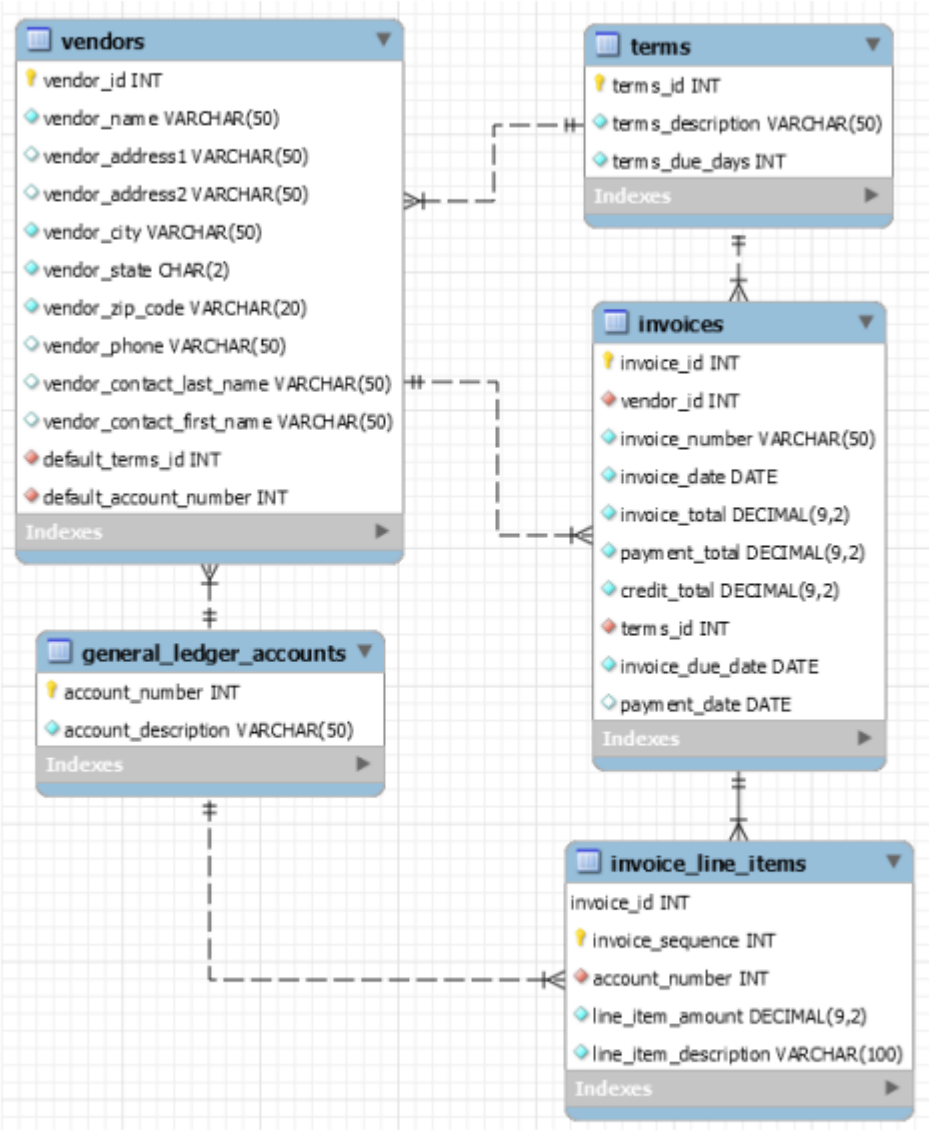
Sean Bupara
MGP/B/T 287
Winter 2021

HOMEWORK 3

Using the script provided on Canvas, install the Accounts Payable (ap) database. Run the script to create the database and load the data.

The ERD for the ap database is given below.



Use the ap database that you just created and write MySQL queries for the following questions.

1. ***(2 points)*** Select all data from the Invoices table. **Paste a screenshot of a partial resultset.** ***(114 rows)***

```
1 •  SELECT *
2    FROM INVOICES;
```

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

invoice_id	vendor_id	invoice_number	invoice_date	invoice_total	payment_total	credit_total	terms_id	invc
1	122	989319-457	2014-04-08	3813.33	3813.33	0.00	3	201
2	123	263253241	2014-04-10	40.20	40.20	0.00	3	201
3	123	963253234	2014-04-13	138.75	138.75	0.00	3	201
4	123	2-000-2993	2014-04-16	144.70	144.70	0.00	3	201
5	123	963253251	2014-04-16	15.50	15.50	0.00	3	201
6	123	963253261	2014-04-16	42.75	42.75	0.00	3	201
7	123	963253237	2014-04-21	172.50	172.50	0.00	3	201
8	89	125520-1	2014-04-24	95.00	95.00	0.00	1	201
9	121	97/488	2014-04-24	601.95	601.95	0.00	3	201
10	123	263253250	2014-04-24	42.67	42.67	0.00	3	201

INVOICES 2

Output

Action Output

#	Time	Action	Message
1	17:48:16	SELECT * FROM INVOICES LIMIT 0, 1000	114 row(s) returned

2. ***(2 points)*** Display the Invoice number, Invoice date, and the Invoice total. Sort in descending sequence by Invoice Total. **Paste a screenshot of a partial resultset.** ***(114 rows)***

```
5   -- Question 2
6 • SELECT invoice_number, invoice_date, invoice_total
7   FROM INVOICES
8   ORDER BY invoice_total Desc;
```

