Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

**ЗВІТ**

про виконання комп’ютерного практикуму № 3

з дисципліни

“ OLAP та сховища даних ”

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Олійник Ю. О. |  | студент 3-го курсу  гр. ІП-51 ФІОТ  Зарічковий Олександр  Анатолійович |

Київ – 2017

**ЗМІСТ:**

[1 Завдання 3](#_Toc495230724)

[2 Виконання Завдання 4](#_Toc495230725)

# Завдання

Завдання 1

Виконати запуск скриптів KP3\script(create\_scheme).sql та КР3\tables\_data.sql

Необхідно сформувати наступні аналітичні запити:

1. Показати ранг кожного товару у групі (відповідно до зменшення ціни в групі). Запит реалізувати через функції rank() та dense\_rank(). Порівняти результати виконання.
2. Вивести список із двох найдешевших товарів у кожній групі.
3. Показати які товари по кожній групі мають найбільші продажі .
4. Вивести список товарів з найбільшою вартістю, які найдовше зберігаються на полицях складу.
5. Побудувати запит з використанням функцій LEAD. Той же запит реалізувати через функцію LAG. Порівняти результати їх виконання.
6. Побудувати аналітичні запити, які б використовували наступні функцій (один запит – одна функція):

* FIRST\_VALUE (або LAST\_VALUE)
* MAX ( або MIN)
* COUNT( або SUM).

1. Вивести список товарів, які було придбано за певний період часу вивести вартість товарів по різним вікнам використовуючи наступну структуру :
2. …over( order by )
3. …over (partition … order by… )
4. … over (partition col1, col2)
5. … over (partition col1, col2… order by)

**Предметна область**: Меблевий салон

# Виконання Завдання

1. Завдання по написанню запитів
   1. Показати ранг кожного товару у групі (відповідно до зменшення ціни в групі). Запит реалізувати через функції rank() та dense\_rank(). Порівняти результати виконання.

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

t.name type,

RANK() OVER

(PARTITION BY p.typeId ORDER BY p.price DESC) rank,

DENSE\_RANK() OVER

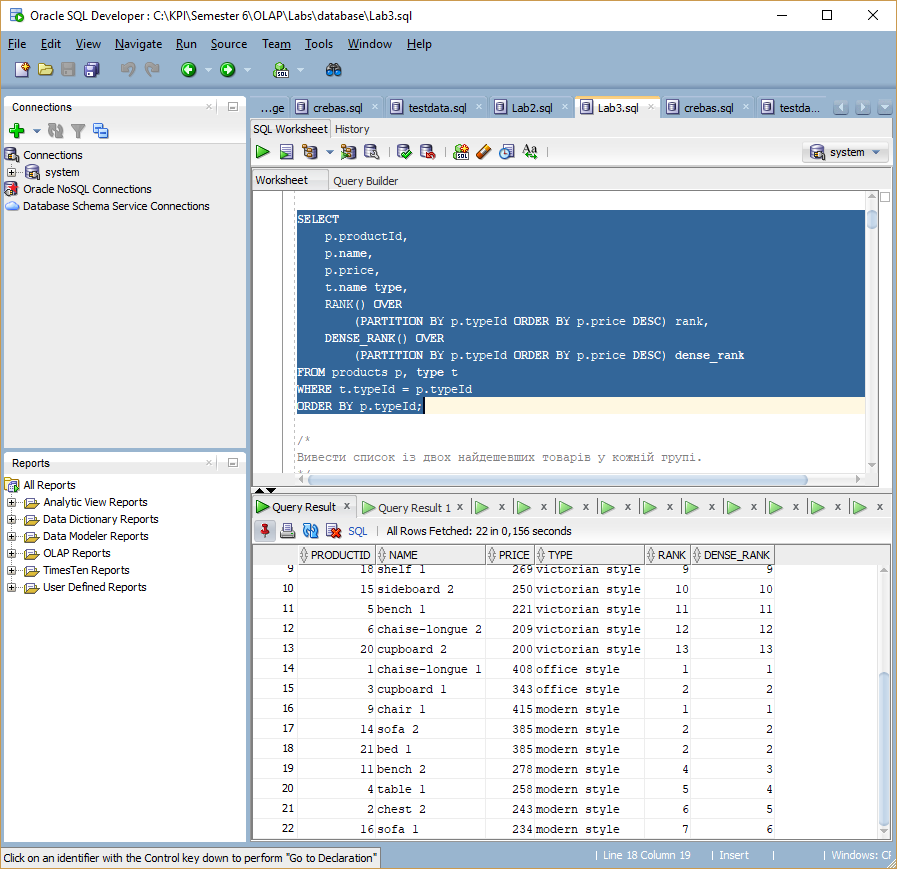
(PARTITION BY p.typeId ORDER BY p.price DESC) dense\_rank

FROM products p, type t

WHERE t.typeId = p.typeId

ORDER BY p.typeId;

*Результат*:



* 1. Вивести список із двох найдешевших товарів у кожній групі.

