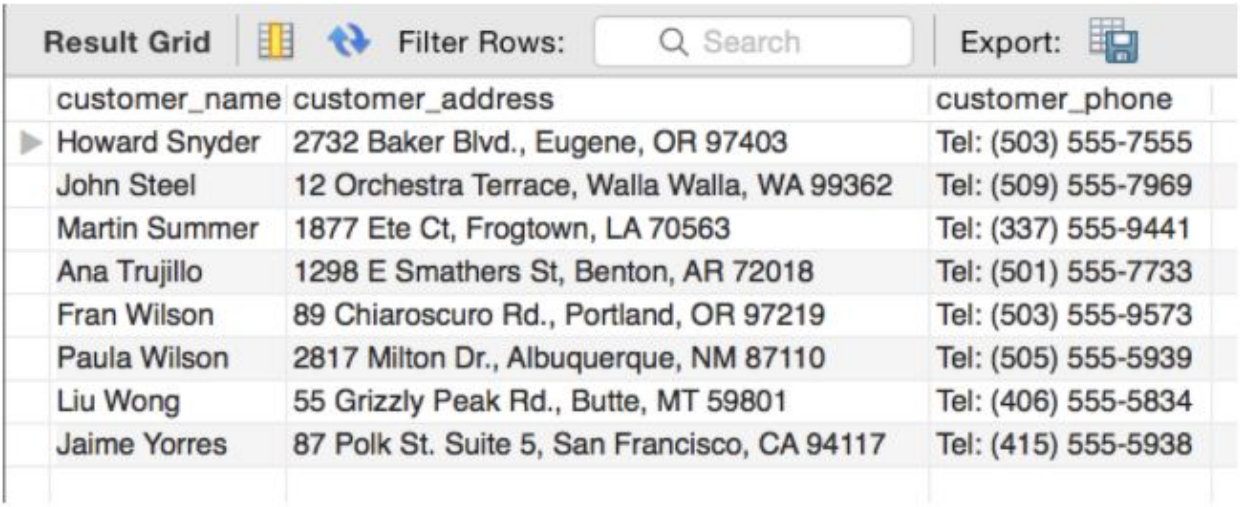
Robert Bruffey
INST327

Assignment 1

```
#Q.1
#create customer_name by concat customer_first_name and last_name
#create customer_address by concatting customer address, city, and zip from customers
#use WHERE to retrieve only data between two names

SELECT CONCAT(customer_first_name, ' ',customer_last_name) AS customer_name,
          CONCAT(customer_address, ', ', customer_city, ', ', customer_state,
         ' ', customer_zip) AS customer_address,
          CONCAT('Tel: ',customer_phone) AS customer_phone
 FROM customers
 WHERE customer_last_name >= 'Snyder' AND customer_last_name <= 'Yorres'
 ORDER BY customer_last_name;
```



Result Grid | Filter Rows: Search | Export:

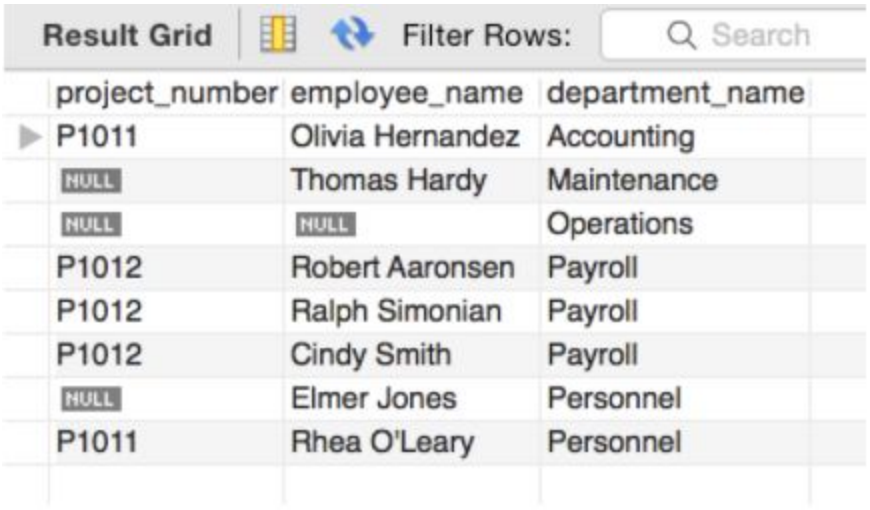
customer_name	customer_address	customer_phone
Howard Snyder	2732 Baker Blvd., Eugene, OR 97403	Tel: (503) 555-7555
John Steel	12 Orchestra Terrace, Walla Walla, WA 99362	Tel: (509) 555-7969
Martin Summer	1877 Ete Ct, Frogtown, LA 70563	Tel: (337) 555-9441
Ana Trujillo	1298 E Smathers St, Benton, AR 72018	Tel: (501) 555-7733
Fran Wilson	89 Chiaroscuro Rd., Portland, OR 97219	Tel: (503) 555-9573
Paula Wilson	2817 Milton Dr., Albuquerque, NM 87110	Tel: (505) 555-5939
Liu Wong	55 Grizzly Peak Rd., Butte, MT 59801	Tel: (406) 555-5834
Jaime Yorres	87 Polk St. Suite 5, San Francisco, CA 94117	Tel: (415) 555-5938

```
#Q.2
#uses columns invoice number, date, total, and due_date from active_invoices
#must use DATE_FORMAT to convert invoce_due_date into a written out date as month_due
#must use UNION b/c the date formats for each table are different and we need to combime
them

SELECT invoice_number, invoice_date, invoice_total, invoice_due_date,
                CONCAT('DUE IN ', UCASE(DATE_FORMAT(invoice_due_date, '%M %Y'))) AS
month_due
        FROM active_invoices
   WHERE invoice_due_date BETWEEN '2014-06-01' AND '2014-06-31'
UNION
SELECT invoice_number, invoice_date, invoice_total, invoice_due_date,
                CONCAT('Due in ', DATE_FORMAT(invoice_due_date,'%M %Y')) AS
month_due
        FROM active_invoices
   WHERE invoice_due_date BETWEEN '2014-07-01' AND '2014-07-31'
   ORDER BY invoice_total DESC; #descending order of invoice totals
```

Result Grid | Filter Rows: Search | Export:

invoice_number	invoice_date	invoice_total	invoice_due_date	month_due
40318	2014-07-18	21842.00	2014-07-20	Due in July 2014
P-0608	2014-04-11	20551.18	2014-06-30	DUE IN JUNE 2014
0-2436	2014-05-07	10976.06	2014-07-17	Due in July 2014
367447	2014-05-31	2433.00	2014-06-30	DUE IN JUNE 2014
989319-497	2014-04-17	2312.20	2014-06-26	DUE IN JUNE 2014
97/522	2014-04-30	1962.13	2014-07-10	Due in July 2014
989319-487	2014-04-18	1927.54	2014-06-19	DUE IN JUNE 2014
31359783	2014-05-23	1575.00	2014-06-09	DUE IN JUNE 2014
97/553	2014-04-27	904.14	2014-07-09	Due in July 2014
I77271-O01	2014-06-05	662.00	2014-06-24	DUE IN JUNE 2014
31361833	2014-05-23	579.42	2014-06-09	DUE IN JUNE 2014
9982771	2014-06-03	503.20	2014-06-18	DUE IN JUNE 2014
97/553B	2014-04-26	313.55	2014-07-09	Due in July 2014
547481328	2014-05-20	224.00	2014-06-25	DUE IN JUNE 2014
547480102	2014-05-19	224.00	2014-06-24	DUE IN JUNE 2014
963253271	2014-05-09	158.00	2014-06-28	DUE IN JUNE 2014
547479217	2014-05-17	116.00	2014-06-22	DUE IN JUNE 2014
963253263	2014-05-10	109.50	2014-06-22	DUE IN JUNE 2014
134116	2014-06-01	90.36	2014-07-02	Due in July 2014
39104	2014-06-20	85.31	2014-07-20	Due in July 2014
263253270	2014-05-18	67.92	2014-06-25	DUE IN JUNE 2014
963253272	2014-05-09	61.50	2014-06-29	DUE IN JUNE 2014
263253268	2014-05-18	59.97	2014-06-24	DUE IN JUNE 2014