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

invoice_number	invoice_date	invoice_total
963253261	2014-04-16	42.75
263253250	2014-04-24	42.67
963253262	2014-04-25	42.50
109596	2014-06-24	41.80
963253245	2014-06-10	40.75
263253241	2014-04-10	40.20
111-92R-10093	2014-07-06	39.77
963253252	2014-07-12	38.75
963253260	2014-05-25	36.00
1-202-2978	2014-06-03	33.00
111-92R-10095	2014-07-15	32.70
263253253	2014-06-18	31.95
263253273	2014-07-22	30.75
7548906-20	2014-06-24	27.00
963253269	2014-06-15	26.75
263253265	2014-07-02	26.25
24946731	2014-06-15	25.67
963253267	2014-06-17	23.50
263253257	2014-07-30	22.57
111-92R-10094	2014-06-01	19.67
203339-13	2014-07-05	17.50
111897	2014-07-15	16.62
111-92R-10096	2014-04-30	16.33
111-92R-10097	2014-06-04	16.33
963253251	2014-04-16	15.50
4-314-3057	2014-06-11	13.75
4-321-2596	2014-05-05	10.00
4-342-8069	2014-05-14	10.00
43966316	2014-06-22	10.00
21-4748363	2014-05-03	9.95
21-4923721	2014-05-13	9.95

INVOICES 1

Output

Action Output

#	Time	Action	Message
17	10:38:40	INSERT INTO invoices VALUES (1,122,'989319-457','2014-04-08','3813.33','3813.33','0.00',3,'2014-05-08','2014-05-07'), (2,123,'263253241','2014-04-...	114 row(s) affected Records: 114 Duplicates: 0 Warnings: 0
18	10:38:40	INSERT INTO invoice_line_items VALUES (1,1,553,'3813.33','Freight'), (2,1,553,'40.20','Freight'), (3,1,553,'138.75','Freight'), (4,1,553,'144.70','Int\'l ship ...	118 row(s) affected Records: 118 Duplicates: 0 Warnings: 0
19	10:38:52	SELECT invoice_number, invoice_date, invoice_total FROM INVOICES ORDER BY invoice_total Desc LIMIT 0, 1000	114 row(s) returned

3. ***(2 points)*** Display all invoices between 2014-06-01 and 2014-06-30. **Paste a screenshot of a partial resultset.** ***(37 rows)***

```
10    -- Question 3
11 •  SELECT invoice_number, invoice_date
12    FROM INVOICES
13    WHERE invoice_date LIKE '2014-06-%';
```

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

invoice_number	invoice_date
40318	2014-06-01
111-92R-10094	2014-06-01
989319-437	2014-06-01
547481328	2014-06-03
31359783	2014-06-03
1-202-2978	2014-06-03
111-92R-10097	2014-06-04
547479217	2014-06-07
989319-477	2014-06-08
Q545443	2014-06-09
111-92R-10092	2014-06-09
97/553B	2014-06-10
963253245	2014-06-10
367447	2014-06-11
75C-90227	2014-06-11
963253256	2014-06-11
4-314-3057	2014-06-11
989319-497	2014-06-12
24946731	2014-06-15
963253269	2014-06-15
989319-427	2014-06-16
963253267	2014-06-17
509786	2014-06-18
263253253	2014-06-18
989319-487	2014-06-20
MABO1489	2014-06-21
133560	2014-06-22
24780512	2014-06-22
963253254	2014-06-22
43966316	2014-06-22
CBM9920-M-T...	2014-06-23

INVOICES 2

Output

Action Output

#	Time	Action	Message
18	10:38:40	INSERT INTO invoice_line_items VALUES (1,1,553,'3813.33','Freight'), (2,1,553,'40.20','Freight'), (3,1,553,'138.75','Freight'), (4,1,553,'144.70','Int\1 ship...	118 row(s) affected Rec
19	10:38:52	SELECT invoice_number, invoice_date, invoice_total FROM INVOICES ORDER BY invoice_total Desc LIMIT 0, 1000	114 row(s) returned
20	11:16:15	SELECT invoice_number, invoice_date FROM INVOICES WHERE invoice_date LIKE '2014-06-%' LIMIT 0, 1000	37 row(s) returned

4. ***(2 points)*** Write a SELECT statement that returns three columns:
   Vendor Name
   Vendor Contact Last Name
   Vendor Contact First Name.

   Then sort the result set by last name and then first name, both in ascending sequence. **Paste a screenshot of a partial resultset.** ***(122 rows)***

```
16 •  SELECT vendor_name as 'Vendor Name',
17        vendor_contact_last_name as 'Vendor Contact Last Name',
18        vendor_contact_first_name as 'Vendor Contact First Name'
19    FROM VENDORS
20    ORDER BY vendor_contact_last_name DESC, vendor_contact_first_name DESC;
21
```