*Текст запиту*:

SELECT \* FROM

(SELECT

p.productId,

p.name,

p.price,

t.name type,

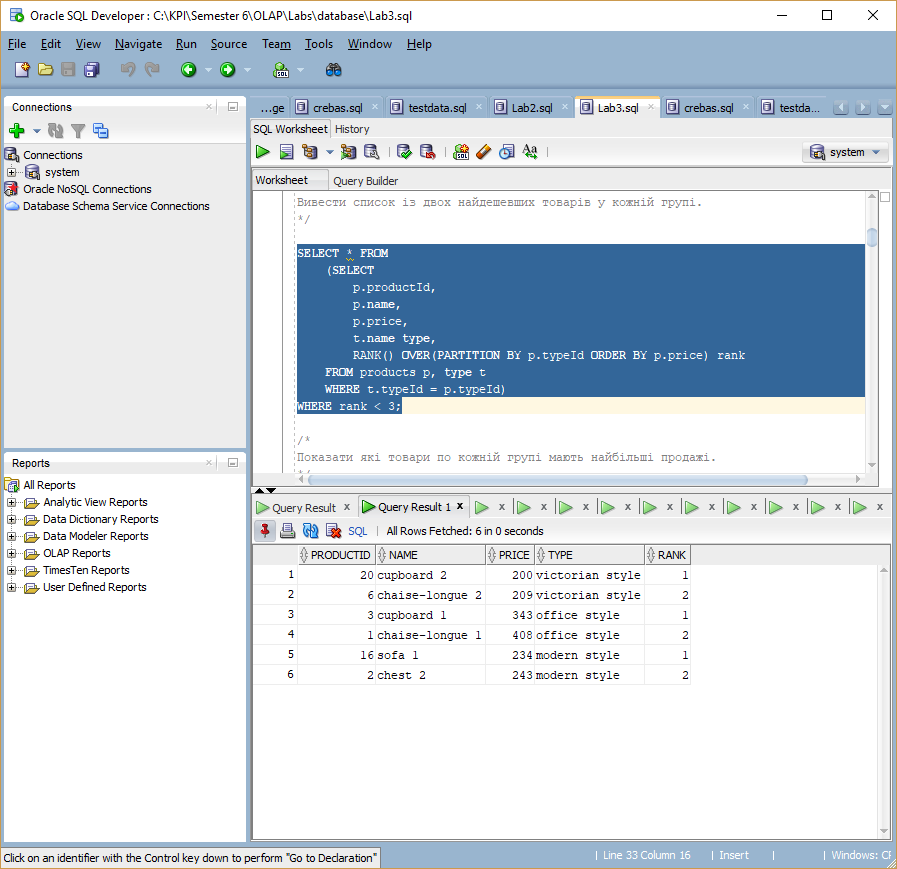
RANK() OVER(PARTITION BY p.typeId ORDER BY p.price) rank

FROM products p, type t

WHERE t.typeId = p.typeId)

WHERE rank < 3;

*Результат*:



* 1. Показати які товари по кожній групі мають найбільші продажі.

*Текст запиту*:

SELECT \* FROM

(SELECT

productId,

name,

price,

type,

allQuantity,

profit,

RANK() OVER(PARTITION BY type ORDER BY profit DESC) rank

FROM

(SELECT DISTINCT

p.productId,

p.name,

p.price,

t.name type,

SUM(sl.quantity) OVER

(PARTITION BY p.productId) allQuantity,

SUM(p.price \* sl.quantity) OVER

(PARTITION BY p.productId) profit

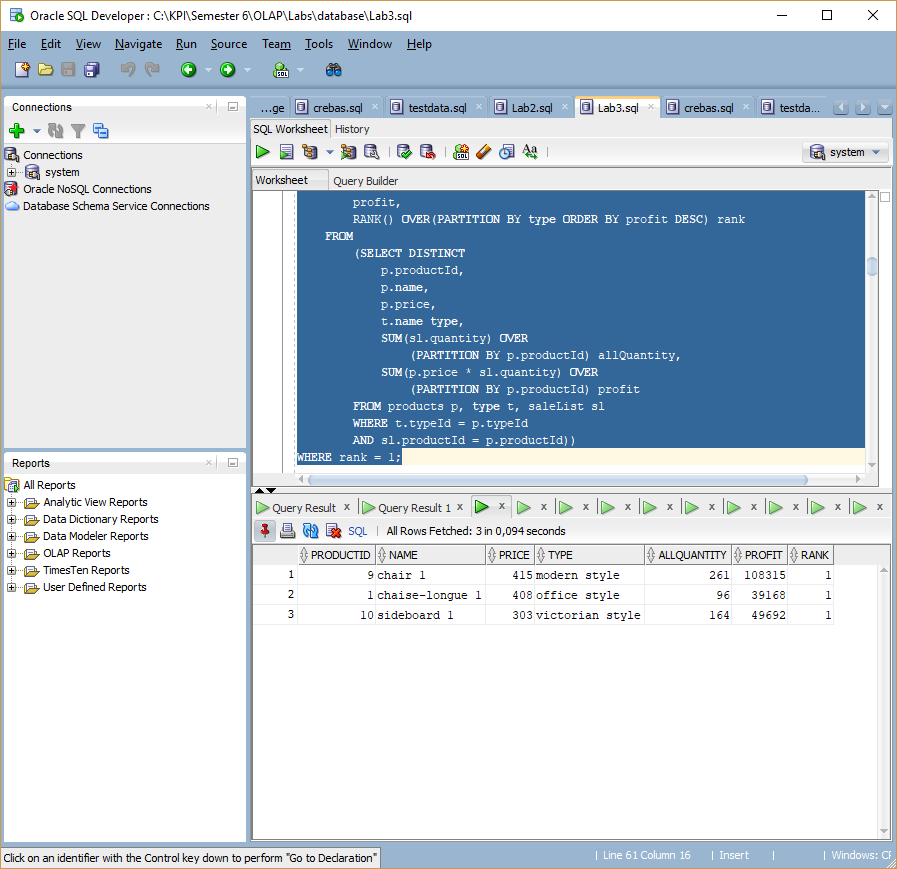
FROM products p, type t, saleList sl

WHERE t.typeId = p.typeId

AND sl.productId = p.productId))

WHERE rank = 1;

*Результат виконання*:



* 1. Вивести список товарів з найбільшою вартістю, які найдовше зберігаються на полицях складу.

*Текст запиту*:

SELECT

st.productId,

p.price,

st.quantity,

st.shelf,

su.supplyDate

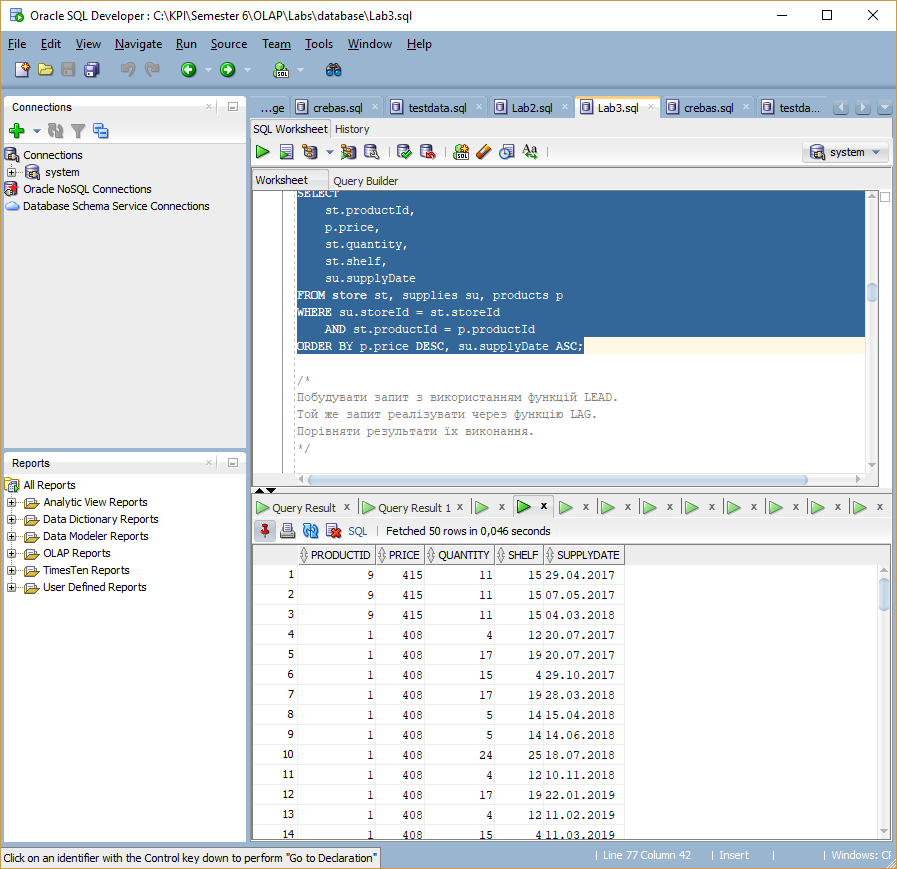
FROM store st, supplies su, products p

WHERE su.storeId = st.storeId

AND st.productId = p.productId

ORDER BY p.price DESC, su.supplyDate ASC;

*Результат виконання*:



* 1. Побудувати запит з використанням функцій LEAD. Той же запит реалізувати через функцію LAG. Порівняти результати їх виконання.

