

Due Date: Nov 04, 2015 11:00 PM (Late date Nov 05, 11:00 pm)

Points: 35 point max

General Directions

Use the books tables.

These tasks focus on the use of Subqueries. Consequently, you **must use subqueries** to solve the problems. In many cases you could solve the task without the use of subqueries- **but that will not earn any credit** for the assignment.

The following rules must be followed.

- You must do each task with a single query. The query will have sub components but it will be a single query. For a few tasks you might need to use the same query with different values for the filters.
- Do not use a join of any kind. Do not use a correlated subquery. Do not use a comma join.
- Do not use any views.
- Do not use set operations (Union, intersect, etc)
- You may use a CTE, since that is actually a subquery.
- Do not use a variable except for assignment of a simple literal.
- You may not use ranking functions (we use them in a future unit)
- If you use column aliases, then use different column aliases in each subquery. Using the same column alias may be legit but it makes your query very hard to read.
- Do not hard code a literal for the current year, the last year etc.
- If you are testing a date column, use temporal functions. **Do not use wildcard for testing dates. You may use to_char**
- Use only the data supplied in the task to write the query and do not make assumption about the data that are not supported by the create table statements.
- If a task asks you to display the 'largest', 'fewest' etc, display any ties for that position.
- A book with an order quantity of 0 is still considered a book that is ordered
- **An order header without any detail lines is still an order.**

Queries that use sub queries tend to be longer and harder to read if they are not formatted properly. Queries that are hard to read will lose points. The key words From, Where, Group By, Order By, Having start new lines and need to align. Subqueries are indented; the indentation should be 2 or 3 spaces. See examples of formatting below

Formatting subqueries: Acceptable format for a subquery. Note that you can easily find the subquery.

```
select cust_id, ord_id
, ( select sum(quantity_ordered)
    from oe_order_details OD
    where OH.ord_id = OD.ord_id
  ) as "NumItemsPerOrder"
from oe_order_headers OH;
```

Not acceptable- the subquery runs on a single line making it harder to see the components

```
select cust_id, ord_id,
( select sum(quantity_ordered) from oe_order_details OD where OH.ord_id = OD.ord_id) as
"NumItemsPerOrder"
from oe_order_headers OH;
```

Not acceptable: The Select for the subquery needs to be indented a few spaces. It is hard to see where the the subquery starts and ends.

```
select cust_id, ord_id, (
select sum(quantity_ordered)
from oe_order_details OD
where OH.ord_id = OD.ord_id) as "NumItemsPerOrder"
from oe_order_headers OH;
```

Not acceptable: The Select for the subquery needs to be indented a few spaces- not a lot of spaces. This also is hard to read.

```
select cust_id, ord_id, (
select sum(quantity_ordered)
from oe_order_details OD
where OH.ord_id = OD.ord_id) as "NumItemsPerOrder"
from oe_order_headers OH;
```

Tasks

If you have decided to try the queries without reading the directions, go back and read the directions.

- Task 01:** Display the book id and title for any books which someone has ordered and the book is categorized as **either** an SQL book and a database book or possibly both topics. Use the topic_id to filter for DB and SQL. Sort by the book_id.
- Task 02:** Display the book id and title for any books which someone has ordered and the book is categorized as **both** an SQL book and a database book. Use the topic_id to filter for DB and SQL. Sort by the book_id.
- Task 03:** Display the book id and title for any books which someone has ordered and the same book is categorized as an SQL book but it is **not** categorized as a database book. Use the topic_id to filter for DB and SQL. Sort by the book_id.
- Task 04:** Display the ID, title of the books and the publication year for the book with the largest sales amount; include ties. For this query, use the total extended cost when determining the sales of a book. Sort by the book_id.
- Task 05:** Display the ID and last name of the customers who have bought any ART books in the current year. Use the topic id to filter for ART books. Sort by the cust_id. Copy the query and use it to filter for history (HIST) books.
- Task 06:** Display the book id and title for any books where we have orders for more than 500 copies of the book. Use the quantity attribute. Sort by the book_id.
- Task 07:** Display the year and month which has the fewest orders. This analysis only considers year/month where we have any orders. You can display the result as two columns (year, month) or as a single column in a format such as '2014.11' or '2014 Nov'. You may not use the literal 1 in your query.
- Task 08:** Display the year and month which has the most orders. This analysis only considers year/month where we have any orders. You can display the result as two columns (year, month) or as a single column in a format such as '2014.11' or '2014 Nov'.