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

Vendor Name	Vendor Contact Last Name	Vendor Contact First Name
Zylka Design	Ronaldsen	Jaime
California Business Machines	Rohansen	Anders
Reiter's Scientific & Pro Books	Rodolfo	Carlee
Malloy Lithographing Inc	Regging	Abe
Vision Envelope & Printing	Raven	Jamari
Crown Printing	Randrup	Leann
Cmg Information Services	Randall	Yash
Expedata Inc	Quintin	Marvin
Jobtrak	Quinn	Kenzie
Franchise Tax Board	Prado	Anita
Bertelsmann Industry Svcs...	Potter	Lance
Office Depot	Pinsippi	Val
Opamp Technical Books	Paris	Gideon
Texaco	Oren	Grace
Pacific Bell	Nickalus	Kurt
Rich Advertising	Neil	Ingrid
Venture Communications Int'l	Neftaly	Thalia
Custom Printing Company	Myles	Harley
Quality Education Data	Misael	Kayle
Frank E Wilber Co	Millerton	Johnathon
The Drawing Board	Mckayla	Jeffery
City Of Fresno	Mayte	Kendall
Capital Resource Credit	Maxwell	Jayda
California Chamber Of Com...	Mauro	Anton
The Library Ltd	Marques	Malia

VENDORS 3

Output

Action Output

#	Time	Action	Message
19	10:38:52	SELECT invoice_number, invoice_date, invoice_total FROM INVOICES ORDER BY invoice_total Desc LIMIT 0, 1000	114 row(s) returned
20	11:16:15	SELECT invoice_number, invoice_date FROM INVOICES WHERE invoice_date LIKE '2014-06-%' LIMIT 0, 1000	37 row(s) returned
21	11:18:27	SELECT vendor_name as 'Vendor Name', vendor_contact_last_name as 'Vendor Contact Last Name', vendor_contact_first_name as 'Vendor Cont...	122 row(s) returned

5. ***(5 points)*** Write a SELECT statement that returns the Vendor Contact Last Name and Vendor Contact First Name columns. Return only the contacts whose last name begins with the letter A, B, C, or E. Sort the result set by last name and then first name in ascending sequence. **Paste a screenshot of a partial resultset.** *(41 rows)*

```
22    -- Question 5
23 •  SELECT vendor_contact_last_name as 'Vendor Contact Last Name',
24        vendor_contact_first_name as 'Vendor Contact First Name'
25    FROM VENDORS
26    WHERE vendor_contact_last_name LIKE 'A%' OR vendor_contact_last_name LIKE 'B%'
27        OR vendor_contact_last_name LIKE 'C%' OR vendor_contact_last_name LIKE 'E%'
28    ORDER BY vendor_contact_last_name ASC, vendor_contact_first_name ASC;
29
```

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

Vendor Contact Last Name	Vendor Contact First Name
Aaronsen	Thom
Aileen	Joan
Alberto	Francesco
Alexis	Alexandro
Alondra	Zev
Angelica	Nashalie
Antavius	Troy
Anthoni	Kaitlyn
Anum	Trisha
Aranovitch	Robert
Armando	Jan
Arodondo	Cesar
Articunia	Mercedez
Aryn	Leroy
Baylee	Dakota
Beauregard	Violet
Bernard	Lucy
Bernardo	Brittnee
Blanca	Korah
Bluzinski	Rachael
Bradlee	Daniel
Bragg	Walter
Braydon	Anne
Brenton	Kila

VENDORS 4

Output

Action Output

#	Time	Action	Message
20	11:16:15	SELECT invoice_number, invoice_date FROM INVOICES WHERE invoice_date LIKE '2014-06-%' LIMIT 0, 1000	37 row(s) returned
21	11:18:27	SELECT vendor_name as 'Vendor Name', vendor_contact_last_name as 'Vendor Contact Last Name', vendor_contact_first_name as 'Vendor Cont...	122 row(s) returned
22	11:20:09	SELECT vendor_contact_last_name as 'Vendor Contact Last Name', vendor_contact_first_name as 'Vendor Contact First Name' FROM VENDORS W...	41 row(s) returned

6. ***(5 points)*** Display the following columns from the database:

   Due Date: The invoice due date
   Invoice Total
   10%: The 10% of the value of the invoice total
   Plus 10%: A 10% increase on the invoice total

   Return only the rows with an invoice total that is greater than or equal to 500 and less than or equal to 1000. Sort the result set in descending sequence by the invoice due date. **Paste a screenshot of the complete resultset.** ***(12 rows)***

```
30    -- Question 6
31 •  SELECT invoice_due_date AS 'Due Date',
32        invoice_total AS 'Invoice Total',
33        invoice_total*.1 AS '10%',
34        invoice_total*1.1 AS 'Plus 10%'
35    FROM INVOICES
36    WHERE invoice_total BETWEEN 500 AND 1000
37    ORDER BY invoice_due_date DESC;
```

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

Due Date	Invoice Total	10%	Plus 10%
2014-08-23	503.20	50.320	553.520
2014-08-10	579.42	57.942	637.362
2014-08-08	600.00	60.000	660.000
2014-08-06	739.20	73.920	813.120
2014-07-25	904.14	90.414	994.554
2014-07-11	936.93	93.693	1030.623
2014-06-29	639.77	63.977	703.747
2014-06-24	565.15	56.515	621.665
2014-06-20	953.10	95.310	1048.410
2014-06-02	856.92	85.692	942.612
2014-05-24	601.95	60.195	662.145
2014-05-16	662.00	66.200	728.200

Result 9

Output

Action Output

#	Time	Action	Message
1	11:25:14	SELECT invoice_due_date AS 'Due Date', invoice_total AS 'Invoice Total', invoice_total*.1 AS '10%', invoice_total*1.1 AS 'Plus 10%' FROM INVO...	12 row(s) returned

7. ***(2 points)*** Write a SELECT statement that returns the following columns.