*Текст запиту*:

SELECT

su1.supplierId,

su2.name,

su1.supplyDate,

LEAD(su1.supplyDate, 1) OVER

(PARTITION BY su1.supplierId ORDER BY su1.supplierId) lead,

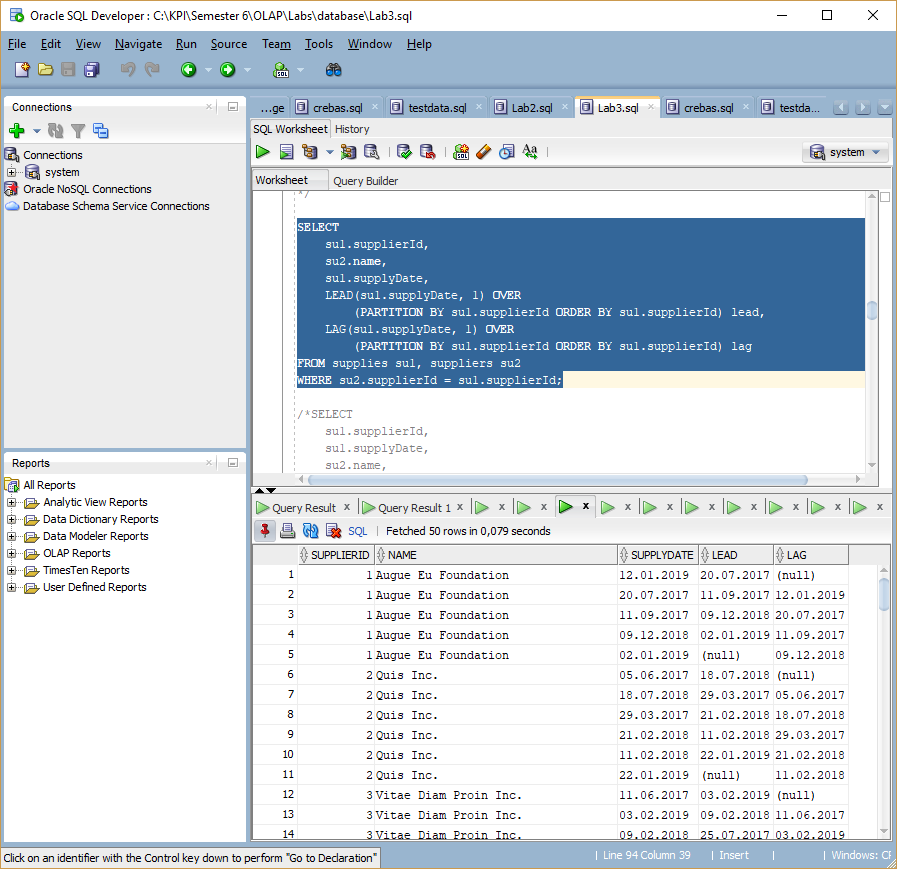
LAG(su1.supplyDate, 1) OVER

(PARTITION BY su1.supplierId ORDER BY su1.supplierId) lag

FROM supplies su1, suppliers su2

WHERE su2.supplierId = su1.supplierId;

*Результат виконання*:



* 1. Побудувати аналітичні запити, які б використовували наступні функції (один запит – одна функція):
* FIRST\_VALUE (або LAST\_VALUE)

