Assignment 3

Name \_\_\_\_\_\_\_\_\_\_\_Christopher Holmes\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

How to retrieve data using multiple tables and perfrom summaries of data

* Read chapter 4 on how to retrieve data from two or more tables using join and union operations
* Do exercises 1-7 at end of chapter 4
* Read chapter 6 on how to code summary queries using aggregation function AVG, SUM, MIN, MAX and COUNT, and use the GROUP BY and HAVING clauses
* Do exercises 1-7 at end of chapter 6
* Read chapter 7 on how to code subqueries
* Do exercies at end of chapter 7
* Read chapter 8 about data types and how to convert or cast data from one data type to another and use the FORMAT and CHAR functions
* Read chapter 9 about how to work with string and date-time data types in SQL statements

Copy and paste your working select statement from chapter 4 exercises here.

1-

|  |
| --- |
| select \*  from vendors inner join invoices  on vendors.vendor\_id = invoices.vendor\_id  order by invoice\_number; |

2-

|  |
| --- |
| select vendor\_name as V, invoice\_number as I, invoice\_date, invoice\_total - payment\_total - credit\_total as balance\_due  from vendors inner join invoices  on vendors.vendor\_id = invoices.vendor\_id  where invoice\_total - payment\_total - credit\_total > 0  order by vendor\_name asc; |

3-

|  |
| --- |
| select vendor\_name, default\_account\_number as default\_account, account\_description as description  from vendors inner join general\_ledger\_accounts  on vendors.default\_account\_number = general\_ledger\_accounts.account\_number  order by account\_description, vendor\_name |

4-

|  |
| --- |
| select vendor\_name, invoice\_date, invoice\_number,  invoice\_sequence as li\_sequence,  line\_item\_amount as li\_amount  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  order by vendor\_name, invoice\_date, invoice\_number, invoice\_sequence |

5-

|  |
| --- |
| select v1.vendor\_id, v1.vendor\_name, concat(v1.vendor\_contact\_first\_name, ' ', v1.vendor\_contact\_last\_name) as contact\_name  from vendors v1 join vendors v2  on v1.vendor\_id <> v2.vendor\_id and  v1.vendor\_contact\_last\_name = v2.vendor\_contact\_last\_name  order by v1.vendor\_contact\_last\_name |

6-

|  |
| --- |
| select g.account\_number, account\_description, invoice\_id  from general\_ledger\_accounts g left join invoice\_line\_items i  on g.account\_number = i.account\_number  where invoice\_id is null  order by g.account\_number |

7-

|  |
| --- |
| select vendor\_name, vendor\_state  from vendors  where vendor\_state = 'CA'  union  select vendor\_name, 'Outside CA'  from vendors  where vendor\_state <> 'CA'  order by vendor\_name |

Copy and paste your working select statements from chapter 6 exercises here.

1-

|  |
| --- |
| select vendor\_id, sum(invoice\_total) as invoice\_total\_sum  from invoices  group by vendor\_id |

2-

|  |
| --- |
| select vendor\_name, sum(payment\_total) as payment\_total\_sum  from vendors join invoices  on vendors.vendor\_id = invoices.vendor\_id  group by vendor\_name  order by payment\_total\_sum desc |

3-

|  |
| --- |
| select vendor\_name, count(\*) as invoice\_count, sum(invoice\_total) as invoice\_total\_sum  from vendors join invoices  on vendors.vendor\_id = invoices.vendor\_id  group by vendor\_name  order by invoice\_count desc |

4-

|  |
| --- |
| select account\_description, count(\*) as count\_invoices, sum(line\_item\_amount) as sum\_line  from general\_ledger\_accounts gl join invoice\_line\_items il  on gl.account\_number = il.account\_number  group by gl.account\_description  having count(\*) > 1  order by sum\_line desc |

5-

|  |
| --- |
| select account\_description, count(\*) as count\_invoices, sum(line\_item\_amount) as sum\_line  from general\_ledger\_accounts gl join invoice\_line\_items il  on gl.account\_number = il.account\_number  join invoices i  on il.invoice\_id = i.invoice\_id  where invoice\_date between '2011/04/01' and '2011/06/30'  group by gl.account\_description  having count(\*) > 1  order by sum\_line desc |

6-

|  |
| --- |
| select il.account\_number, sum(line\_item\_amount) as sum\_line  from invoice\_line\_items il join general\_ledger\_accounts gl  where il.account\_number = gl.account\_number  group by il.account\_number  with rollup |

7-

|  |
| --- |
| select vendor\_name, count(distinct il.account\_number) as 'Number of Accounts'  from vendors v join invoices i  on v.vendor\_id = i.vendor\_id  join invoice\_line\_items il  on i.invoice\_id = il.invoice\_id  group by vendor\_name  having count(distinct il.account\_number)>1  order by vendor\_name |

Copy and paste your working select statement with subqueries from chapter 7 here.

1-

|  |
| --- |
| select vendor\_name  from vendors  where vendor\_id in (select vendor\_id from invoices)  order by vendor\_name |

2-

|  |
| --- |
| select invoice\_number, invoice\_total  from invoices  where payment\_total > (select avg(payment\_total) from invoices where payment\_total > 0)  order by invoice\_total desc |

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 |

4-

|  |
| --- |
| select vendor\_name, i.invoice\_id, invoice\_sequence, line\_item\_amount  from vendors v join invoices i  on v.vendor\_id = i.vendor\_id  join invoice\_line\_items il  on i.invoice\_id = il.invoice\_id  where i.invoice\_id in (select invoice\_id from invoice\_line\_items where invoice\_sequence > 1)  order by vendor\_name, i.invoice\_id, invoice\_sequence |

5-

|  |
| --- |
|  |