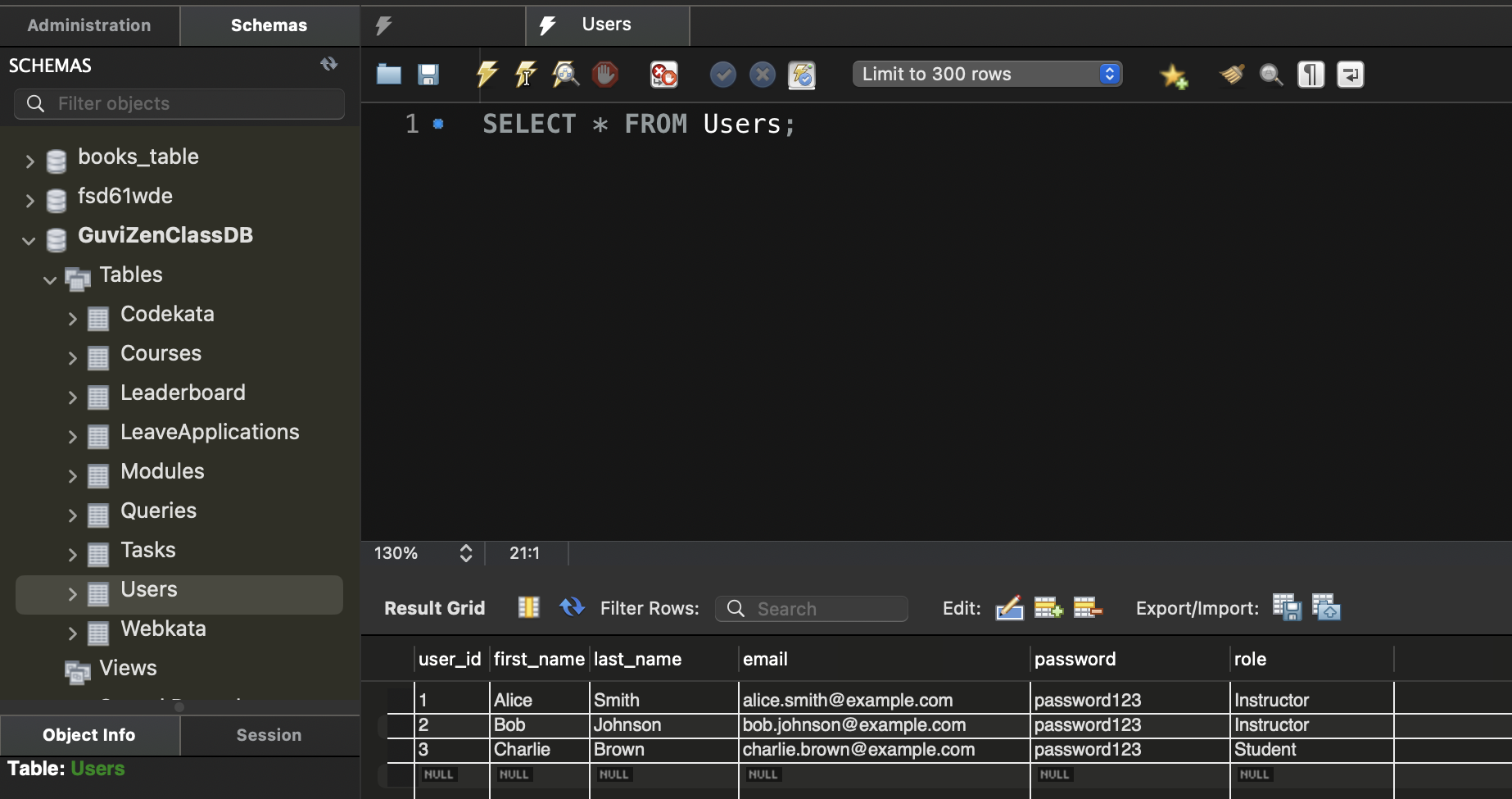
values

('Alice', 'Smith', 'alice.smith@example.com', 'password123', 'Instructor'),

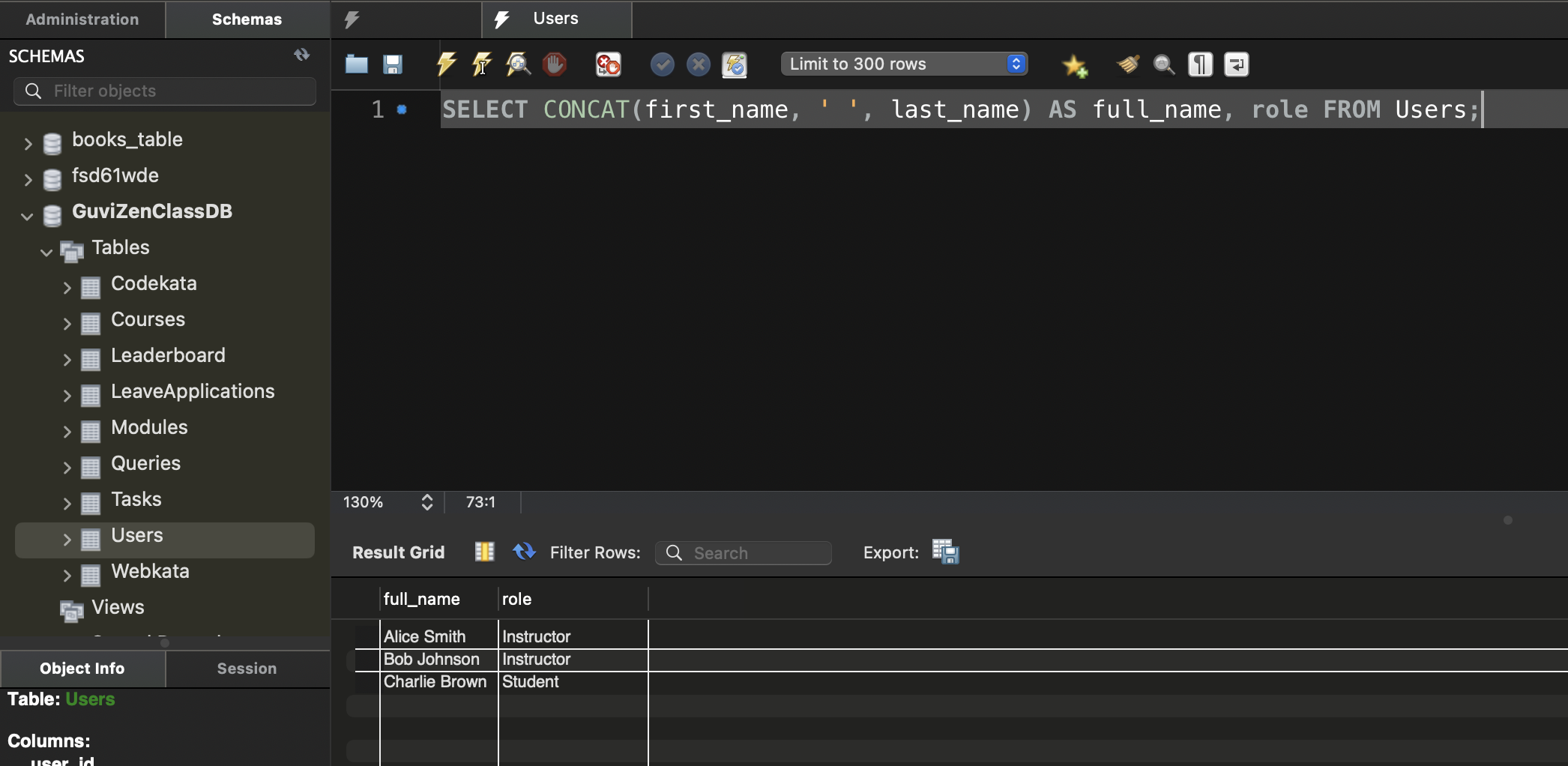
('Bob', 'Johnson', 'bob.johnson@example.com', 'password123', 'Instructor'),

('Charlie', 'Brown', 'charlie.brown@example.com', 'password123', 'Student');

