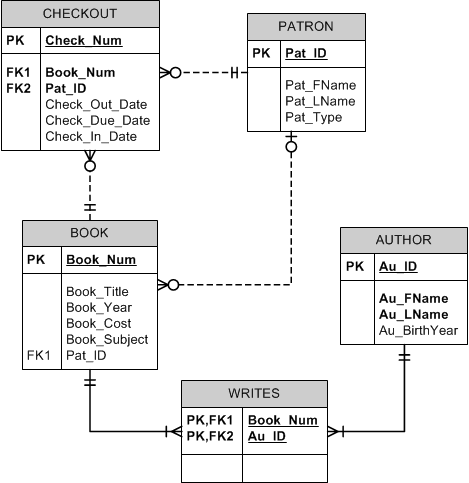
**Assignment 3 SQL**

Total points: 50

**This assignment should be completed individually. For each problem, submit your SQL statements and a screen shot of the SQL results in a single Word document or pdf file. Submit the file via eLearning.**

I recommend creating a new user and workspace named after your netid, log in as that user and load the database script facts.sql (provided in this week's assignment folder).  Before you attempt to write any SQL queries, familiarize yourself with the database structure and data. I have provided a relational diagram and sample data for this database.

Write queries to address each of the problems below. **Submit both the SQL statements and the screen prints of the outputs from Oracle.** (2 points for the screen shots).  **Be sure the workspace name (your id) is included in your screen shots!!!**



FACT Description and ERD

The CIS Department at Tiny College maintains the Free Access to Current Technology (FACT) library of eBooks. FACT is a collection of current technology eBooks for use by faculty and students. Agreements with the publishers allow patrons to electronically check out a book, which gives them exclusive access to the book online through the FACT website, but only one patron at a time can have access to a book. A book must have at least one author but can have many. An author must have written at least one book to be included in the system, but may have written many. A book may have never been checked out, but can be checked out many times by the same patron or different patrons over time. Because all faculty and staff in the department are given accounts at the online library, a patron may have never checked out a book or they may have checked out many books over time. To simplify determining which patron currently has a given book checked out, a redundant relationship between BOOK and PATRON is maintained.

*Please note: When a book is checked out a row is insert into the checkout table. The book table is updated and the pat\_id is updated with the patron who checked out the book. When a book is returned the checkout table is updated – the dates are updated and the book table is updated, the pat\_id is set to null.*

1. **Create a new table to track the library location.** *Screen shot not required, please paste the SQL statement to create the table.* ***(7 pts)***

LIBRARY (lib\_id, lib\_name, lib\_address, lib\_city, lib\_state, lib\_zip)

LIB\_ID is the primary key and should be numeric.

LIB\_NAME, LIB\_ADDRESS, and LIB\_CITY is between 1 and 35 characters. – These should not be null.

LIB\_STATE is 2 characters – default to TX.

LIB\_ZIP is 5 numbers. Check for one of the following zip codes – 75081, 75080, 75082, 75079, 75078

create table library (

lib\_id number constraint library\_lib\_id\_pk primary key,

lib\_name varchar2(35) not null,

lib\_address varchar2(35) not null,

lib\_city varchar2(35) not null,

lib\_state varchar2(2) default 'TX',

lib\_zip number(5,0) check(lib\_zip in (75081, 75080, 75082, 75079, 75078))

)

1. **Insert the following records into the LIBRARY table –** *Show each SQL statement and then provide a screen shot listing all the rows in the table.* ***(6 pts)***

1000, JFK Library, 800 West Campbell Road, Richardson, 75080

1001, MLK Library, 105 King Blvd., Richardson, TX, 75081

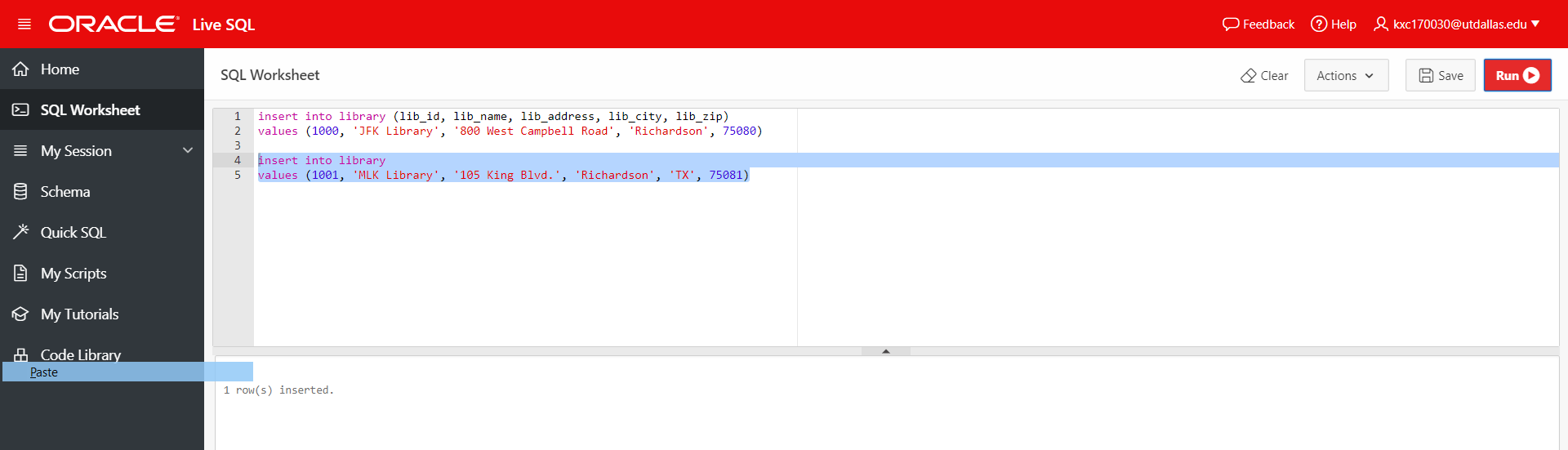
insert into library (lib\_id, lib\_name, lib\_address, lib\_city, lib\_zip)