   Invoice Number
   Invoice Total
   Payment Credit Total: Sum of Payment Total and Credit Total columns
   Balance Due: Invoice Total minus Payment Total minus Credit Total columns

   Return only invoices that have a balance due greater that is than $50. Sort the result set by balance due in descending sequence. **Paste a screenshot of the complete resultset.** ***(10 rows)***

```
39    -- Question 7
40 •  SELECT  invoice_number,
41        invoice_total,
42        (payment_total + credit_total) AS 'Payment Credit Total',
43        (invoice_total - payment_total - credit_total) AS 'Balance Due'
44    FROM INVOICES
45    WHERE invoice_total - payment_total - credit_total > 50
46    ORDER BY invoice_total - payment_total - credit_total DESC;
47
```

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

invoice_number	invoice_total	Payment Credit Total	Balance Due
P-0608	20551.18	1200.00	19351.18
0-2436	10976.06	0.00	10976.06
31361833	579.42	0.00	579.42
9982771	503.20	0.00	503.20
547480102	224.00	0.00	224.00
134116	90.36	0.00	90.36
39104	85.31	0.00	85.31
263253270	67.92	0.00	67.92
263253268	59.97	0.00	59.97
963253264	52.25	0.00	52.25

Result 10 ×

Output

Action Output

#	Time	Action	Message
1	11:28:19	SELECT invoice_number, invoice_total, (payment_total + credit_total) AS 'Payment Credit Total', (invoice_total - payment_total - credit_total) AS 'Ba...	10 row(s) returned

8. ***(2 points)*** Write a SELECT statement that returns these columns:

   Invoice Number
   Invoice Date
   Balance Due: Invoice Total minus Payment Total minus Credit Total columns
   Payment Date

Return only the rows where the Payment date contains a null value. **Paste a screenshot of the complete resultset.** *(11 rows)*

```
56    -- Question 9
57 •  SELECT v.vendor_name,
58        i.invoice_number,
59        i.invoice_date,
60        i.invoice_total - i.payment_total - i.credit_total AS 'Balance Due'
61    FROM VENDORS v
62    RIGHT JOIN INVOICES i
63    ON v.vendor_id = i.vendor_id
64    WHERE i.invoice_total - i.payment_total - i.credit_total <> 0
65    ORDER BY v.vendor_name ASC;
```

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

vendor_name	invoice_number	invoice_date	Balance Due
Blue Cross	547480102	2014-08-01	224.00
Cardinal Business Media, Inc.	134116	2014-07-28	90.36
Data Reproductions Corp	39104	2014-07-10	85.31
Federal Express Corporation	963253264	2014-07-18	52.25
Federal Express Corporation	263253268	2014-07-21	59.97
Federal Express Corporation	263253270	2014-07-22	67.92
Federal Express Corporation	263253273	2014-07-22	30.75
Ford Motor Credit Company	9982771	2014-07-24	503.20
Ingram	31361833	2014-07-21	579.42
Malloy Lithographing Inc	P-0608	2014-07-23	19351.18
Malloy Lithographing Inc	0-2436	2014-07-31	10976.06

Result 12 ×

Output

Action Output

#	Time	Action	Message
1	11:33:05	SELECT v.vendor_name, i.invoice_number, i.invoice_date, i.invoice_total - i.payment_total - i.credit_total AS 'Balance Due' FROM VENDORS v RIG...	11 row(s) returned

10. ***(5 points)*** Write a SELECT statement that returns the following columns:

Vendor Name
Default Account Number
Account Description

Return one row for each vendor. Sort the results by Account Description and then Vendor Name. **Paste a screenshot of a partial resultset.** *(122 rows)*

```
67    -- Question 10
68 •  SELECT v.vendor_name, v.default_account_number, gl.account_description
69    FROM VENDORS v
70    LEFT JOIN GENERAL_LEDGER_ACCOUNTS gl
71    ON v.default_account_number = gl.account_number
72    ORDER BY gl.account_description, v.vendor_name;
```

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

vendor_name	default_account_number	account_description
Dristas Groom & McCormick	591	Accounting
DMV Renewal	568	Auto License Fee
Newbrige Book Clubs	394	Book Club Royalties
Bertelsmann Industry Svcs. Inc	400	Book Printing Costs
Courier Companies, Inc	400	Book Printing Costs
Crown Printing	400	Book Printing Costs
Data Reproductions Corp	400	Book Printing Costs
Diversified Printing & Pub	400	Book Printing Costs
Malloy Lithographing Inc	400	Book Printing Costs
Fresno Photoengraving Company	403	Book Production Costs
Register of Copyrights	403	Book Production Costs
Shields Design	403	Book Production Costs
Zylka Design	403	Book Production Costs
Baker & Taylor Books	572	Books, Dues, and Su...
California Chamber Of Commerce	572	Books, Dues, and Su...
Compuserve	572	Books, Dues, and Su...
Computerworld	572	Books, Dues, and Su...
Jobtrak	572	Books, Dues, and Su...
Mcgraw Hill Companies	572	Books, Dues, and Su...

