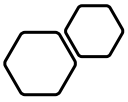

Subquery Challenges

1

First Challenge

Challenge:

Return the following product details for the cheapest product(s) in the `oes.products` table:

- `product_id`
- `product_name`
- `list_price`
- `category_id`

2

Second Challenge

Challenge:

Use a correlated subquery to return the following product details for the cheapest product(s) in each product category as given by the `category_id` column:

- `product_id`
- `product_name`
- `list_price`
- `category_id`

3

Third Challenge

Challenge:

Return the same result as challenge 2 i.e. the cheapest product(s) in each product category except this time by using an inner join to a derived table.

4

Fourth Challenge

Challenge:

Return the same result as challenge 2 and 3 i.e. the cheapest product(s) in each product category except this time by using a common table expression.

5

Fifth Challenge

Challenge:

Repeat challenge 4, except this time include the product category name as given in the `oes.product_categories` table.

6

Sixth Challenge

Background:

The `employee_id` column in the `oes.orders` table gives the `employee_id` of the salesperson who made the sale.

Challenge:

Use the NOT IN operator to return all employees who have never been the salesperson for any customer order. Include the following columns from `hcm.employees`:

- employee_id
- first_name
- last_name

7

Seventh Challenge

Challenge:

Return the same result as challenge 6, except use WHERE NOT EXISTS

8

Eighth Challenge

Challenge:

Return unique customers who have ordered the 'PBX Smart Watch 4'. Include:

- customer_id
- first_name
- last_name
- email

9