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

t.name type,

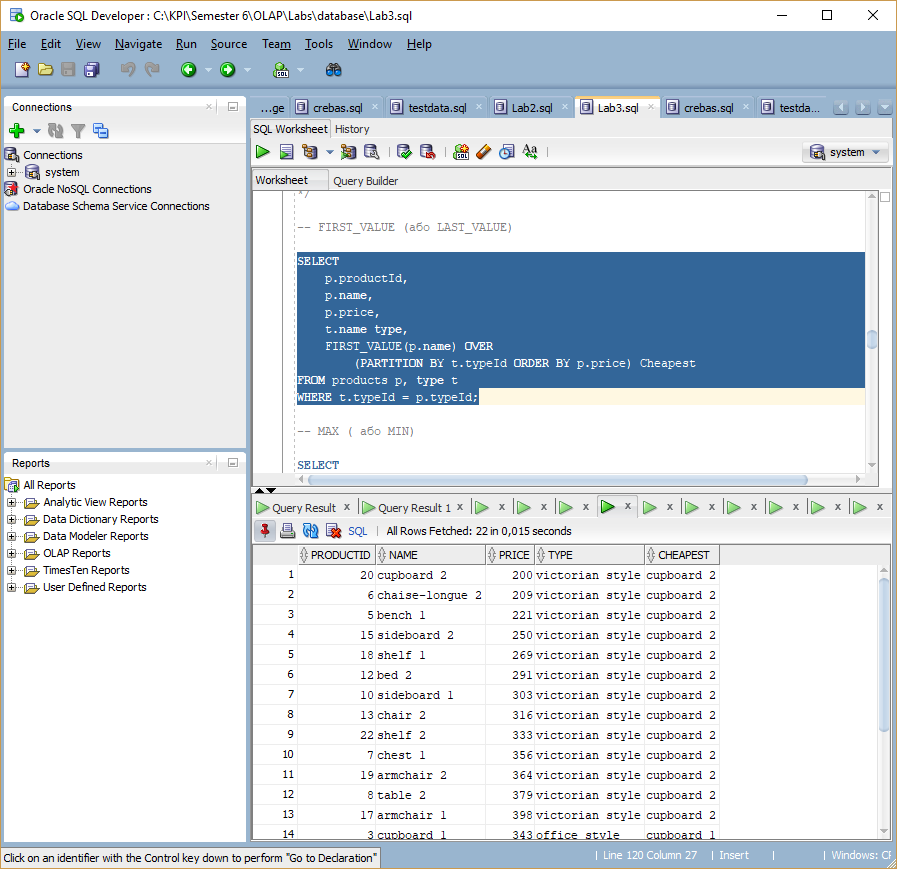
FIRST\_VALUE(p.name) OVER

(PARTITION BY t.typeId ORDER BY p.price) Cheapest

FROM products p, type t

WHERE t.typeId = p.typeId;

*Результат виконання*:



* MAX (або MIN)

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

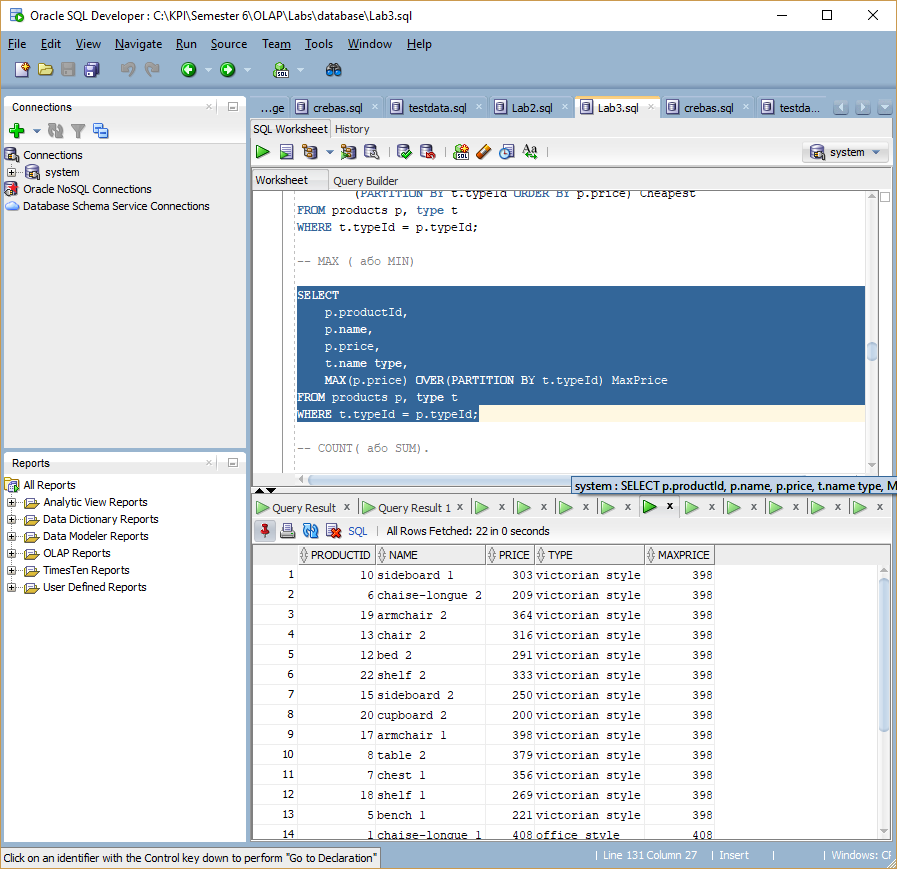
t.name type,

MAX(p.price) OVER(PARTITION BY t.typeId) MaxPrice

FROM products p, type t

WHERE t.typeId = p.typeId;