*Output:*



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



**Courses:**

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)

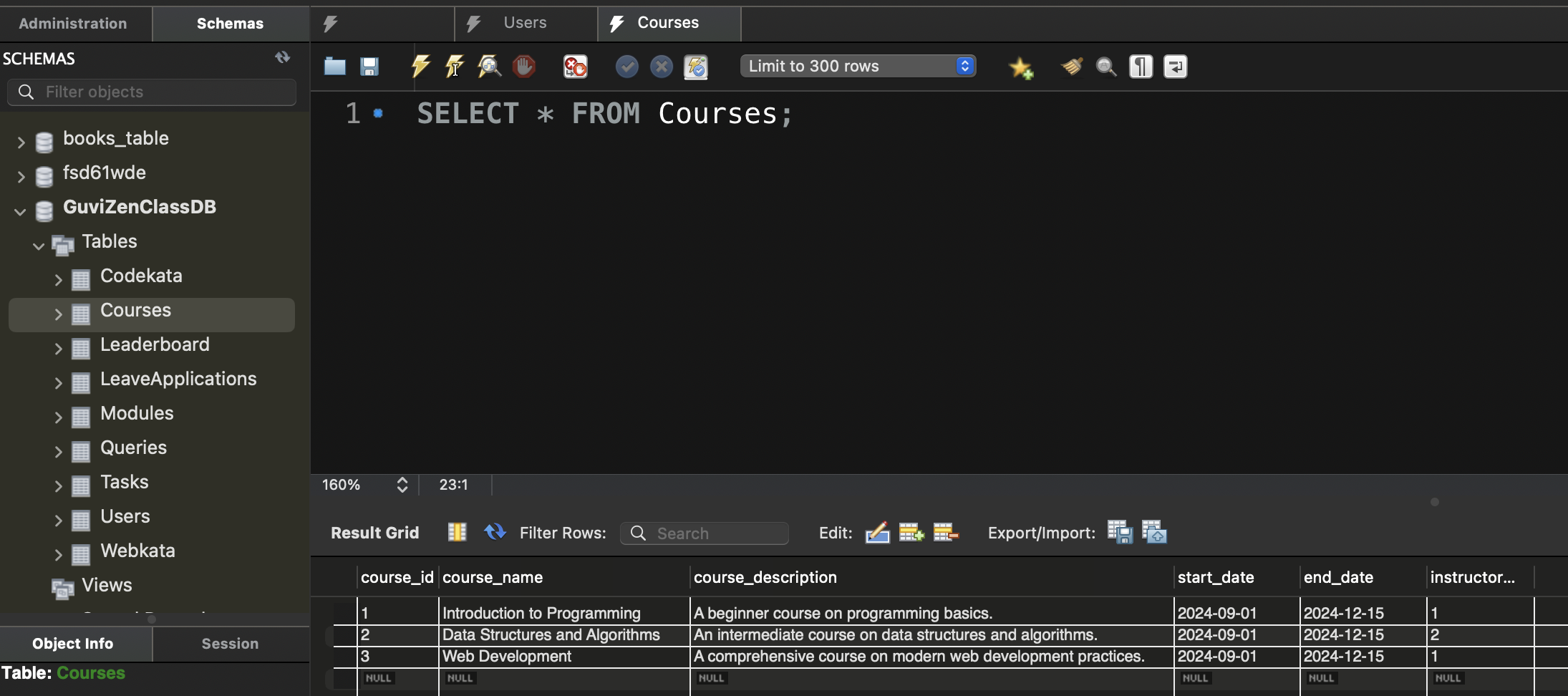
values

('Introduction to Programming', 'A beginner course on programming basics.', '2024-09-01', '2024-12-15', 1),