Result 14

Output

Action Output

#	Time	Action	Message
1	11:36:46	SELECT v.vendor_name, v.default_account_number, gl.account_description FROM VENDORS v LEFT JOIN GENERAL_LEDGER_ACCOUNTS gl ON ...	122 row(s) returned

11. ***(5 points)*** Write a SELECT statement that returns the following columns:

    Vendor Name
    Invoice Date
    Invoice Number
    Invoice Sequence
    Line Item Amount

    Sort the final results by Vendor Name, Invoice Date, Invoice Number, and Invoice Sequence.
    Paste a screenshot of a partial resultset. ***(118 rows)***

```
74    -- Question 11
75 •  SELECT v.vendor_name,
76        i.invoice_date,
77        i.invoice_number,
78        il.invoice_sequence,
79        il.line_item_amount
80    FROM VENDORS v
81    RIGHT JOIN INVOICES i
82    ON v.vendor_id = i.vendor_id
83    RIGHT JOIN INVOICE_LINE_ITEMS il
84    ON i.invoice_id = il.invoice_id
85    ORDER BY v.vendor_name, i.invoice_date, i.invoice_number, il.invoice_sequence;
```

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

vendor_name	invoice_date	invoice_number	invoice_sequence	line_item_amount
Abbey Office Furnishings	2014-07-05	203339-13	1	17.50
Bertelsmann Industry Svcs. Inc	2014-06-18	509786	1	6940.25
Blue Cross	2014-06-03	547481328	1	224.00
Blue Cross	2014-06-07	547479217	1	116.00
Blue Cross	2014-08-01	547480102	1	224.00
Cahners Publishing Company	2014-06-30	587056	1	2184.50
Cardinal Business Media, Inc.	2014-06-22	133560	1	175.00
Cardinal Business Media, Inc.	2014-07-28	134116	1	90.36
Coffee Break Service	2014-06-24	109596	1	41.80

Result 17 ×

Output

Action Output

#	Time	Action	Message
1	11:40:06	SELECT v.vendor_name, i.invoice_date, i.invoice_number, il.invoice_sequence, il.line_item_amount FROM VENDORS v RIGHT JOIN INVOIC...	118 row(s) returned

12. ***(5 points)*** Write a SELECT statement that returns the following columns:

Account Number: The general ledger account number
Account Description: The general ledger account description
Invoice ID from the Invoice Line Items table

Return one row for each account number that has never been used. Sort the result set by Account Number. **Paste a screenshot of a partial resultset.** *(54 rows)*

```
87    -- Question 12
88 •  SELECT distinct gl.account_number, gl.account_description, il.invoice_id
89    FROM GENERAL_LEDGER_ACCOUNTS gl
90    LEFT JOIN INVOICE_LINE_ITEMS il
91    ON gl.account_number = il.account_number
92    WHERE il.invoice_id IS NULL
93    ORDER BY account_number;
94
```

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

account_number	account_description	invoice_id
100	Cash	NULL
110	Accounts Receivable	NULL
120	Book Inventory	NULL
162	Capitalized Lease	NULL
167	Software	NULL
181	Book Development	NULL
200	Accounts Payable	NULL
205	Royalties Payable	NULL
221	401K Employee Contributions	NULL
230	Sales Taxes Payable	NULL
234	Medicare Taxes Payable	NULL
235	Income Taxes Payable	NULL
237	State Payroll Taxes Payable	NULL
238	Employee FICA Taxes Payable	NULL

Result 18

Output

Action Output

#	Time	Action	Message
1	11:40:54	SELECT distinct gl.account_number, gl.account_description, il.invoice_id FROM GENERAL_LEDGER_ACCOUNTS gl LEFT JOIN INVOICE_LINE_ITEM...	54 row(s) returned

13. ***(10 points)*** Generate the result set containing the following columns:

Vendor Name
Vendor State

If the vendor is in California, the value in the Vendor State column value should be "CA"; otherwise, the value should be "Outside CA." Sort the final result set by Vendor Name. **Paste a screenshot of a partial resultset.** *(122 rows)*

```
95      -- QUESTION 13
96  •   SELECT vendor_name, vendor_state
97      FROM Vendors
98      WHERE vendor_state = 'CA'
99      UNION
100     SELECT vendor_name, 'Outside CA'
101     FROM Vendors
102     WHERE vendor_state <> 'CA'
103     ORDER BY vendor_name;
```

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

vendor_name	vendor_state
Abbey Office Furnishings	CA
American Booksellers Assoc	Outside CA
American Express	CA
ASC Signs	CA
Ascom Hasler Mailing Systems	Outside CA
AT&T	Outside CA
Aztek Label	CA
Baker & Taylor Books	Outside CA
Bertelsmann Industry Svcs. Inc	CA
BFI Industries	CA
Bill Jones	CA
Bill Marvin Electric Inc	CA
Blanchard & Johnson Associates	CA
Blue Cross	CA
Blue Shield of California	CA
Boucher Communications Inc	Outside CA
Cahners Publishing Company	Outside CA
Cal State Termite	CA
California Business Machines	CA
California Chamber Of Commerce	CA
California Data Marketing	CA
Capital Resource Credit	Outside CA
Cardinal Business Media, Inc.	Outside CA

Result 25

Output

Action Output

#	Time	Action	Message
1	12:24:09	SELECT vendor_name, vendor_state FROM Vendors WHERE vendor_state = 'CA' UNION SEL...	122 row(s) returned

14. ***(2 points)*** Write a SELECT statement that returns one row for each vendor in the Invoices table that contains these columns:
    - The vendor_id column from the Vendors table
    - The sum of the invoice_total columns in the Invoices table for that vendor

    Paste a screenshot of the complete resultset. ***(34 rows)***

```
 99    -- QUESTION 14
100 •  SELECT vendor_id, sum(invoice_total)
101    FROM INVOICES
102    GROUP BY vendor_id;
```