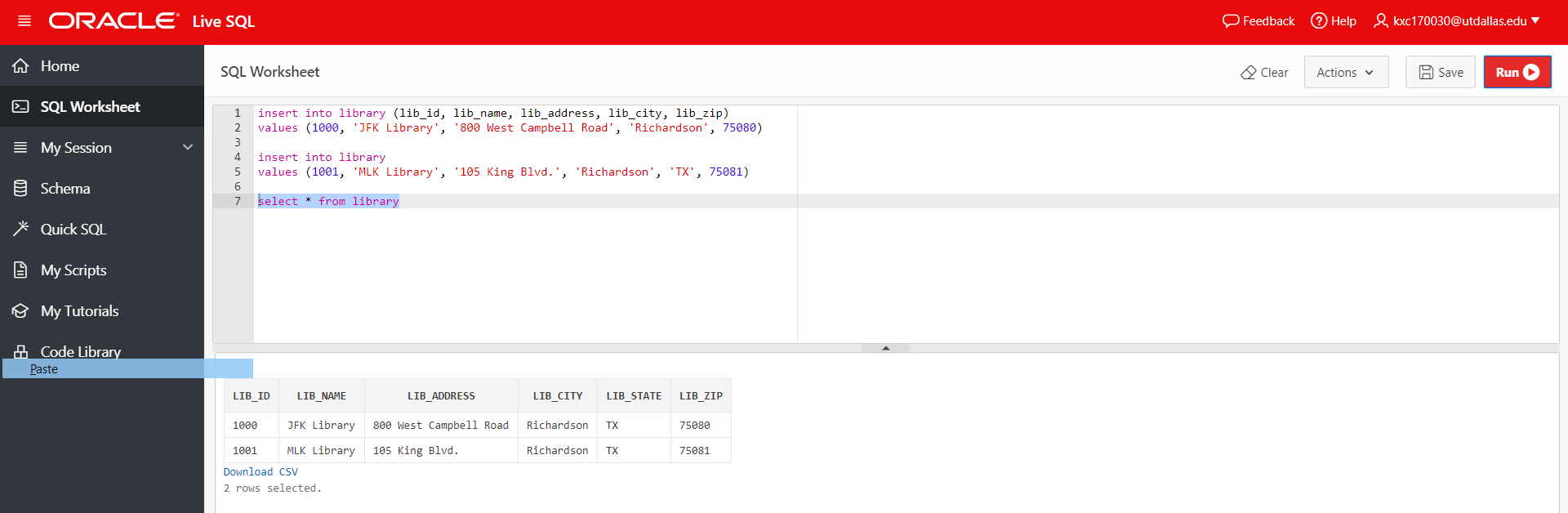
values (1000, 'JFK Library', '800 West Campbell Road', 'Richardson', 75080)

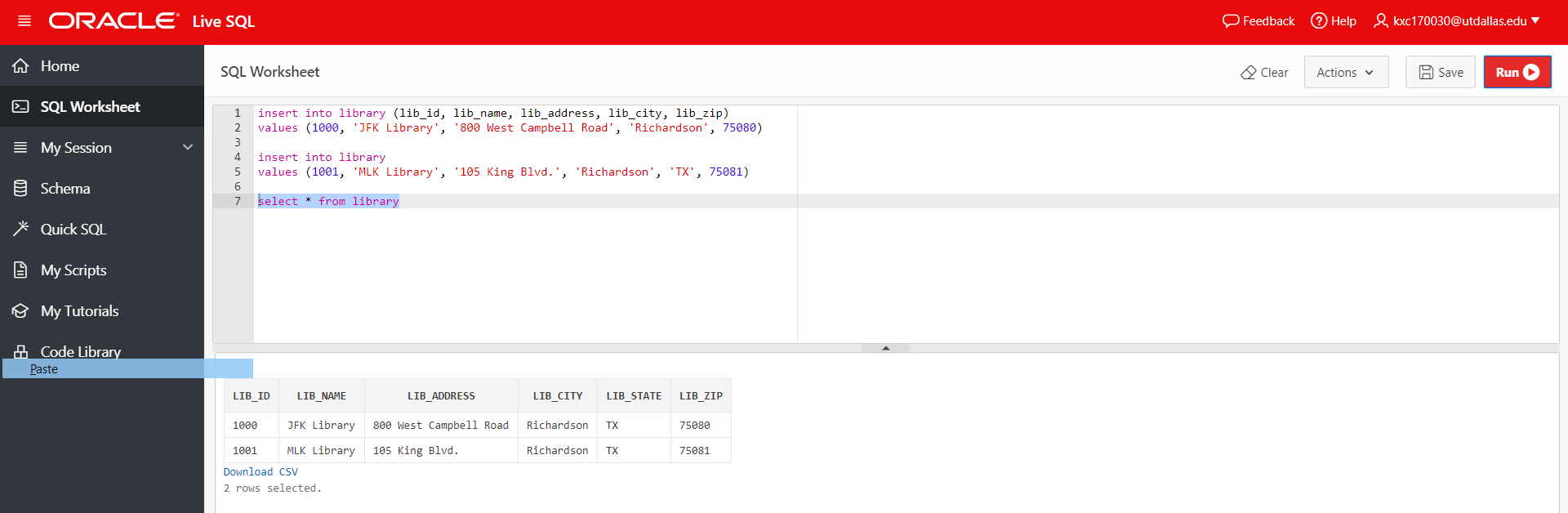


insert into library

values (1001, 'MLK Library', '105 King Blvd.', 'Richardson', 'TX', 75081)



select \* from library



1. **Due to inflation, the cost of all books published in 2017 will increase by 6%. Write a single SQL command to increase all book costs by 6%.** *Show the update statement and then select the updated rows and show in them in your screen shot. (6 pts)*