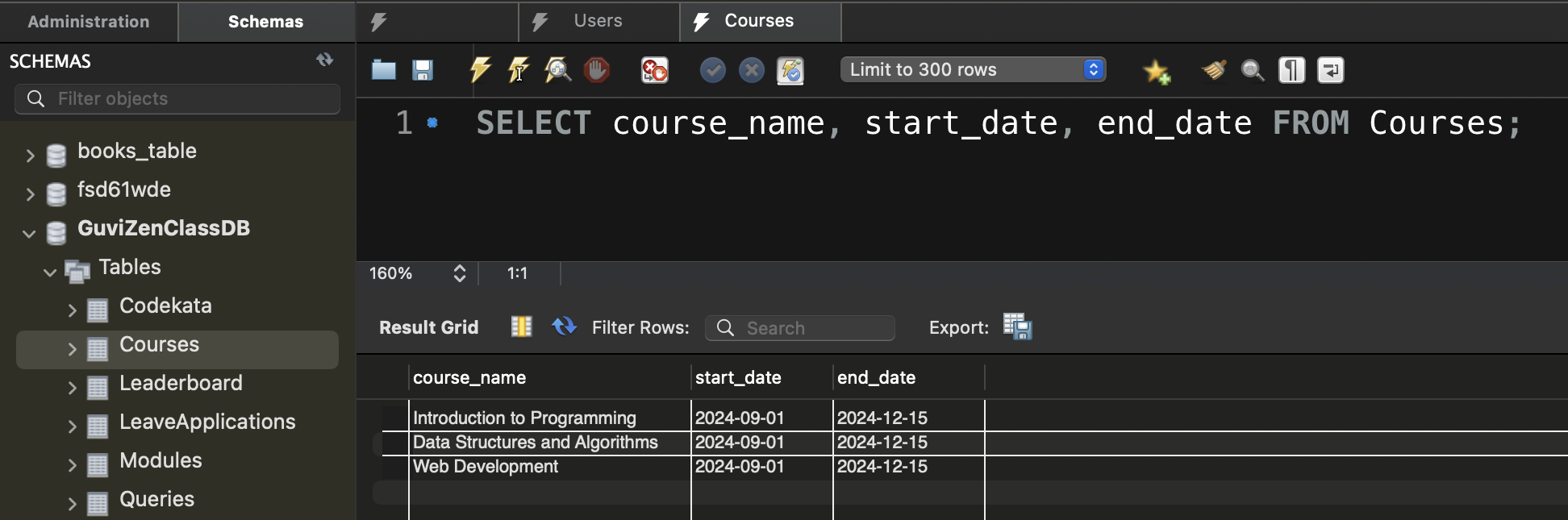
('Data Structures and Algorithms', 'An intermediate course on data structures and algorithms.', '2024-09-01', '2024-12-15', 2),

('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 ?*



**Modules:**

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)