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

vendor_id	sum(invoice_total)
34	1200.12
37	564.00
48	856.92
72	21927.31
80	265.36
81	936.93
82	600.00
83	2154.42
86	2433.00
88	207.78
89	95.00
90	356.48
94	17.50
95	171.01
96	662.00
97	19.90
99	6940.25
100	2184.50
102	41.80
103	1367.50
104	7125.34
105	220.00
106	503.20
107	1600.00
108	450.00
110	119892.41
113	1750.00
114	290.00
115	43.67
117	16.62
119	4901.26
121	6940.25
122	23177.96
123	4378.02

Result 21

Output

Action Output

#	Time	Action	Message
1	11:44:07	SELECT v.vendor_name, sum(payment_total) FROM INVOICES i LEFT JOIN VENDORS v ON i...	34 row(s) returned

15. ***(2 points)*** Write a SELECT statement that returns one row for each vendor that contains these columns:
    - The vendor_name column from the Vendors table
    - The sum of the payment_total columns in the Invoices table for that vendor

    Sort the result set in descending sequence by the payment total sum for each vendor. **Paste a screenshot of the complete resultset.** ***(34 rows)***

```
104     -- QUESTION 15
105     SELECT v.vendor_name, sum(payment_total)
106     FROM INVOICES i
107     LEFT JOIN VENDORS v
108     ON i.vendor_id = v.vendor_id
109     GROUP BY i.vendor_id
110     ORDER BY sum(payment_total) DESC;
```

Result Grid | Filter Rows: | Export:

vendor_name	sum(payment_total)
Malloy Lithographing Inc	86069.22
United Parcel Service	23177.96
Data Reproductions Corp	21842.00
Digital Dreamworks	7125.34
Bertelsmann Industry Svcs. Inc	6940.25
Zylka Design	6740.25
Yesmed, Inc	4901.26
Federal Express Corporation	4167.13
Computerworld	2433.00
Cahners Publishing Company	2184.50
Pollstar	1750.00
Franchise Tax Board	1600.00
Ingram	1575.00
Dean Witter Reynolds	1367.50
IBM	1200.12
Wang Laboratories, Inc.	936.93
Fresno County Tax Collector	856.92
Wells Fargo Bank	662.00
Reiter's Scientific & Pro Books	600.00
Gostanian General Building	450.00
Wakefield Co	356.48
Blue Cross	340.00
Postmaster	290.00
Dristas Groom & McCormick	220.00
Edward Data Services	207.78
Cardinal Business Media, Inc.	175.00
Pacific Bell	171.01
Evans Executone Inc	95.00
Roadway Package System, Inc	43.67
Coffee Break Service	41.80
Compuserve	19.90
Abbey Office Furnishings	17.50
Suburban Propane	16.62
Ford Motor Credit Company	0.00

3 11:47:46 SELECT v.vendor_name, sum(payment_total) FROM INVOICES i LEFT JOIN VENDORS v ON i... 34 row(s) returned

16. ***(2 points)*** Write a SELECT statement that returns one row for each vendor that contains three columns:
    - The vendor_name column from the Vendors table
    - The count of the invoices in the Invoices table for each vendor
    - The sum of the invoice_total columns in the Invoices table for each vendor

    Sort the result set so the vendor with the most invoices appears first. **Paste a screenshot of the complete resultset.** ***(34 rows)***

```
112     -- QUESTION 16
113 •   SELECT v.vendor_name, count(i.invoice_id), sum(i.invoice_total)
114     FROM INVOICES i
115     LEFT JOIN VENDORS v
116     ON i.vendor_id = v.vendor_id
117     GROUP BY i.vendor_id
118     ORDER BY sum(invoice_total) DESC;
```

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

vendor_name	count(i.invoice_id)	sum(i.invoice_total)
Malloy Lithographing Inc	5	119892.41
United Parcel Service	9	23177.96
Data Reproductions Corp	2	21927.31
Digital Dreamworks	1	7125.34
Zylka Design	8	6940.25
Bertelsmann Industry Svcs. Inc	1	6940.25
Yesmed, Inc	1	4901.26
Federal Express Corporation	47	4378.02
Computerworld	1	2433.00
Cahners Publishing Company	1	2184.50
Ingram	2	2154.42
Pollstar	1	1750.00
Franchise Tax Board	1	1600.00
Dean Witter Reynolds	1	1367.50
IBM	2	1200.12
Wang Laboratories, Inc.	1	936.93
Fresno County Tax Collector	1	856.92
Wells Fargo Bank	1	662.00
Reiter's Scientific & Pro Books	1	600.00
Blue Cross	3	564.00
Ford Motor Credit Company	1	503.20
Gostanian General Building	1	450.00
Wakefield Co	1	356.48
Postmaster	1	290.00
Cardinal Business Media, Inc.	2	265.36
Dristas Groom & McCormick	1	220.00
Edward Data Services	1	207.78
Pacific Bell	6	171.01
Evans Executone Inc	1	95.00
Roadway Package System, Inc	4	43.67
Coffee Break Service	1	41.80
Compuserve	2	19.90
Abbey Office Furnishings	1	17.50
Suburban Propane	1	16.62

1 11:50:36 SELECT v.vendor_name, count(i.invoice_id), sum(i.invoice_total) FROM INVOICES i LEFT JOIN... 34 row(s) returned

17. ***(15 points)*** Write a SELECT statement that returns one row for each general ledger account number that contains three columns:
    - The account_description column from the General_Ledger_Accounts table
    - The count of the items in the Invoice_Line_Items table that have the same account_number
    - The sum of the line_item_amount columns in the Invoice_Line_Items table that have the same account_number

    Return only those rows where the count of line items is greater than 1. This should return 10 rows.