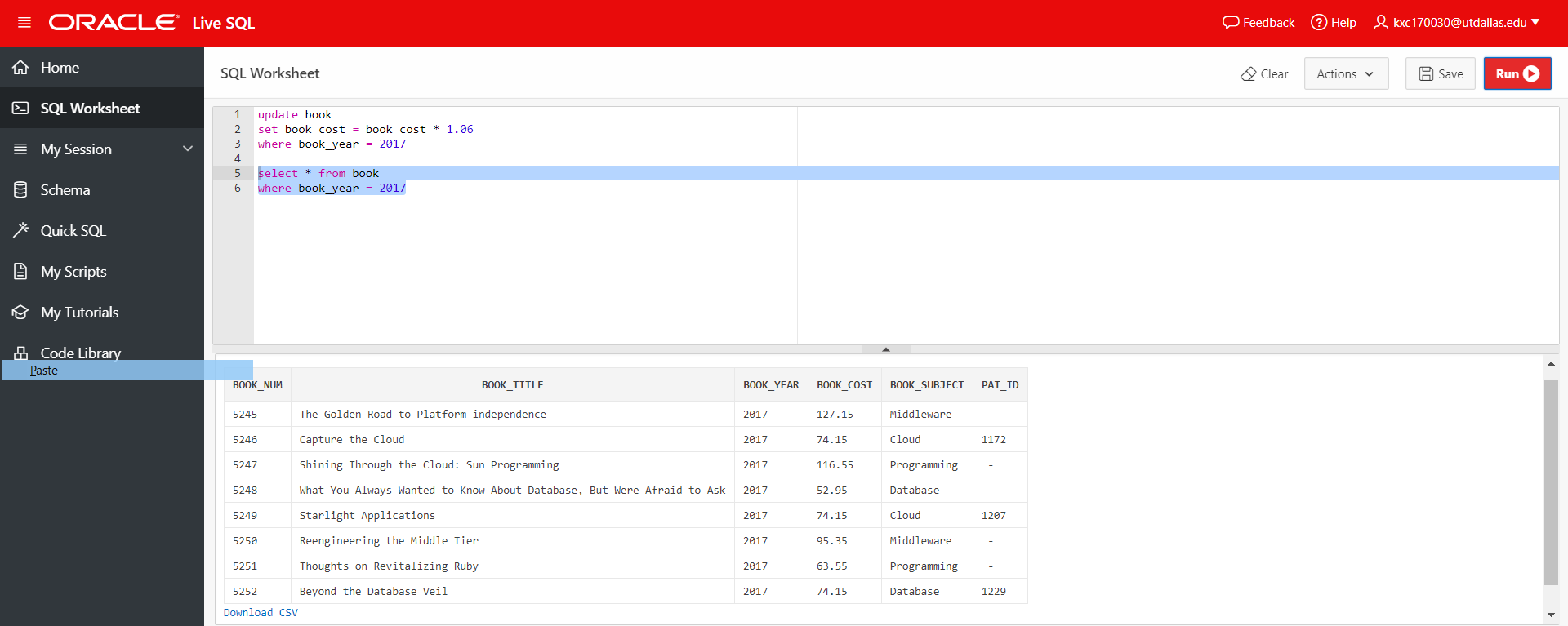
update book

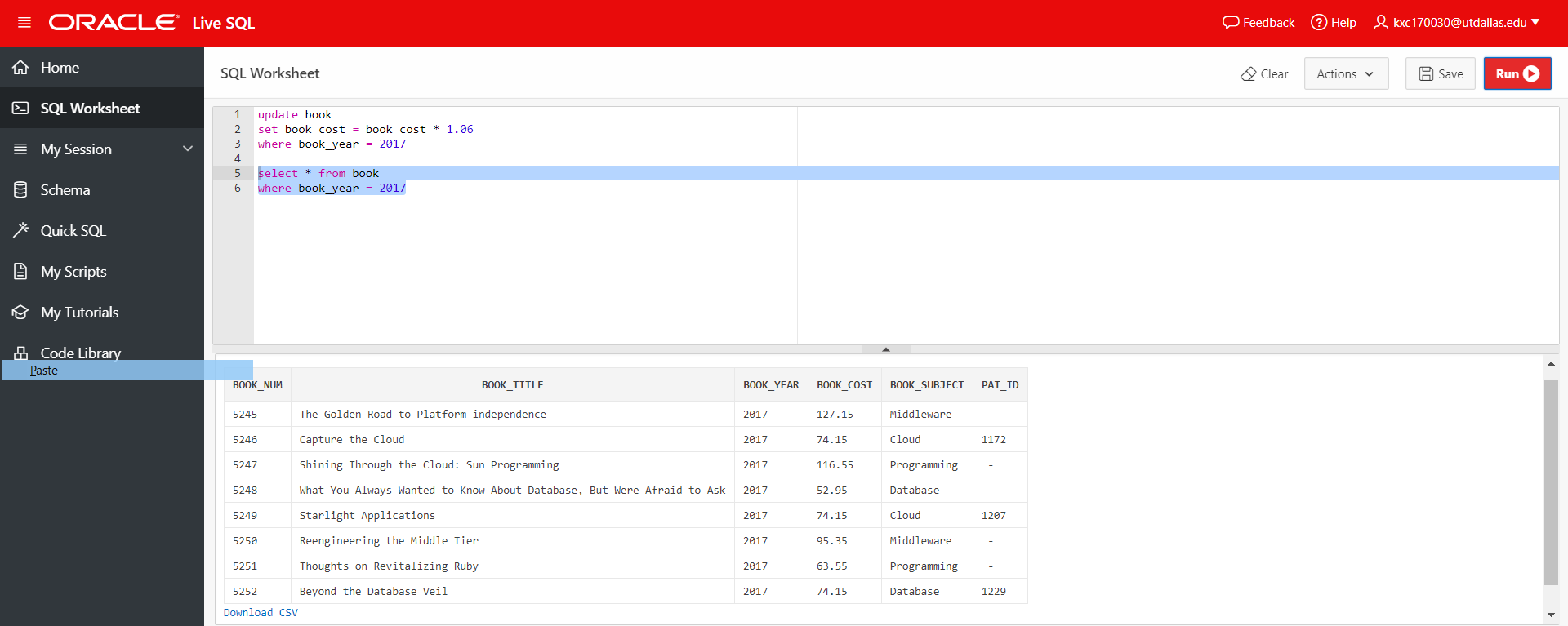
set book\_cost = book\_cost \* 1.06

where book\_year = 2017



select \* from book

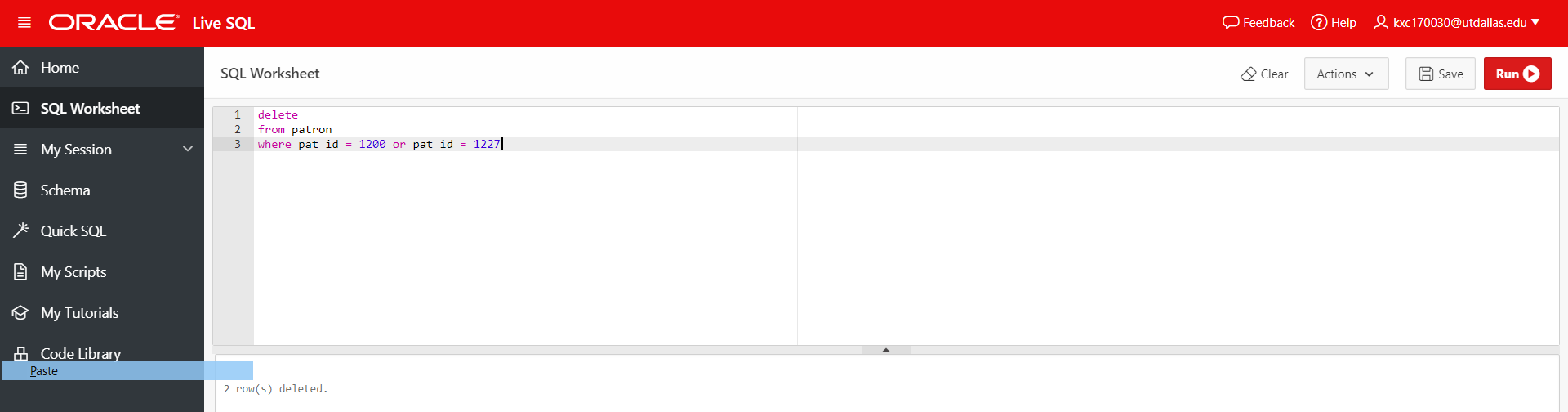
****where book\_year = 2017

****

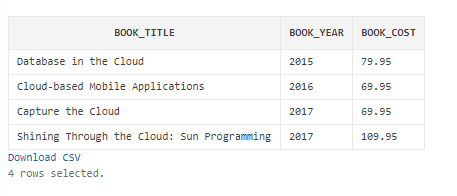