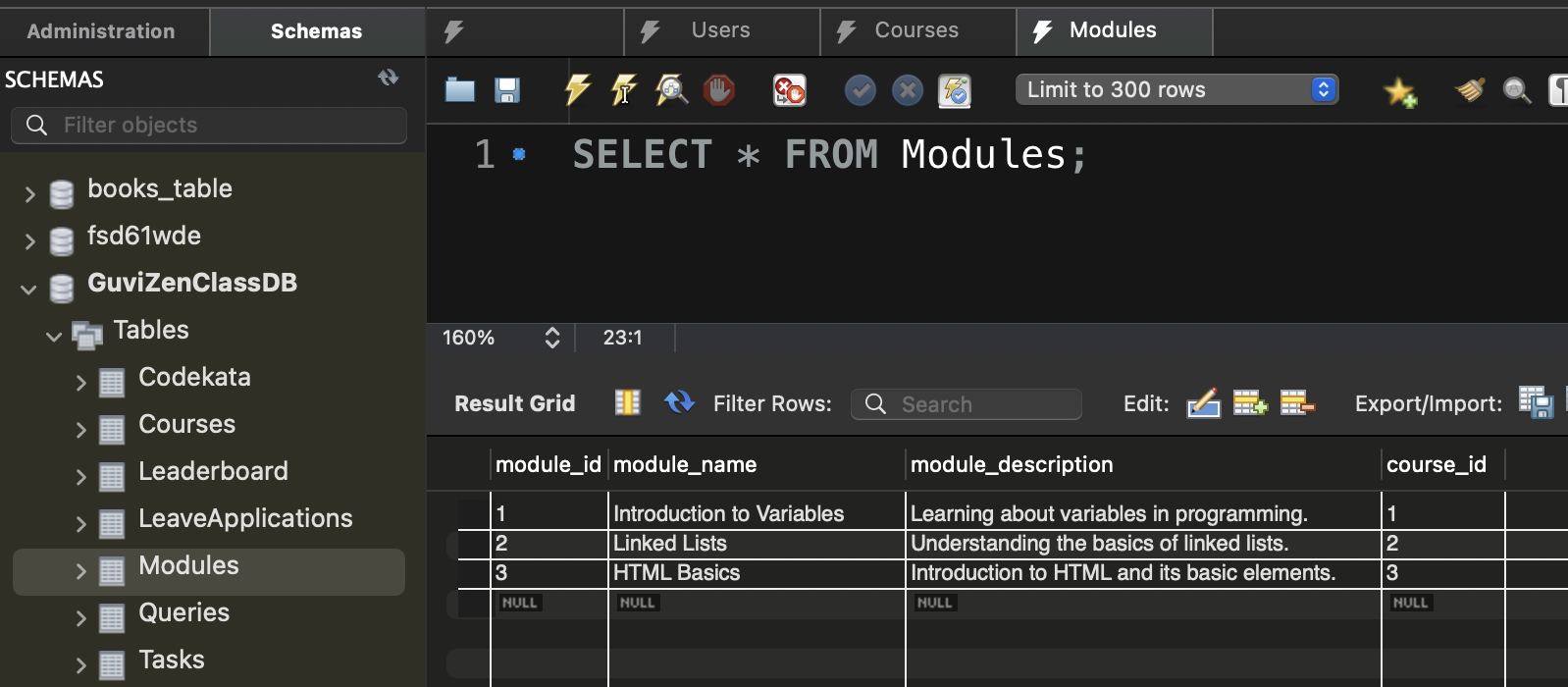
values

('Introduction to Variables', 'Learning about variables in programming.', 1),

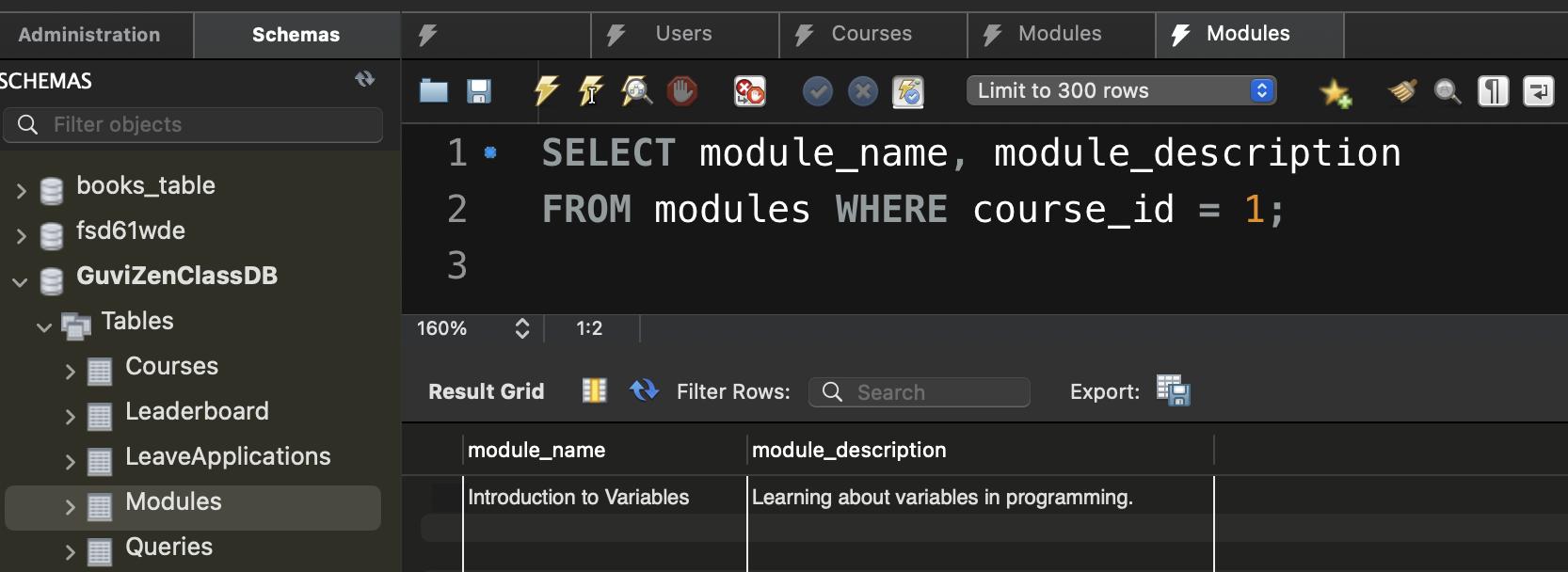
('Linked Lists', 'Understanding the basics of linked lists.', 2),

('HTML Basics', 'Introduction to HTML and its basic elements.', 3);

*Output:*



*Module along with description for specific course identify by the course\_id:*



**Tasks:**

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)

);