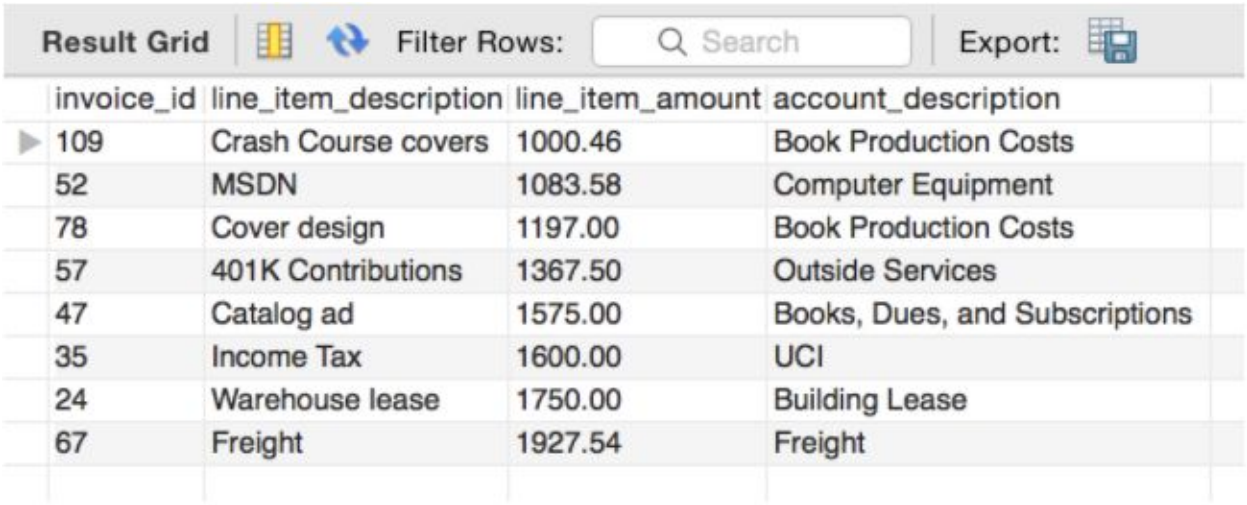
```
#Q.3
#uses project_number, department name, and concats first and last name
#as employee from tables project, employees, and departments
#the table uses a Right JOIN due to the null inputs on the left side
#of the table.
#since we are gathering columns from 3 different tables, we must use two join
#functions of related primary keys (employee_id, department_number)
#the data is then ordered by department_name and then within each department
#by employee_name

SELECT project_number, CONCAT(first_name,' ', last_name) AS employee_name,
        department_name
    FROM projects p RIGHT JOIN employees e
        ON p.employee_id = e.employee_id
    RIGHT JOIN departments d
        ON e.department_number = d.department_number
    ORDER BY department_name, employee_name;
```



Result Grid | Filter Rows: Search

project_number	employee_name	department_name
P1011	Olivia Hernandez	Accounting
NULL	Thomas Hardy	Maintenance
NULL	NULL	Operations
P1012	Robert Aaronsen	Payroll
P1012	Ralph Simonian	Payroll
P1012	Cindy Smith	Payroll
NULL	Elmer Jones	Personnel
P1011	Rhea O'Leary	Personnel

```
#Q.4
#uses the invoice_ide, line_item_description, line_item_amount,
# and account_description from two different tables
# invoice_line_items and general_ledger_accounts that we must JOIN
#by their primary keys of account_number
#then we use only the data with line_item_amount > 1000 and < 2000
#and order by the line_item_amount.

SELECT invoice_id, line_item_description,
        line_item_amount, account_description
   FROM invoice_line_items li JOIN general_ledger_accounts la
                ON li.account_number = la.account_number
        WHERE line_item_amount > '1000' AND
                        line_item_amount < '2000'
        ORDER BY line_item_amount;
```



invoice_id	line_item_description	line_item_amount	account_description
109	Crash Course covers	1000.46	Book Production Costs
52	MSDN	1083.58	Computer Equipment
78	Cover design	1197.00	Book Production Costs
57	401K Contributions	1367.50	Outside Services
47	Catalog ad	1575.00	Books, Dues, and Subscriptions
35	Income Tax	1600.00	UCI
24	Warehouse lease	1750.00	Building Lease
67	Freight	1927.54	Freight