*Результат виконання*:



* COUNT( або SUM)

*Текст запиту*:

SELECT DISTINCT

t.typeId,

t.name type,

SUM(p.price \* sl.quantity) OVER

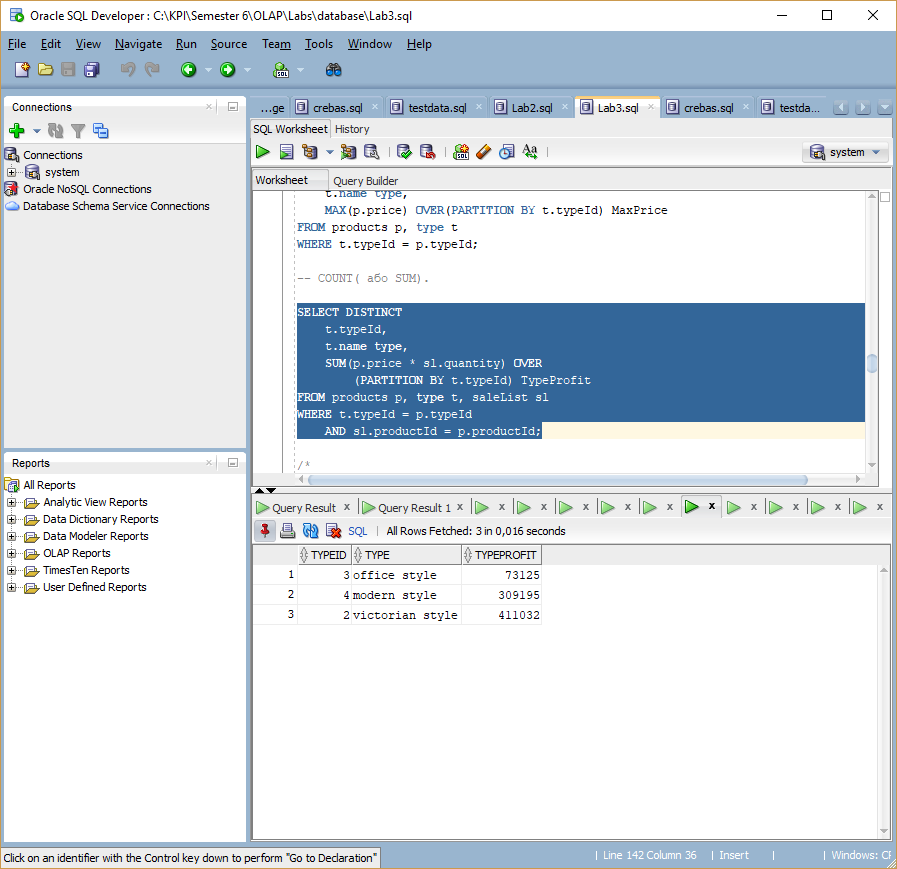
(PARTITION BY t.typeId) TypeProfit

FROM products p, type t, saleList sl

WHERE t.typeId = p.typeId

AND sl.productId = p.productId;

*Результат виконання*:



* 1. Вивести список товарів, які було придбано за певний період часу вивести вартість товарів по різним вікнам використовуючи наступну структуру:
* … over( order by )

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

sl.quantity,

SUM(p.price \* sl.quantity) OVER

(PARTITION BY sl.saleListId) profit,

sa.saleDate,

SUM(p.price \* sl.quantity)