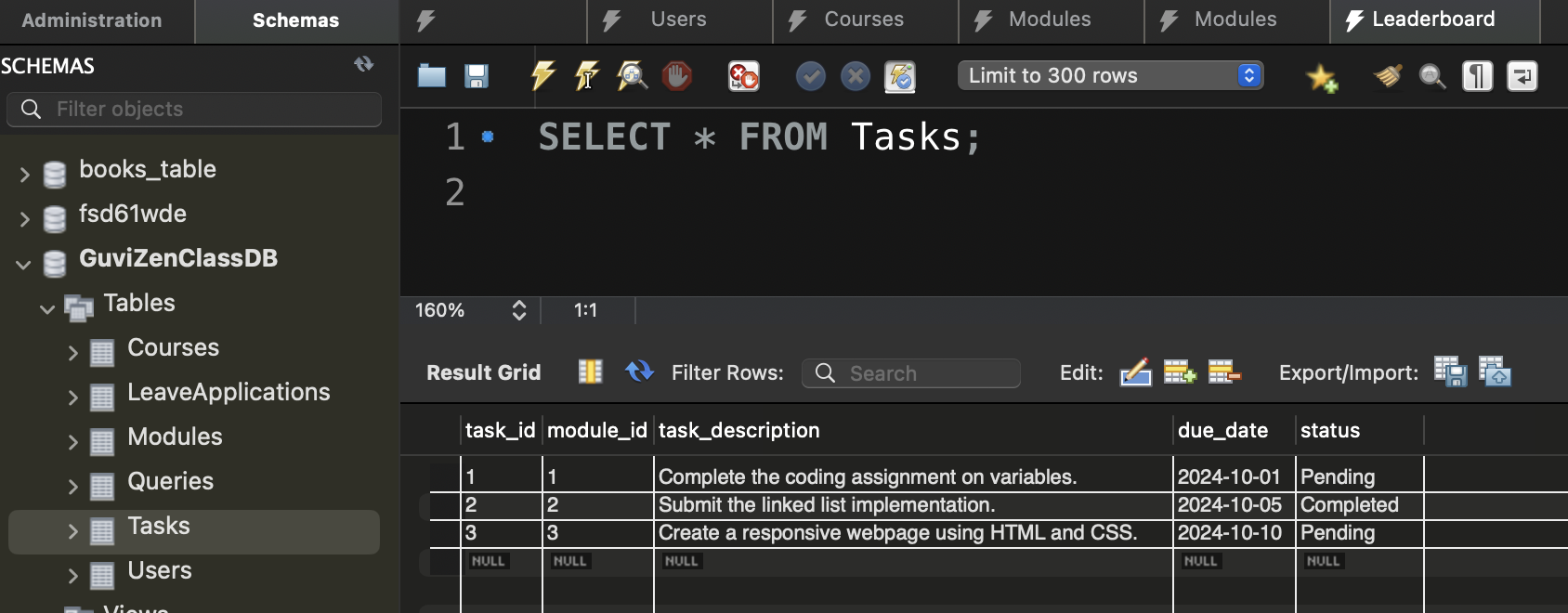
insert into tasks (module\_id, task\_description, due\_date, status) values

(1, 'complete the coding assignment on variables.', '2024-10-01', 'pending'),

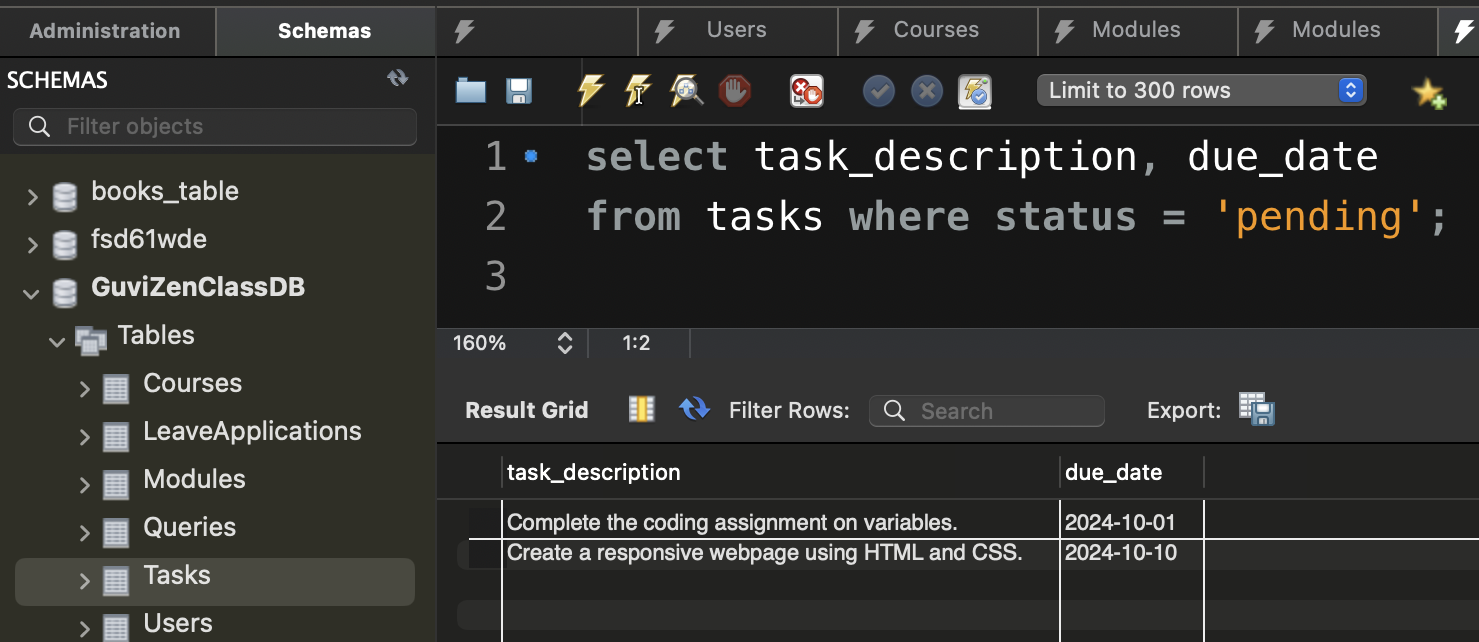
(2, 'submit the linked list implementation.', '2024-10-05', 'completed'),

(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:*



**Queries:**

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)

);

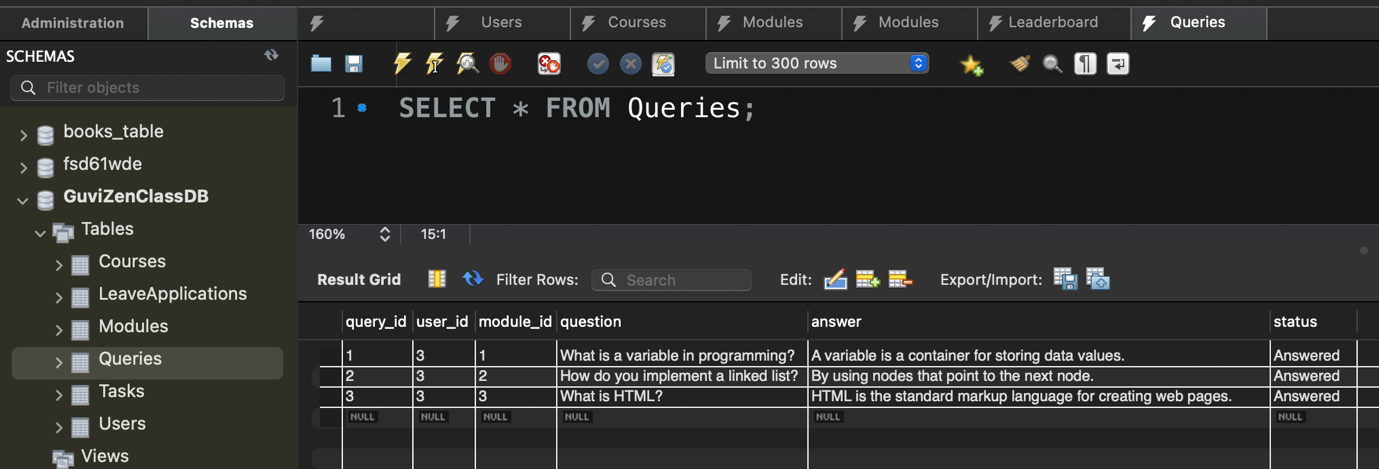
insert into queries (user\_id, module\_id, question, answer, status) values

(3, 1, 'what is a variable in programming?', 'a variable is a container for storing data values.', 'answered'),

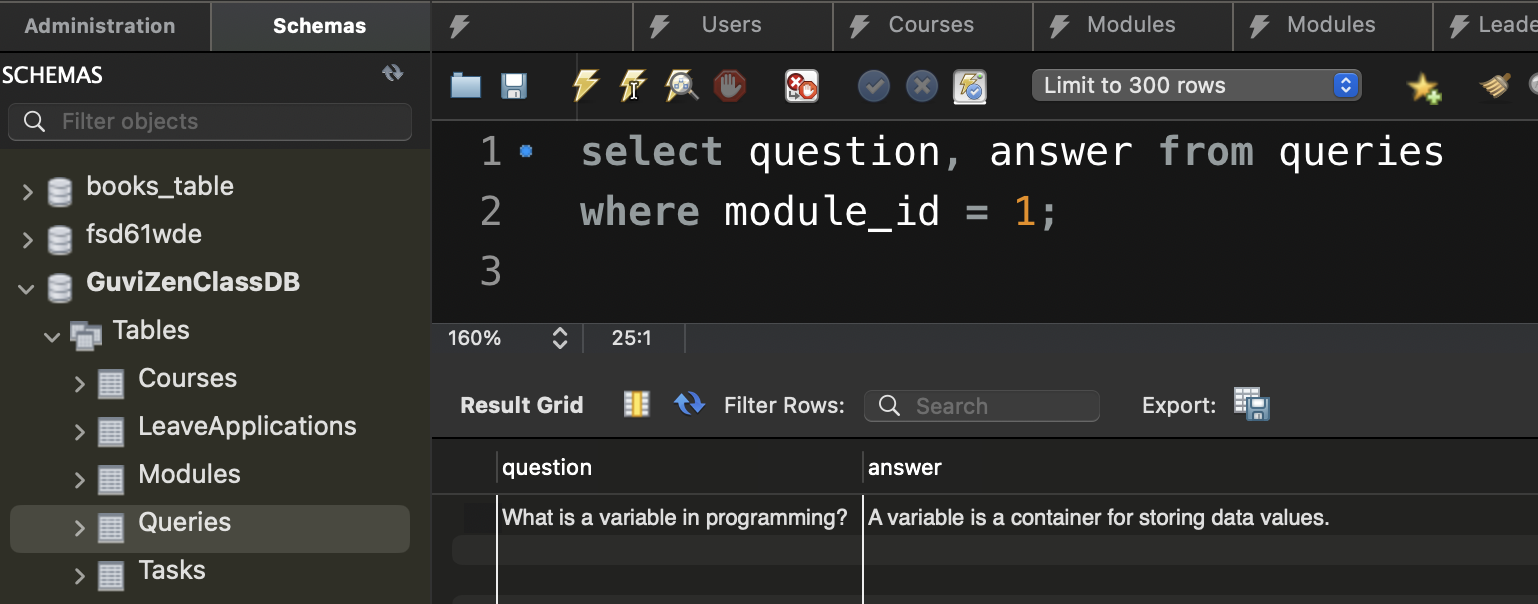
(3, 2, 'how do you implement a linked list?', 'by using nodes that point to the next node.', 'answered'),

(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:*



**LeaveApplication:**

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)

);

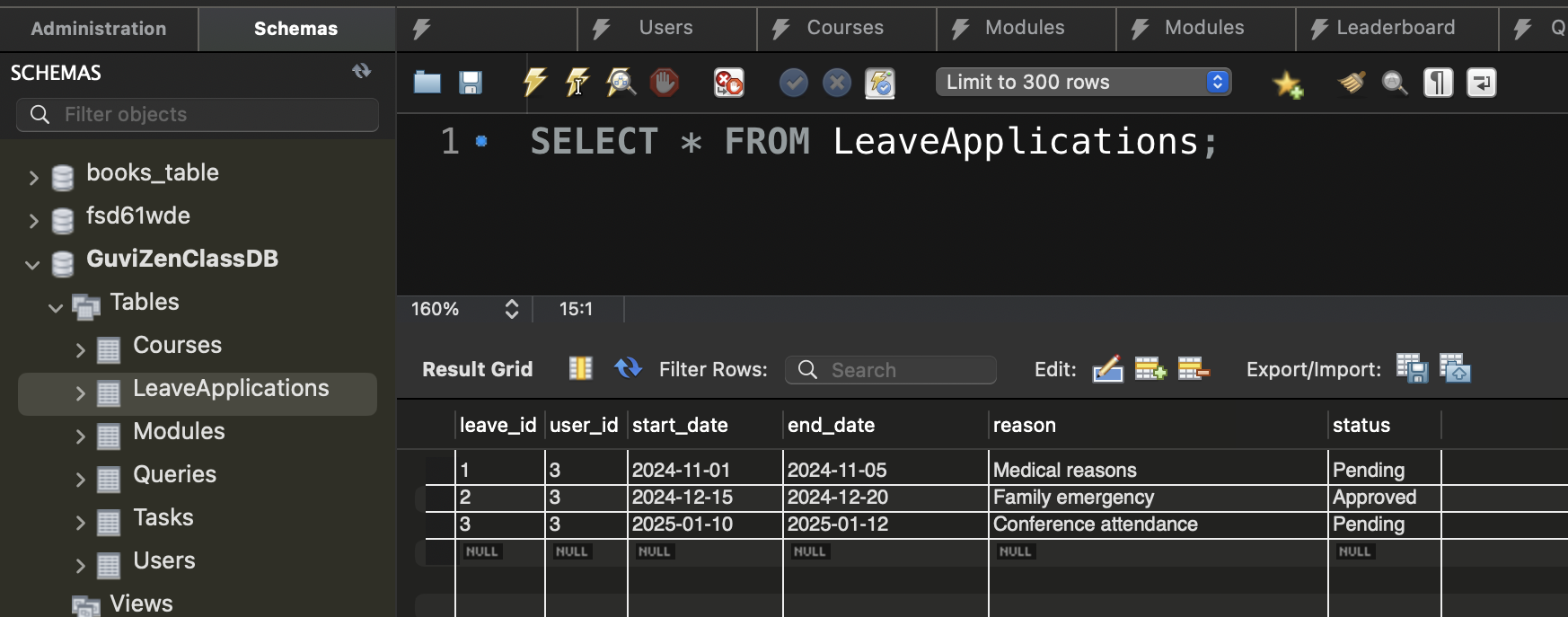
insert into leaveapplications (user\_id, start\_date, end\_date, reason, status) values

(3, '2024-11-01', '2024-11-05', 'medical reasons', 'pending'),

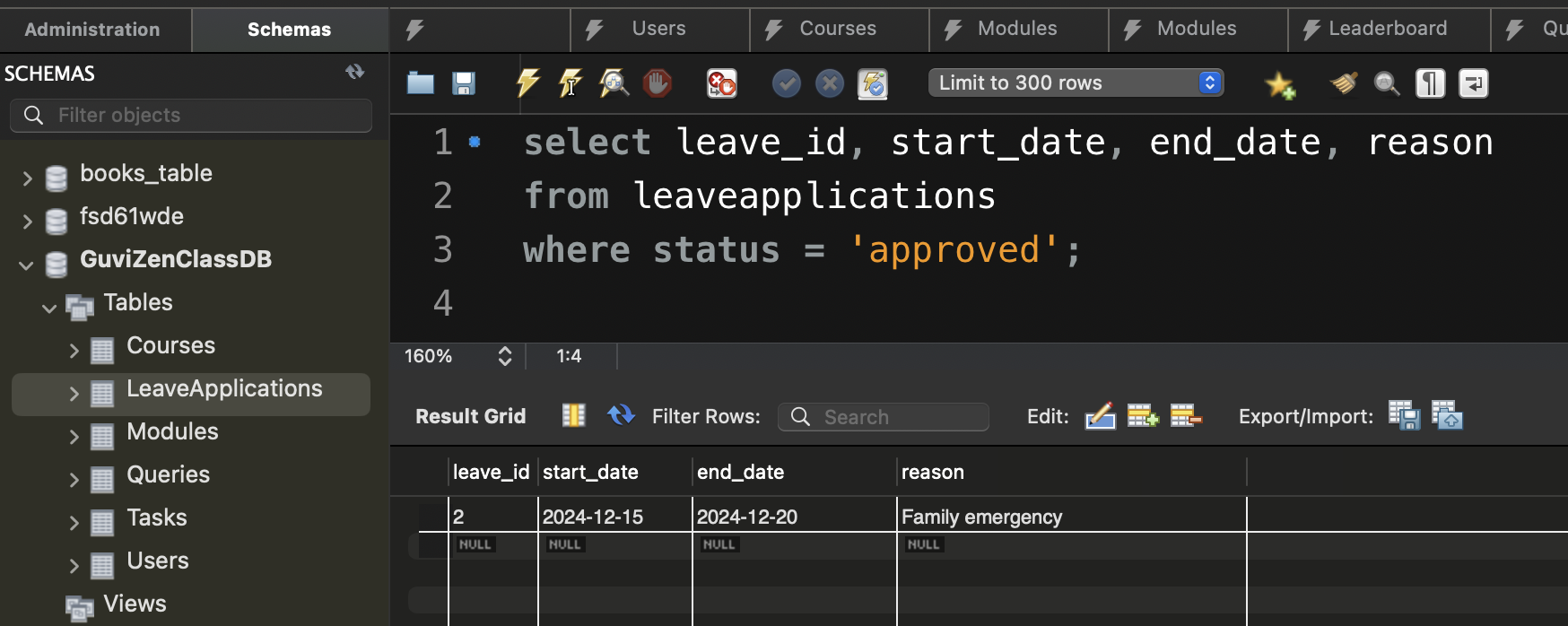
(3, '2024-12-15', '2024-12-20', 'family emergency', 'approved'),

(3, '2025-01-10', '2025-01-12', 'conference attendance', 'pending');

*Output:*

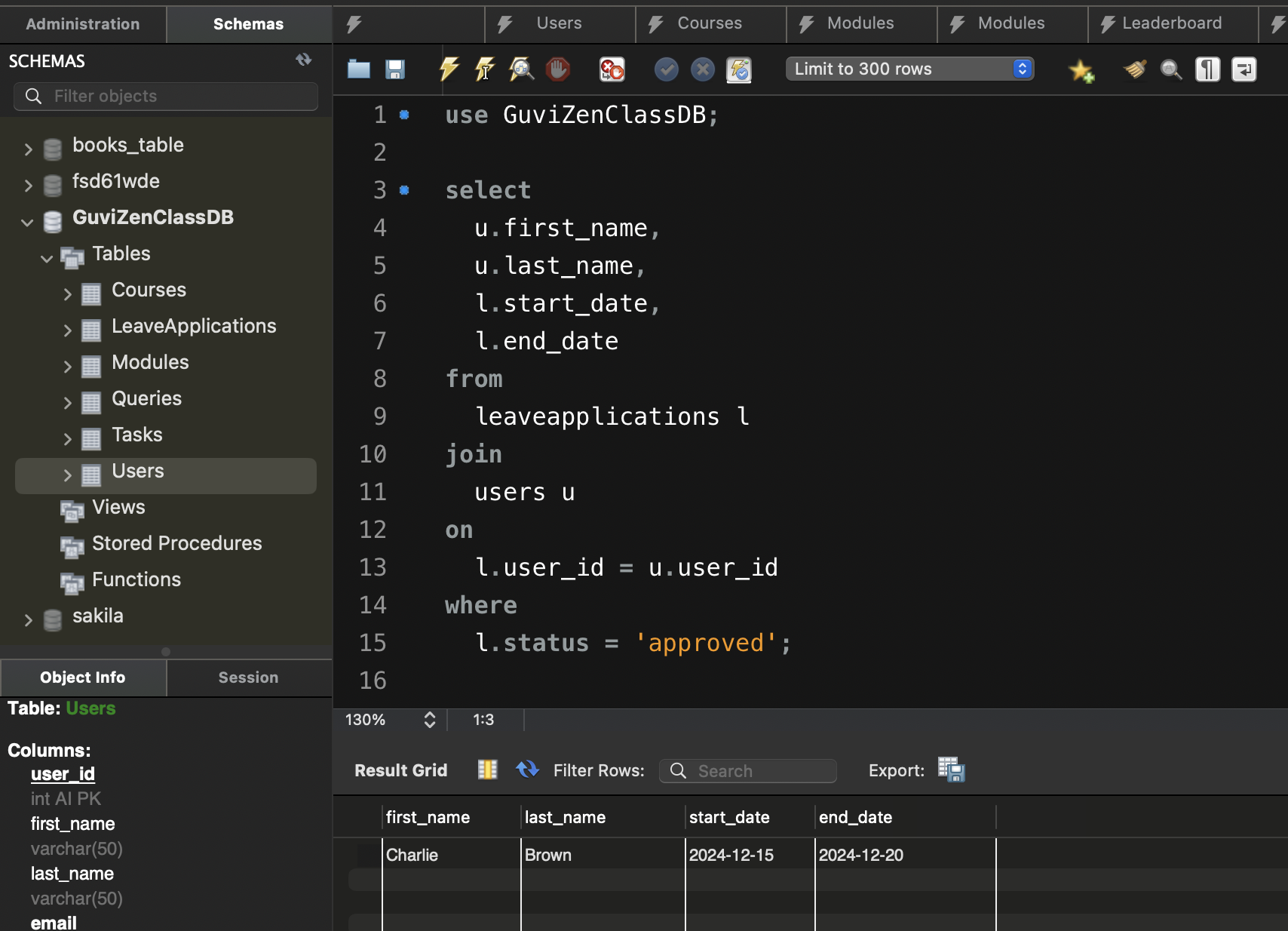


*List all leave applications that have been approved:*



// Join Clause

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



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

