Justin Ganzak

Homework 3:

Question 1:

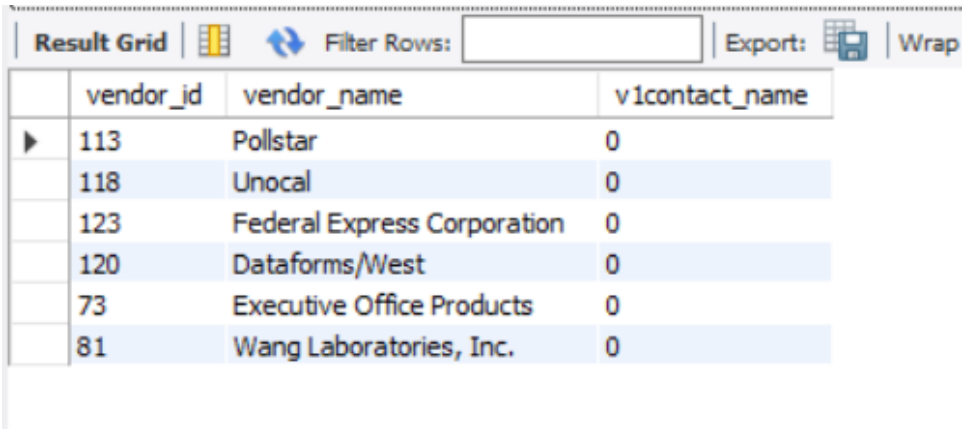
```
SELECT vendor_name, invoice_number, invoice_date,
invoice_total - payment_total - credit_total AS balance_due
FROM invoices i INNER JOIN vendors v
ON v.vendor_id = i.vendor_id
WHERE invoice_total - payment_total - credit_total > 200
ORDER BY balance_due
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vendor_name	invoice_number	invoice_date	balance_due
Blue Cross	547480102	2018-08-01	224.00
Ford Motor Credit Company	9982771	2018-07-24	503.20
Ingram	31361833	2018-07-21	579.42
Malloy Lithographing Inc	0-2436	2018-07-31	10976.06
Malloy Lithographing Inc	P-0608	2018-07-23	19351.18

Question 2:

```
SELECT
  v1.vendor_id
, v1.vendor_name
    , v1.vendor_contact_first_name +' ' + v1.vendor_contact_last_name AS v1contact_name
FROM Vendors v1
INNER JOIN Vendors v2
ON v1.vendor_contact_first_name = v2.vendor_contact_first_name
WHERE NOT v1.vendor_id = v2.vendor_id
ORDER BY v1.vendor_contact_last_name
```

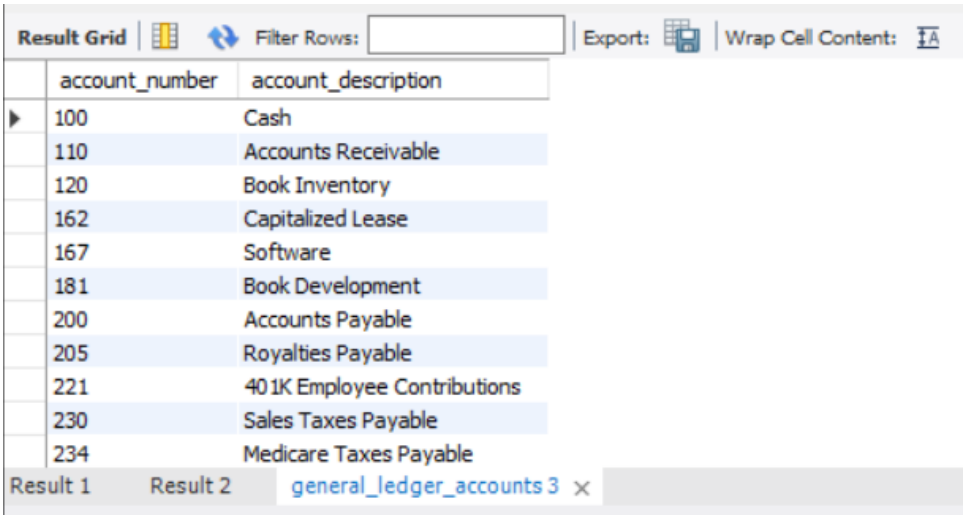


Result Grid | Filter Rows: | Export: | Wrap

vendor_id	vendor_name	v1contact_name
113	Pollstar	0
118	Unocal	0
123	Federal Express Corporation	0
120	Dataforms/West	0
73	Executive Office Products	0
81	Wang Laboratories, Inc.	0

Question 3:

```
SELECT
account_number,
account_description
FROM general_ledger_accounts
WHERE NOT EXISTS
(SELECT * from invoice_line_items WHERE
invoice_line_items.account_number = general_ledger_accounts.account_number)
ORDER BY account_number;
```



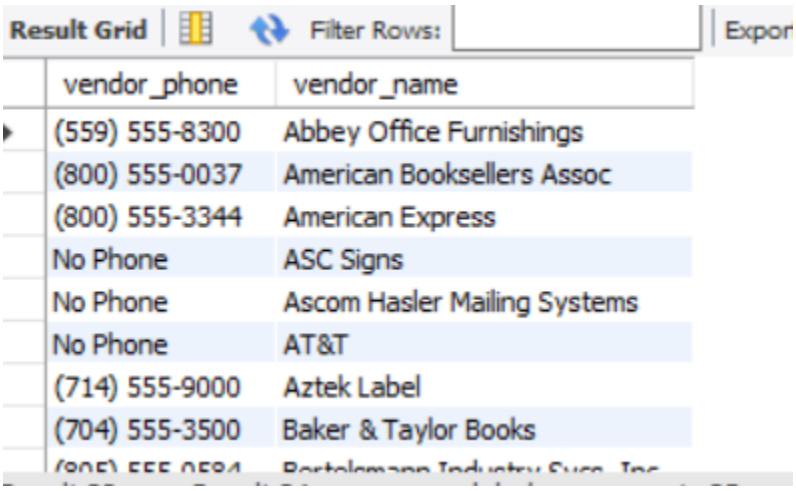
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

account_number	account_description
100	Cash
110	Accounts Receivable
120	Book Inventory
162	Capitalized Lease
167	Software
181	Book Development
200	Accounts Payable
205	Royalties Payable
221	401K Employee Contributions
230	Sales Taxes Payable
234	Medicare Taxes Payable

Result 1 | Result 2 | general_ledger_accounts 3

Question 4:

```
SELECT IFNULL (vendor_phone, "No Phone") AS vendor_phone,
vendor_name
FROM vendors
ORDER BY vendor_name;
```

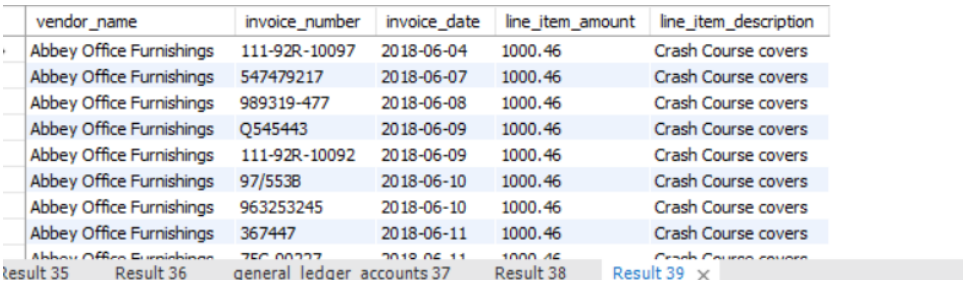


Result Grid | Filter Rows: | Expor

vendor_phone	vendor_name
(559) 555-8300	Abbey Office Furnishings
(800) 555-0037	American Booksellers Assoc
(800) 555-3344	American Express
No Phone	ASC Signs
No Phone	Ascom Hasler Mailing Systems
No Phone	AT&T
(714) 555-9000	Aztek Label
(704) 555-3500	Baker & Taylor Books

Question 5:

```
SELECT
vendor_name,
invoice_number,
invoice_date,
line_item_amount,
line_item_description
FROM vendors, invoice_line_items, invoices
ORDER BY
vendor_name;
```

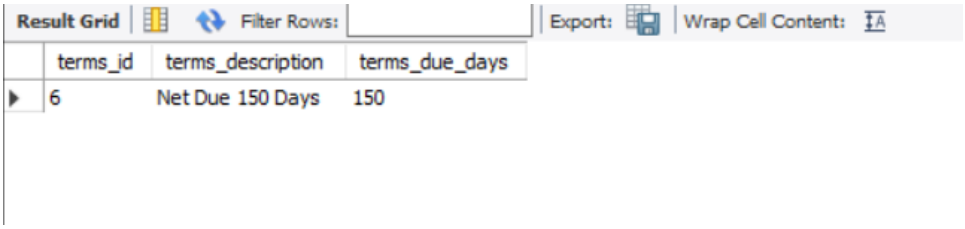


vendor_name	invoice_number	invoice_date	line_item_amount	line_item_description
Abbey Office Furnishings	111-92R-10097	2018-06-04	1000.46	Crash Course covers
Abbey Office Furnishings	547479217	2018-06-07	1000.46	Crash Course covers
Abbey Office Furnishings	989319-477	2018-06-08	1000.46	Crash Course covers
Abbey Office Furnishings	Q545443	2018-06-09	1000.46	Crash Course covers
Abbey Office Furnishings	111-92R-10092	2018-06-09	1000.46	Crash Course covers
Abbey Office Furnishings	97/553B	2018-06-10	1000.46	Crash Course covers
Abbey Office Furnishings	963253245	2018-06-10	1000.46	Crash Course covers
Abbey Office Furnishings	367447	2018-06-11	1000.46	Crash Course covers

Result 35 | Result 36 | general_ledger_accounts 37 | Result 38 | Result 39

Question 6:

```
DROP TABLE new_terms;

CREATE TABLE new_terms

(terms_id INT,

terms_description VARCHAR(150),

terms_due_days INT)

;

INSERT INTO

new_terms (terms_id, terms_description, terms_due_days)

VALUES (6, 'Net Due 150 Days', 150);

SELECT * FROM new_terms

;
```



terms_id	terms_description	terms_due_days
6	Net Due 150 Days	150

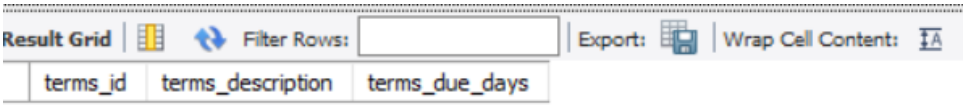
Question 7:

terms_id	terms_description	terms_due_days
6	Net Due 180 Days	180

```
UPDATE new_terms

SET terms_description = 'Net Due 180 Days', terms_due_days = 180

WHERE terms_description = 'Net Due 150 Days';

SELECT * FROM new_terms;
```

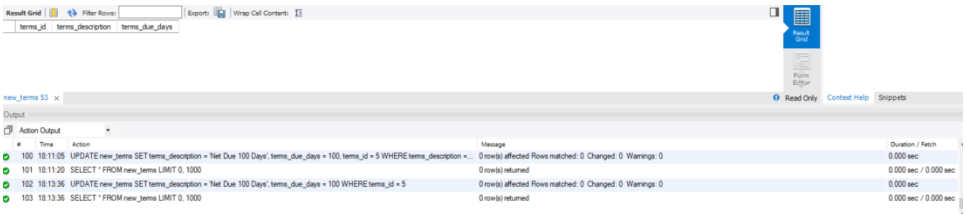
Question 8:

```
DELETE FROM new_terms
WHERE terms_id = 6;
SELECT * FROM new_terms;
```



Question 9:

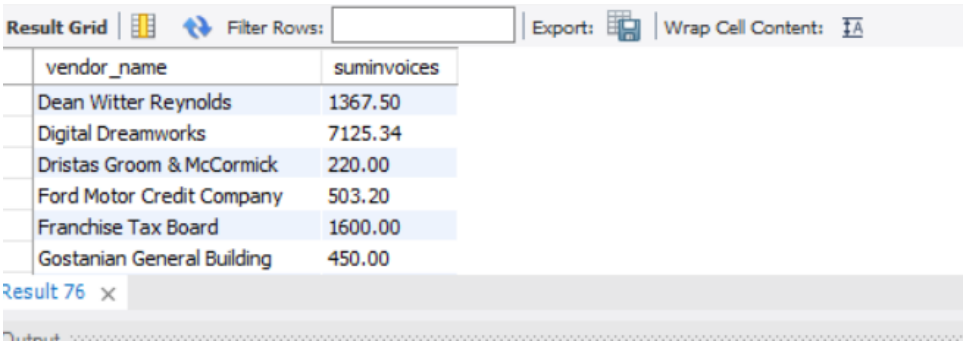
```
UPDATE new_terms
SET terms_description = 'Net Due 100 Days', terms_due_days = 100
WHERE terms_id = 5;
SELECT * FROM new_terms;
```



Question 10:

```
SELECT
vendor_name,
   suminvoices
FROM (
SELECT vendor_id, sum(invoice_total) AS suminvoices FROM invoices
GROUP BY vendor_id
) AS invoicedata
INNER JOIN vendors ON vendors.vendor_id=invoicedata.vendor_id;
```

SELECT * FROM invoices;



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vendor_name	suminvoices
Dean Witter Reynolds	1367.50
Digital Dreamworks	7125.34
Dristas Groom & McCormick	220.00
Ford Motor Credit Company	503.20
Franchise Tax Board	1600.00
Gostanian General Building	450.00

Result 76

Question 11:

A)Many:1. Many vendors have one general_ledger_accounts with them

B) SELECT gl.account_description

FROM general_ledger_accounts

gl join invoice_line_items iv

ON gl.account_number = iv.account_number

WHERE invoice_id

IN (select max(count(invoice_id))

FROM invoice_line_items

GROUP BY invoice_id);

C) SELECT gl.account_description

FROM general_ledger_accounts

gl join invoice_line_items iv ON

gl.account_number = iv.account_number WHERE

invoice_id IN(SELECT max(sum(invoice_id))

FROM

invoice_line_items

GROUP BY invoice_id);

Question 12:

```
SELECT vendor_name,
COUNT(DISTINCT li.account_number) AS 'Number of Accounts'
FROM
vendors v
JOIN invoices i
ON v.vendor_id = i.vendor_id
JOIN invoice_line_items li
ON i.invoice_id - li.invoice.id
GROUP BY vendor_name
HAVING COUNT(DISTINCT li.account_number)>1
ORDER BY vendor_name;
```

Question 13:

```
SELECT v.vendor_id, v.vendor_name, count(*) AS 'no_of_time'
FROM invoices AS i
INNER JOIN venders AS v
ON i.vendor_id = v.vendor_name
HAVING count(*) =
(SELECT max(a.no_of_time)
FROM
(SELECT vendor_id, count(*) AS 'no_of_time'
FROM invoices
WHERE invoice_due_date <payment_date
GROUP BY vendor_id) AS a);
```

Question 15:

```
#A
SELECT account_number, account_description
FROM general_ledger_accounts;
#B
```

```
SELECT DISTINCT account_number, account_description
FROM general_ledger_accounts
WHERE NOT EXISTS
(SELECT account_number FROM invoice_line_items);
```

#C

```
SELECT DISTINCT account_number, account_description
FROM general_ledger_accounts
WHERE NOT EXISTS (SELECT account_number
FROM
invoice_line_items)
ORDER BY account_number
```