1. **Write a single query that will delete patron 1200 and 1227 from the patron table.** *Show the query that will delete those patrons. (5 pts)*

delete

from patron

where pat\_id = 1200 or pat\_id = 1227

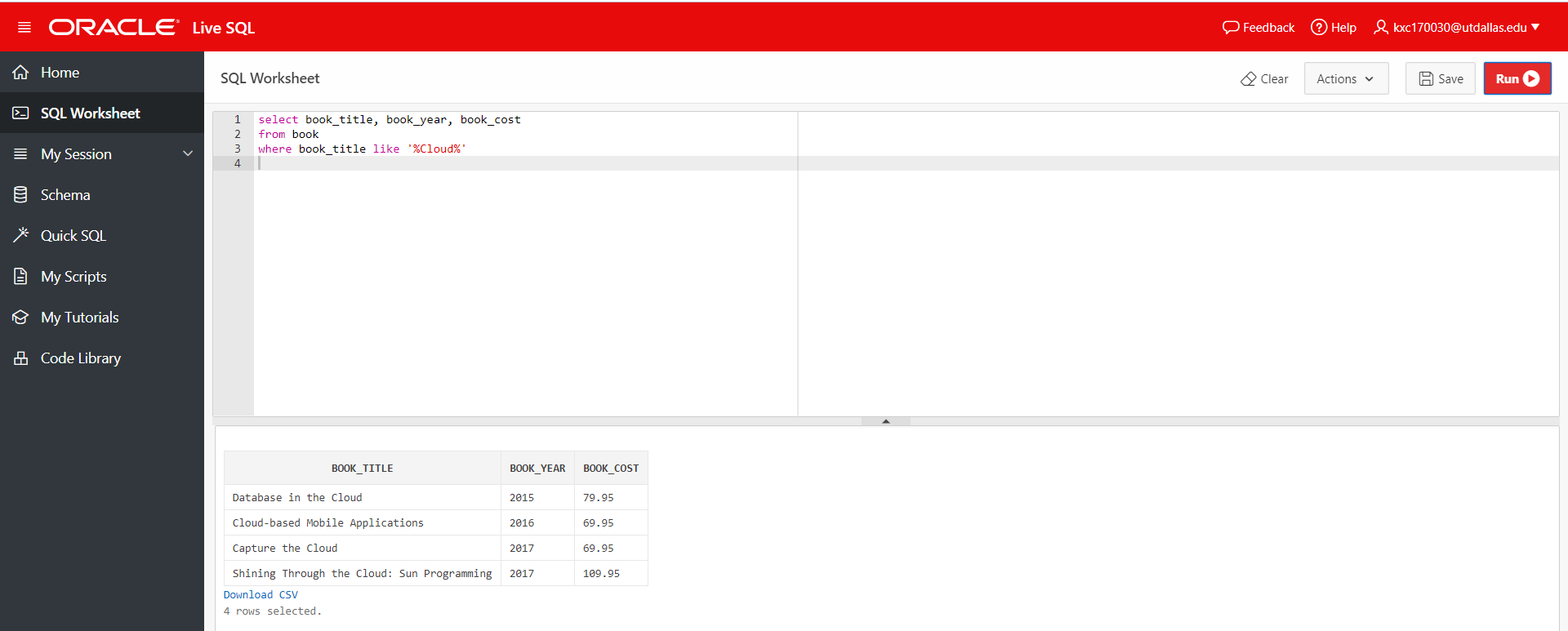
1. **Write a query that will list all the books with “Cloud” in the title. Display the book\_title, book\_year, and book\_cost.** *(5 pts)*