OVER(ORDER BY sa.saleDate range

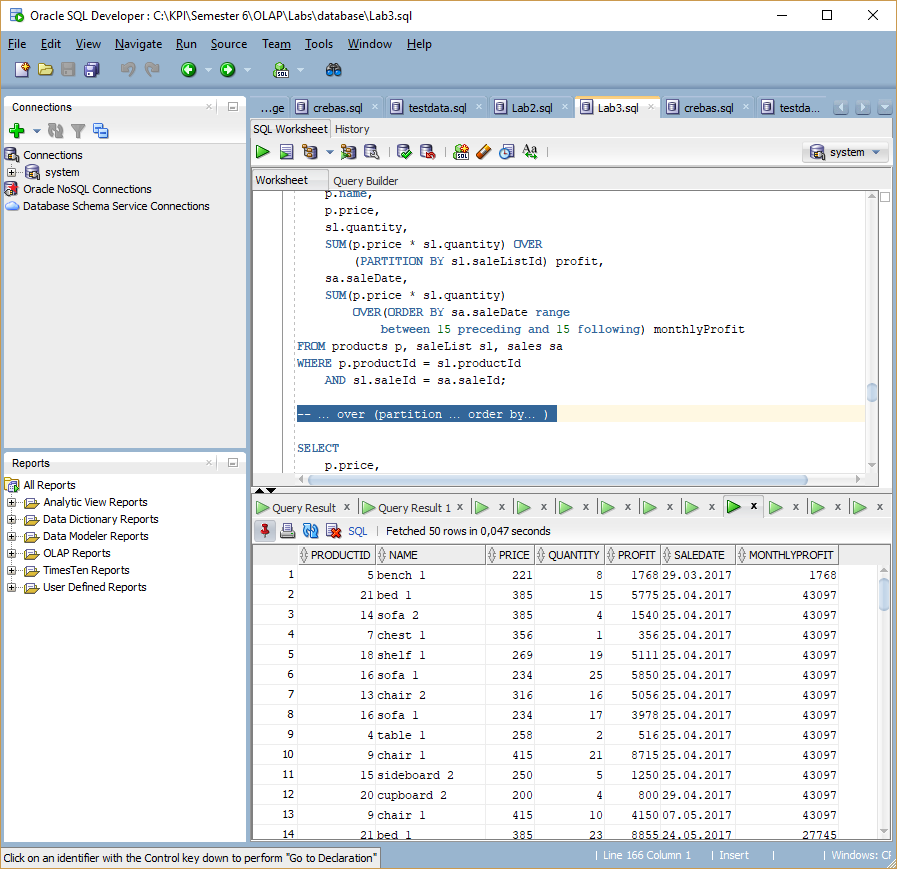
between 15 preceding and 15 following) monthlyProfit

FROM products p, saleList sl, sales sa

WHERE p.productId = sl.productId

AND sl.saleId = sa.saleId;

*Результат виконання*:



* … over (partition … order by… )

*Текст запиту*:

SELECT

p.price,

p.typeId,

sl.quantity,

SUM(p.price \* sl.quantity) OVER

(PARTITION BY sl.saleListId) profit,

sa.saleDate,

SUM(p.price \* sl.quantity)