    Group the result set by account description.

Sort the result set in descending sequence by the sum of the line item amounts. **Paste a screenshot of the complete resultset.** ***(10 rows)***

```
127     -- QUESTION 17
128 •   SELECT gl.account_description, count(il.account_number), sum(il.line_item_amount)
129     FROM GENERAL_LEDGER_ACCOUNTS gl
130     LEFT JOIN INVOICE_LINE_ITEMS il
131     ON gl.account_number = il.account_number
132     GROUP BY account_description
133     HAVING count(il.account_number) > 1
134     ORDER BY sum(il.line_item_amount) DESC;
```

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

account_description	count(il.account_number)	sum(il.line_item_amount)
Book Printing Costs	8	148759.97
Freight	60	27599.65
Outside Services	3	13394.10
Book Production Costs	8	6175.12
Books, Dues, and Subscriptions	6	5207.32
Direct Mail Advertising	6	3900.77
Computer Equipment	3	2137.05
Group Insurance	3	564.00
Telephone	7	266.01
Office Supplies	3	175.80

Result 29

Output

Action Output

#	Time	Action	Message
1	12:39:52	SELECT gl.account_description, count(il.acco...	10 row(s) returned

18. ***(12 points)*** Modify the solution to number 17 above so it returns only invoices dated in the second quarter of 2014 (April 1, 2014 to June 30, 2014). This should still return 10 rows but with some different line item counts for each vendor. **Paste a screenshot of the complete resultset.** ***(10 rows)***

```
136     -- QUESTION 18
137 •   SELECT gl.account_description,
138         i.invoice_date,
139         count(il.account_number),
140         sum(il.line_item_amount)
141     FROM GENERAL_LEDGER_ACCOUNTS gl
142     LEFT JOIN INVOICE_LINE_ITEMS il
143     ON gl.account_number = il.account_number
144     LEFT JOIN INVOICES i
145     ON il.invoice_id = i.invoice_id
146     GROUP BY account_description
147     HAVING count(il.account_number) > 1
148     AND invoice_date BETWEEN '2014-04-01' AND '2014-06-30'
149     ORDER BY sum(il.line_item_amount) DESC;
150
```

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

account_description	invoice_date	count(il.account_number)	sum(il.line_item_amount)
Book Printing Costs	2014-05-28	8	148759.97
Freight	2014-04-08	60	27599.65
Outside Services	2014-05-11	3	13394.10
Book Production Costs	2014-04-24	8	6175.12
Books, Dues, and Subscriptions	2014-05-03	6	5207.32
Direct Mail Advertising	2014-04-26	6	3900.77
Computer Equipment	2014-05-07	3	2137.05
Group Insurance	2014-06-03	3	564.00
Telephone	2014-04-24	7	266.01
Office Supplies	2014-04-26	3	175.80

Result 32

Output

Action Output

#	Time	Action	Message
1	12:51:10	SELECT gl.account_description, i.invoice_dat...	10 row(s) returned

19. ***(5 points)*** Write a SELECT statement that answers this question: *What is the total amount invoiced for each general ledger account number?* Return these columns:
    - The account_number column from the Invoice_Line_Items table
    - The sum of the line_item_amount columns from the Invoice_Line_Items table

    Use the WITH ROLLUP operator to include a row that gives the grand total. **Paste a screenshot of the complete resultset.** ***(22 rows including the grand total row)***

```
151     -- Question 19
152 •   SELECT account_number, sum(line_item_amount)
153     FROM INVOICE_LINE_ITEMS
154     GROUP BY account_number WITH ROLLUP;
```

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

account_number	sum(line_item_amount)
150	17.50
160	2137.05
170	356.48
400	148759.97
403	6175.12
507	1600.00
510	564.00
520	1750.00
521	16.62
522	266.01
523	450.00
540	3900.77
552	290.00
553	27599.65
570	175.80
572	5207.32
574	856.92
580	50.00
582	503.20
589	13394.10
591	220.00
NULL	214290.51

Result 36

Output

Action Output

#	Time	Action	Message
1	12:57:18	SELECT account_number, sum(line_item_amo...	22 row(s) returned

20. ***(20 points)*** Write a SELECT statement that answers this question: *Which vendors are being paid from more than one account?* Return these columns:
    - The vendor name from the Vendors table
    - The count of distinct general ledger accounts that apply to that vendor's invoices

    Paste a screenshot of the complete resultset. *(2 rows)*

```
156     -- Question 20
157 •   SELECT v.vendor_name, count(DISTINCT il.account_number)
158     FROM VENDORS v
159     RIGHT JOIN INVOICES i
160     ON v.vendor_id = i.vendor_id
161     LEFT JOIN invoice_line_items il
162     ON i.invoice_id = il.invoice_id
163     GROUP BY vendor_name
164     HAVING count(DISTINCT il.account_number) > 1;
```

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

vendor_name	count(DISTINCT il.account_number)
Wells Fargo Bank	3
Zylka Design	2

Result 86

Output

Action Output

#	Time	Action	Message
1	13:14:11	SELECT v.vendor_name, count(DISTINCT il.a...	2 row(s) returned

21. ***(10 points)*** Write a SELECT statement which shows the number of unpaid invoices and the total due for all those unpaid invoices. Hint: The total due = invoice total – payment total – credit total. The answer should be 11 and 32020.42.

    Paste a screenshot of the complete resultset. ***(1 row)***

```
166     -- Question 21
167 •   SELECT count(invoice_id), SUM(invoice_total - payment_total - credit_total) AS 'Total Due'
168     FROM INVOICES
169     WHERE payment_date IS NULL
```

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

	count(invoice_id)	Total Due
▶	11	32020.42

Result 91

Output

Action Output

#	Time	Action	Message
1	13:24:11	SELECT count(invoice_id), SUM(invoice_total ...	1 row(s) returned

22. ***(20 points)*** Write a SELECT statement that answers this question: What are the last payment date and total amount due for each vendor with each terms id? Return these columns:
    - The terms_id column from the Invoices table
    - The vendor_id column from the Invoices table
    - The last payment date for each combination of terms_id and vendor_id in the Invoices table
    - The sum of the balance due (invoice_total – payment_total – credit_total) for each combination of terms id and vendor id in the Invoices table

    Use the WITH ROLLUP operator to include rows that give a summary for each terms_id as well as a row that gives the grand total.

    Use the IF and GROUPING functions to replace the null values in the terms_id and vendor_id columns with literal values if they are for summary rows.

Paste a screenshot of the complete resultset. ***(40 rows including all subtotal rows and the grand total row)***

```
170     -- Question 22
171 •   SELECT
172         IF(GROUPING(terms_id), 'Terms ID Total', terms_id) AS 'Terms ID',
173         IF(GROUPING(vendor_id), 'Vendor Total', vendor_id) AS 'Vendor ID',
174         max(payment_date) AS 'Last Payment Date',
175         sum(invoice_total - (payment_total + credit_total)) AS 'Sum of Balance Due'
176     FROM INVOICES
177     GROUP BY terms_id, vendor_id WITH ROLLUP;
```

Terms ID	Vendor ID	Last Payment Date	Sum of Balance Due
1	34	2014-06-23	0.00
1	86	2014-06-17	0.00
1	88	2014-05-27	0.00
1	89	2014-05-01	0.00
1	114	2014-06-29	0.00
1	Vendor Total	2014-06-29	0.00
2	80	2014-07-16	90.36
2	81	2014-07-10	0.00
2	82	2014-08-13	0.00
2	83	2014-06-21	579.42
2	90	2014-08-07	0.00
2	94	2014-07-27	0.00
2	95	2014-08-06	0.00
2	96	2014-05-13	0.00
2	97	2014-05-28	0.00
2	119	2014-05-29	0.00
2	Vendor Total	2014-08-13	669.78
3	37	2014-07-07	224.00
3	48	2014-05-30	0.00
3	72	2014-06-29	85.31
3	99	2014-07-15	0.00
3	104	2014-06-24	0.00
3	105	2014-06-26	0.00
3	106	NULL	503.20
3	110	2014-08-27	30327.24
3	117	2014-08-14	0.00
3	121	2014-08-22	0.00
3	122	2014-08-24	0.00
3	123	2014-09-04	210.89
3	Vendor Total	2014-09-04	31350.64
4	100	2014-08-07	0.00
4	102	2014-08-04	0.00
4	107	2014-07-09	0.00
4	108	2014-07-03	0.00
4	115	2014-07-29	0.00
4	Vendor Total	2014-08-07	0.00
5	103	2014-07-31	0.00
5	113	2014-07-05	0.00
5	Vendor Total	2014-07-31	0.00
Terms I...	Vendor Total	2014-09-04	32020.42

#	Time	Action	Message
1	19:37:16	SELECT IF(GROUPING(terms_id), 'Terms ID Total', terms_id) AS 'Terms ID', IF(GROUPING(...	40 row(s) returned

23. ***(10 points)*** Write a SELECT statement that answers this question: How many vendors are there for each city in the states of IA and NJ? Return these columns:
    - vendor_state from the Vendors table
    - vendor_city from the Vendors table
    - qty_vendors for how many vendors for each combination of vendor_state and vendor_city

    Use the WITH ROLLUP operator to include rows that give a summary for each vendor_state as well as a row that gives the grand total.

    Use the IF and GROUPING functions to replace the null values in the vendor_state and vendor_city columns with literal values if they are for summary rows. **Paste a screenshot of the complete resultset.** ***(8 rows including all subtotal rows and the grand total row)***

```
184     -- Question 23
185 •   SELECT
186         IF(GROUPING(vendor_state), 'Vendor State Total', vendor_state) AS 'Vendor State',
187         IF(GROUPING(vendor_city), 'Vendor City Total', vendor_city) AS 'Vendor City',
188         count(*) AS 'Qty Vendors'
189     FROM VENDORS
190     WHERE vendor_state = 'IA' OR vendor_state = 'NJ'
191     GROUP BY vendor_state, vendor_city WITH ROLLUP;
```

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

Vendor State	Vendor City	Qty Vendors
IA	Fairfield	1
IA	Washington	1
IA	Vendor City Total	2
NJ	East Brunswick	2
NJ	Fairfield	1
NJ	Washington	1
NJ	Vendor City Total	4
Vendor State Total	Vendor City Total	6

Result 273

Output

Action Output

#	Time	Action	Message
1	11:44:50	SELECT IF(GROUPING(vendor_state), 'Vend...	8 row(s) returned