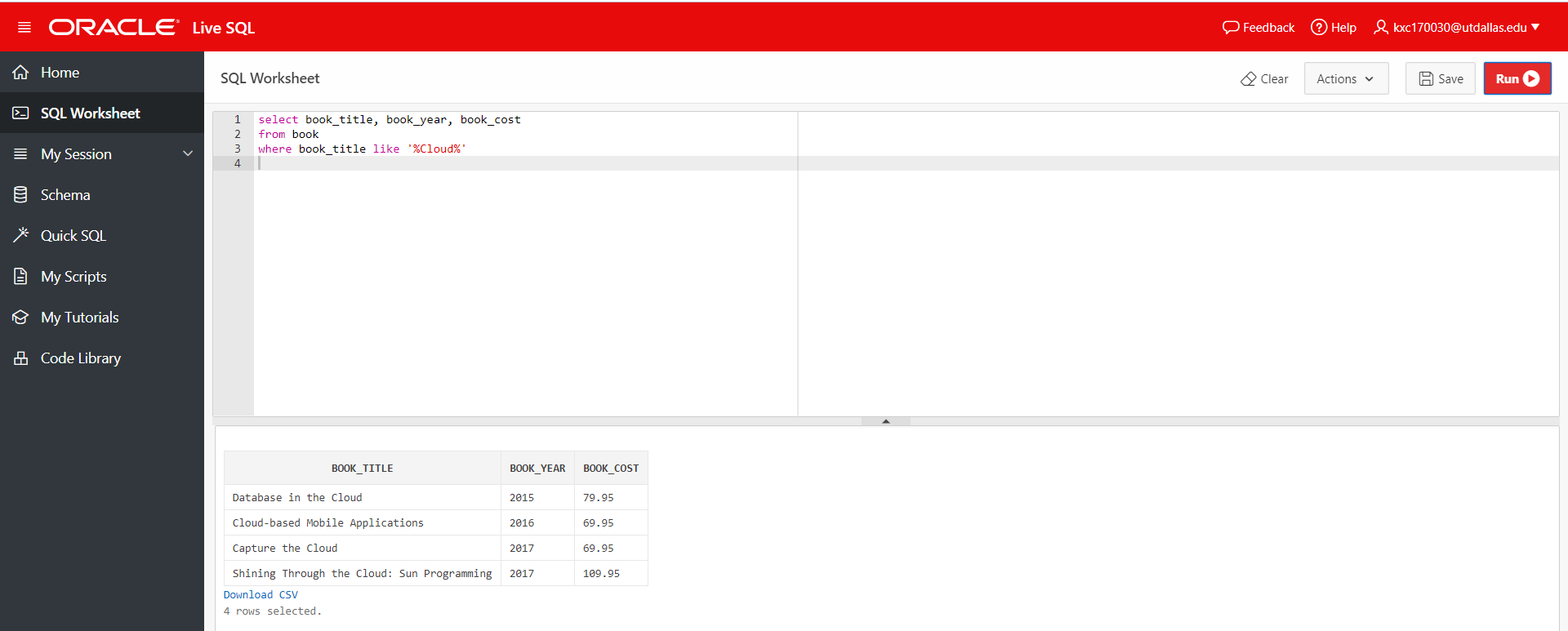


select book\_title, book\_year, book\_cost

from book

where book\_title like '%Cloud%'

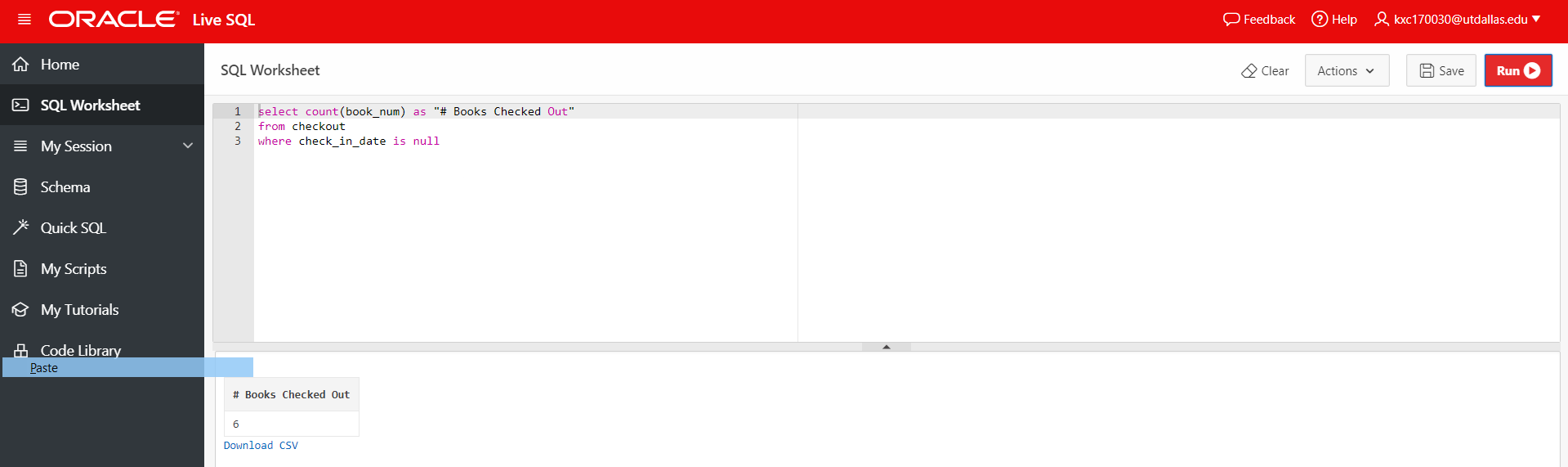


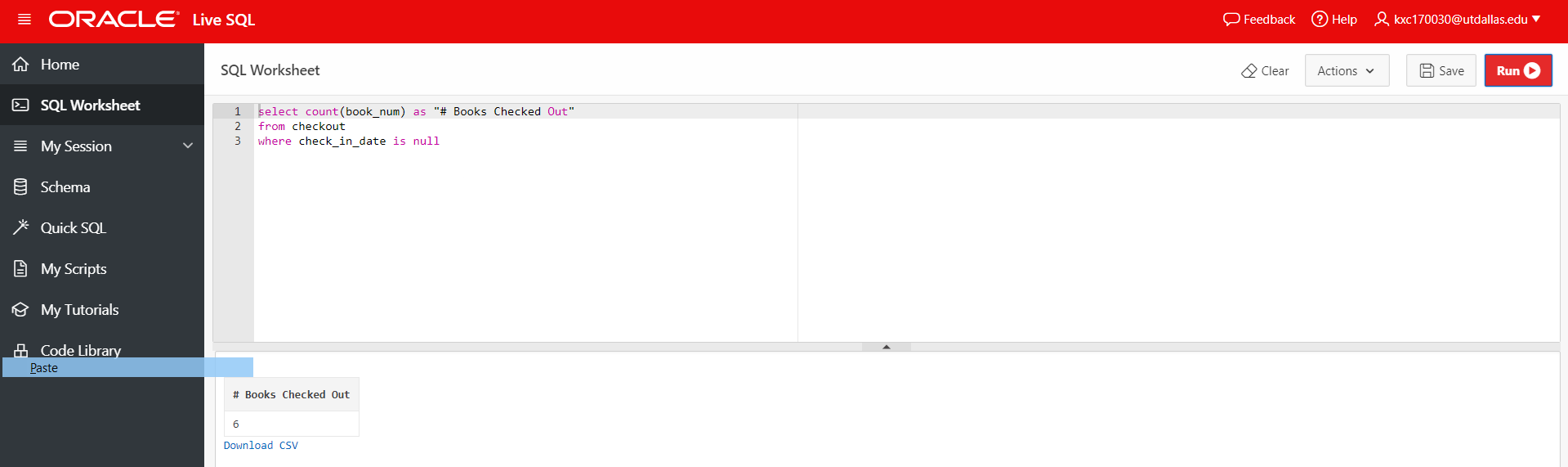


1. **How many books are currently checked out? Write a query that will display the number. Display the column heading as “# Books Checked Out”** *(5 pts)*

select count(book\_num) as "# Books Checked Out"

from checkout

where check\_in\_date is null

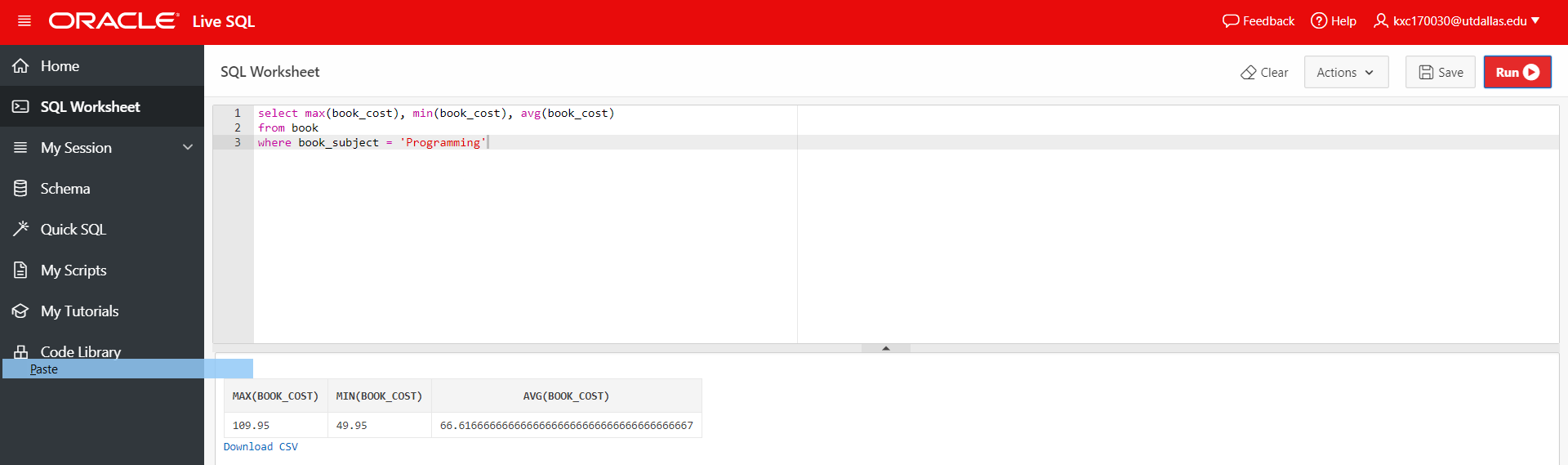


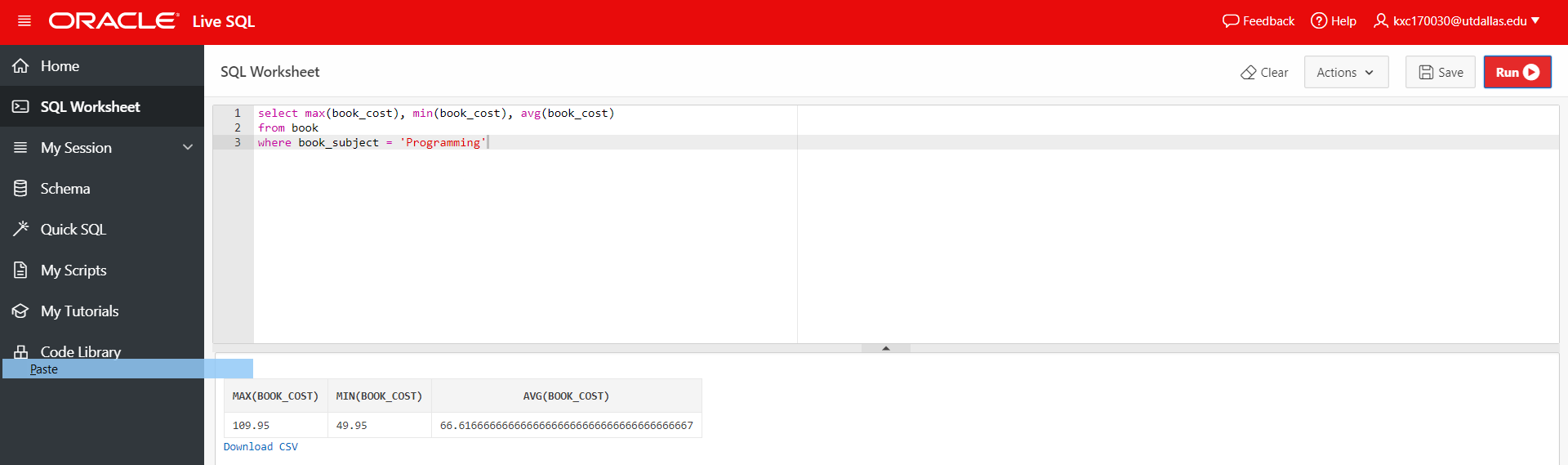
1. **Display the max, min and average book cost for all “Programming” books.** *(5 pts)*

select max(book\_cost), min(book\_cost), avg(book\_cost)

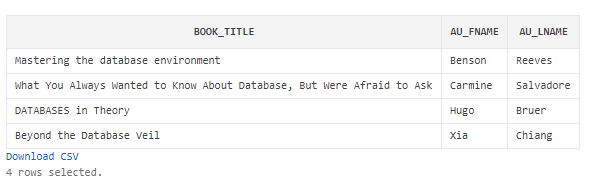
from book

where book\_subject = 'Programming'





1. **Display the book title and author first and last name for all the Database books.** *(5 pts)*



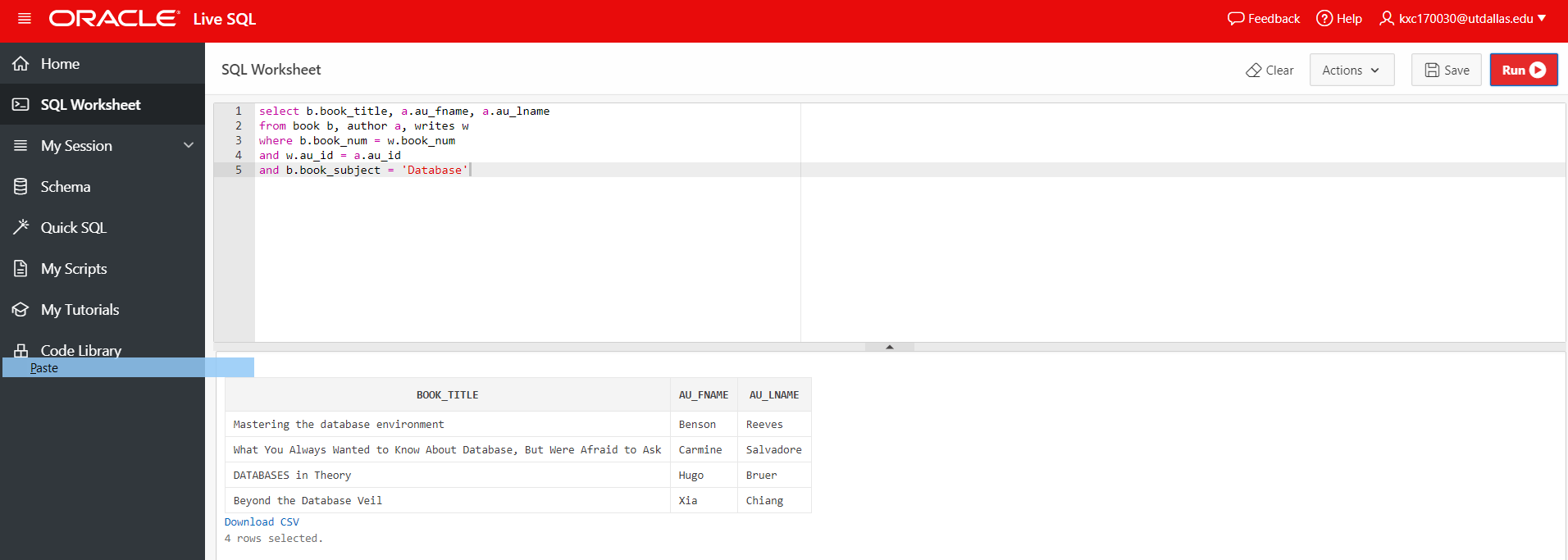
select b.book\_title, a.au\_fname, a.au\_lname

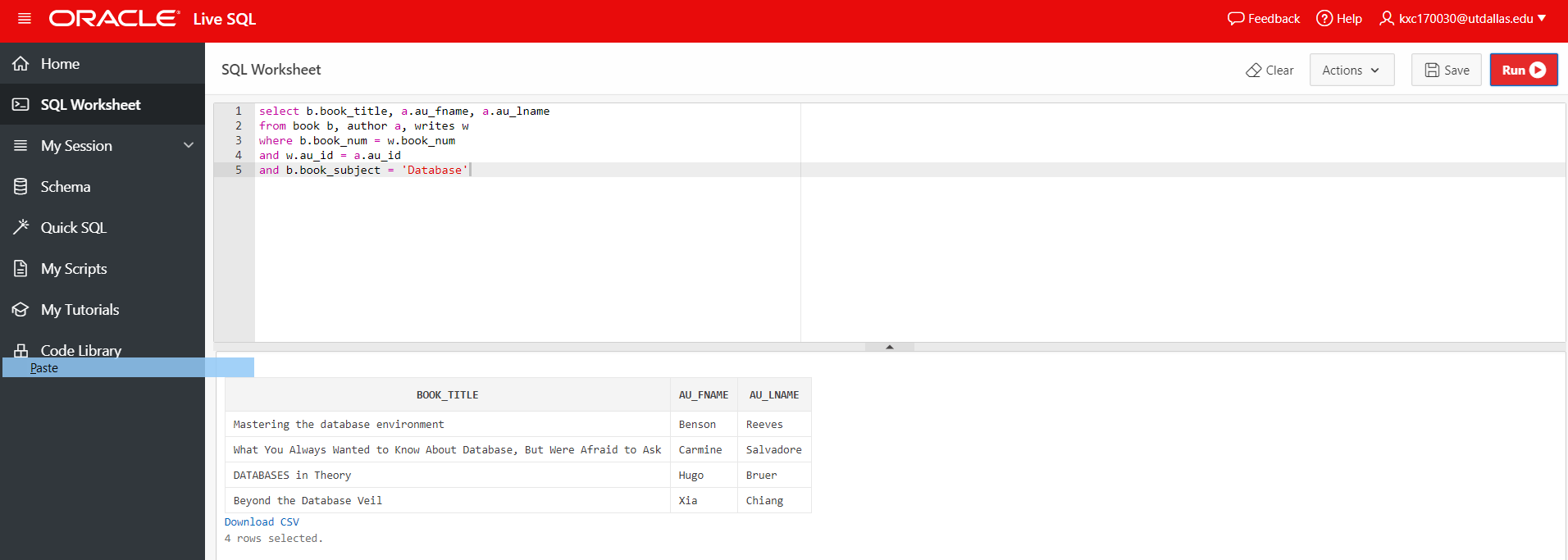
from book b, author a, writes w

where b.book\_num = w.book\_num

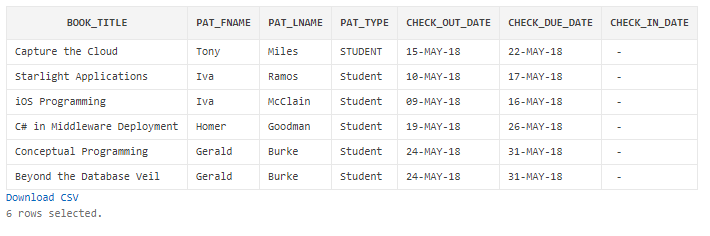
and w.au\_id = a.au\_id

and b.book\_subject = 'Database'





1. **List all the books that were never checked in. Display the book title, patron first and last name, the patron type, the check out date, due date, and check in date.** *(6 pts)*

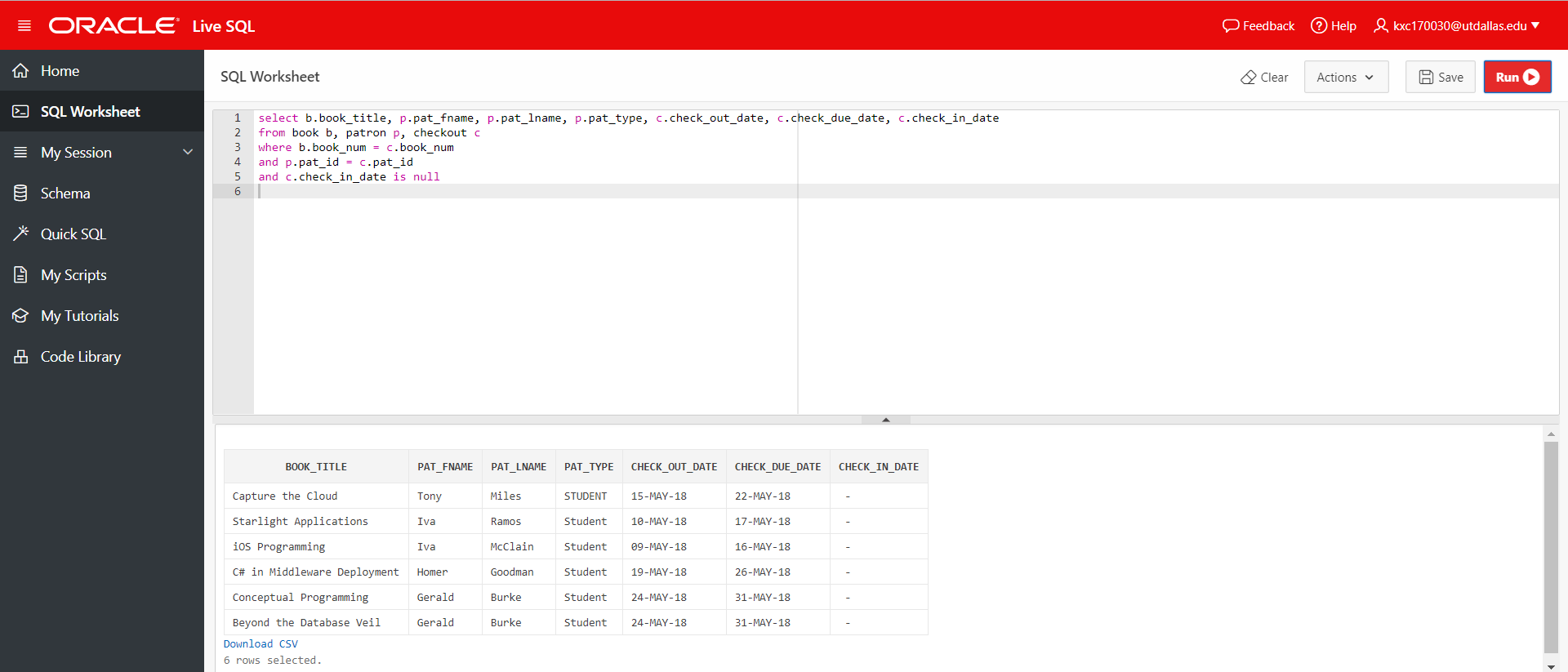


select b.book\_title, p.pat\_fname, p.pat\_lname, p.pat\_type, c.check\_out\_date, c.check\_due\_date, c.check\_in\_date

from book b, patron p, checkout c

where b.book\_num = c.book\_num

and p.pat\_id = c.pat\_id

and c.check\_in\_date is null