OVER(PARTITION BY p.typeId ORDER BY sa.saleDate

range between 45 preceding and 45 following) quarterlyProfitByType

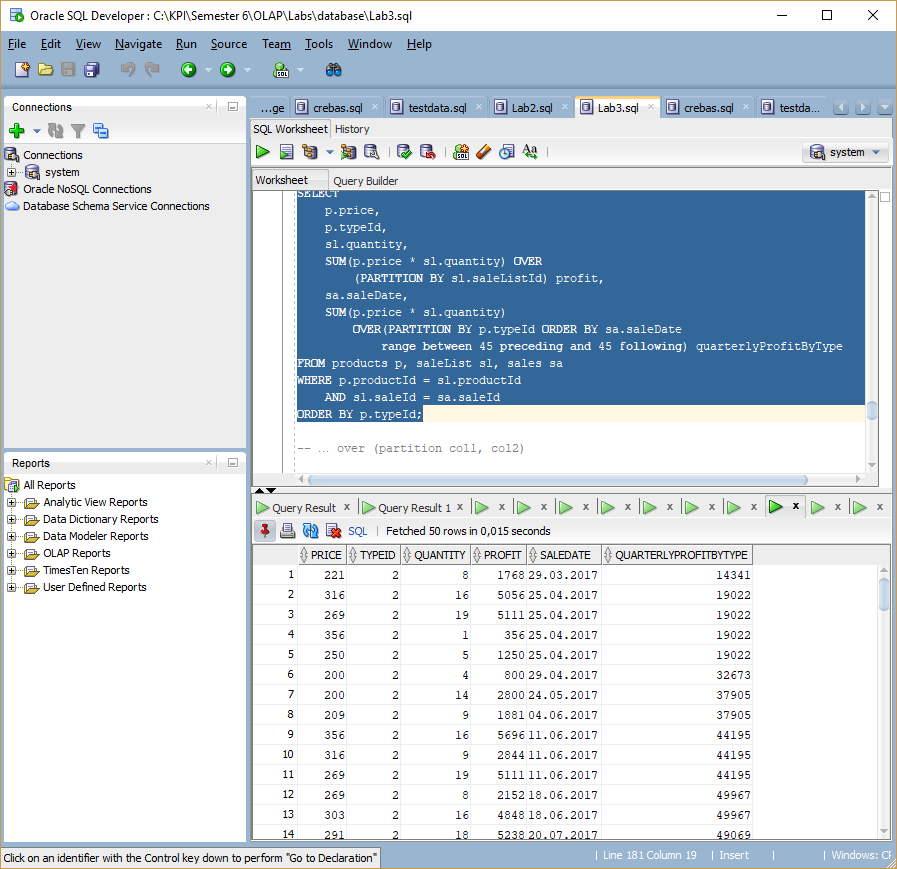
FROM products p, saleList sl, sales sa

WHERE p.productId = sl.productId

AND sl.saleId = sa.saleId

ORDER BY p.typeId;

*Результат виконання*:



* … over (partition col1, col2)

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

p.typeId,

sl.quantity,

SUM(p.price \* sl.quantity) OVER

(PARTITION BY sl.saleListId) profit,

sa.saleDate,

SUM(p.price \* sl.quantity)

OVER(PARTITION BY p.typeId, p.productId) quarterlyProfitByType

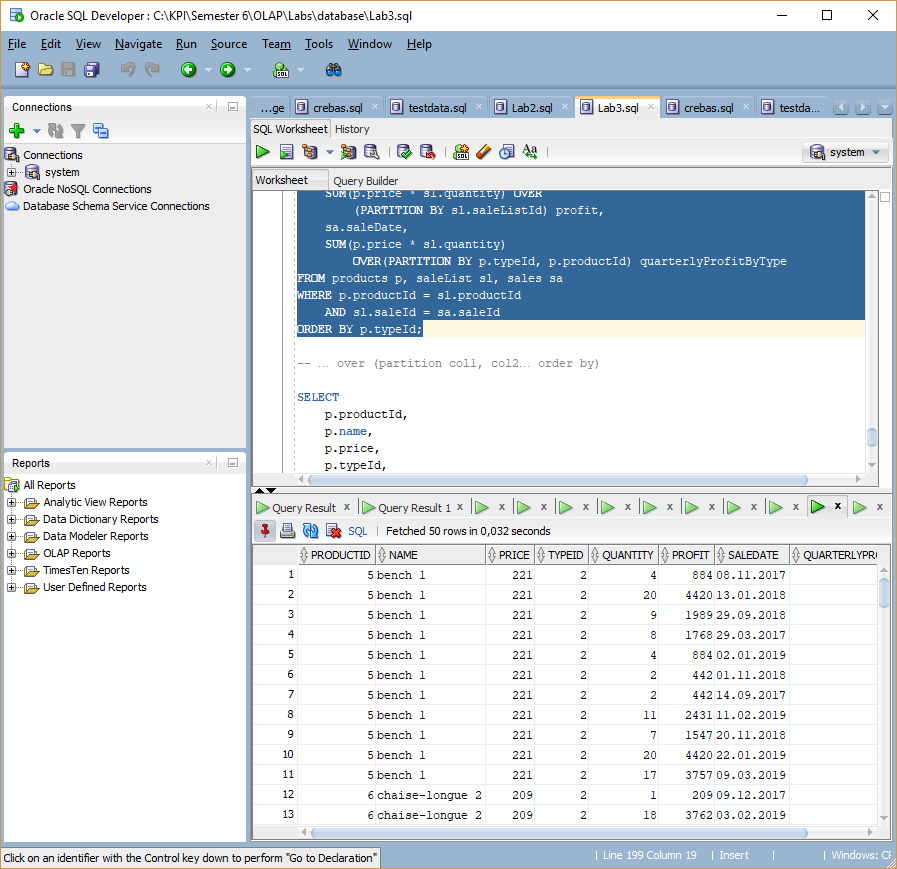
FROM products p, saleList sl, sales sa

WHERE p.productId = sl.productId

AND sl.saleId = sa.saleId

ORDER BY p.typeId;

*Результат виконання*:



* … over (partition col1, col2… order by)

*Текст запиту*:

SELECT

p.productId,

p.name,

p.price,

p.typeId,

sl.quantity,

SUM(p.price \* sl.quantity) OVER

(PARTITION BY sl.saleListId) profit,

sa.saleDate,

SUM(p.price \* sl.quantity)

OVER(PARTITION BY p.typeId, p.productId ORDER BY sa.saleDate

range between 45 preceding and 45 following) quarterlyProfitByType

FROM products p, saleList sl, sales sa

WHERE p.productId = sl.productId

AND sl.saleId = sa.saleId

ORDER BY p.typeId;

*Результат виконання*:

