Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

**ЗВІТ**

про виконання комп’ютерного практикуму № 2

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. Перепроектування бази даних

Згідно наданих варіантів предметних областей:

1. Засобами PowerDesigner допроектуйте модель KP2\task2.pdm, що відносяться до опису товарів.
2. Для всіх таблиць проставте первинні та зовнішні ключі.
3. Проставте «обов’язковість» заповнення даних для необхідних полів.
4. Пропишіть «обмеження» для полів даних.
5. Спроектуйте не менше ніж 5 індексів.
6. Згенеруйте фізичну модель даних.
7. Створіть процедуру заповніть таблиць тестовими даними. Для довідникової інформації не менше ніж 10 записів, для даних з продажу не менше ніж 1000 записів.

Завдання 2. Написання SQL запитів

1. Вивести сумарну вартість поставки по кожному продукту.
2. Визначити, товари якого постачальник жодного разу не купувались.
3. Визначити, які товари ніколи не продаються у другому півріччі року.
4. Визначити 5 покупців, що найчастіше відвідують магазин.
5. Визначити працівника, що продає найбільше товарів.
6. Визначити 3 працівника, що оформили найбільше чеків.
7. Яка суму в середньому сплачує покупець за одне відвідування магазину.
8. Скільки покупок оформлює кожен продавець за день.
9. Визначити які полиці зберігають товари лише одного типу.
10. Визначити товари, що мають найбільшу ціну та найчастіше продаються.

**Предметна область**: Меблевий салон

# Виконання Завдання

1. Завдання по перепроектуванню бази даних



*Опис таблиць:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Назва таблиці** | **Назва атрибуту** | **Тип даних** | **Primary Key** | **Mandatory** |
| Products | ProductId | INTEGER | X | X |
| Name | VARCHAR2(256) |  | X |
| Price | INTEGER |  | X |
| TypeId | INTEGER |  | X |
| SaleList | SaleListId | INTEGER | X | X |
| SaleId | INTEGER |  | X |
| ProductId | INTEGER |  | X |
| Quantity | INTEGER |  | X |
| Sales | SaleId | INTEGER | X | X |
| StaffId | INTEGER |  | X |
| CustomerId | INTEGER |  | X |
| SaleDate | DATE |  | X |
| Staff | StaffId | INTEGER | X | X |
| PositionId | INTEGER |  |  |
| Name | VARCHAR2(256) |  | X |
| Surname | VARCHAR2(256) |  | X |
| Phone | VARCHAR2(256) |  |  |
| Positions | PositionId | INTEGER | X | X |
| Name | VARCHAR2(256) |  | X |
| Salary | INTEGER |  | X |
| Customers | CustomerId | INTEGER | X | X |
| Name | VARCHAR2(256) |  | X |
| Surname | VARCHAR2(256) |  | X |
| Address | VARCHAR2(256) |  |  |
| Supplies | SupplyId | INTEGER | X | X |
| SupplierId | INTEGER |  | X |
| SupplyDate | DATE |  | X |
| StoreId | INTEGER |  | X |
| Suppliers | SupplierId | INTEGER | X | X |
| Name | VARCHAR2(256) |  | X |
| Address | VARCHAR2(256) |  |  |
| Email | VARCHAR2(256) |  |  |
| Store | StoreId | INTEGER | X | X |
| ProductId | INTEGER |  | X |
| Quantity | INTEGER |  | X |
| Shelf | INTEGER |  | X |
| Type | TypeId | INTEGER | X | X |
| Name | VARCHAR2(256) |  | X |

1. Завдання по написанню запитів
   1. Вивести сумарну вартість поставки по кожному продукту.

*Текст запиту*:

SELECT

s.productId,

p.name,

p.price,

SUM(s.quantity) QUANTITY,

SUM(s.quantity \* p.price) ALL\_PRICE

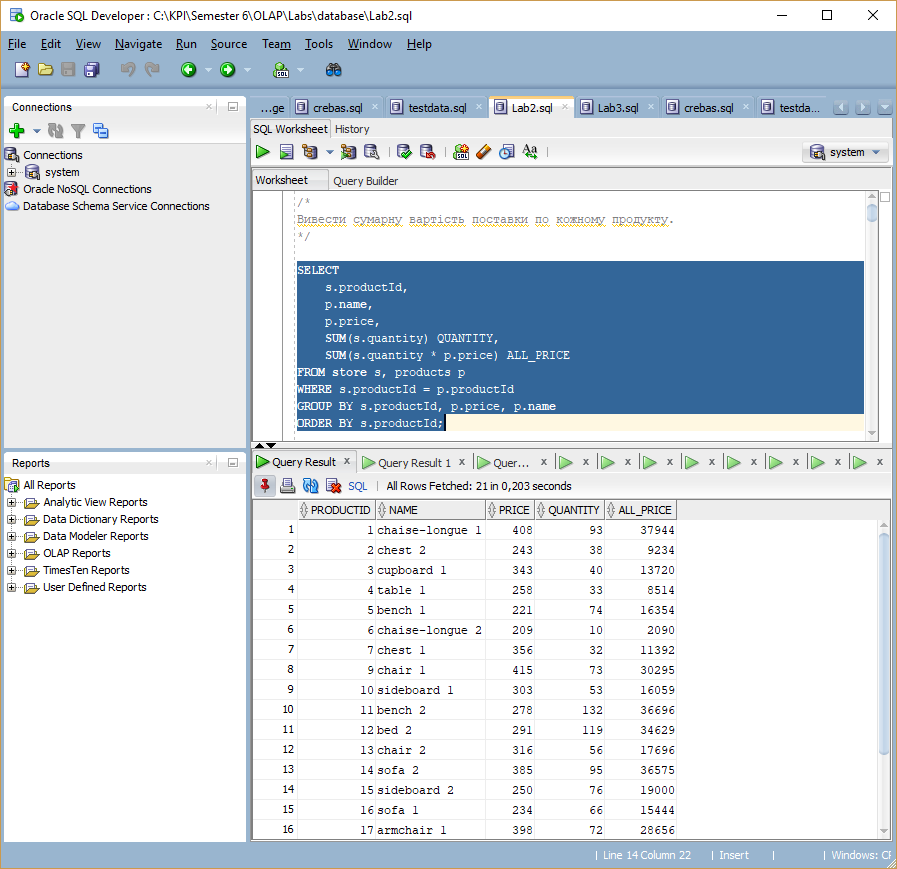
FROM store s, products p

WHERE s.productId = p.productId

GROUP BY s.productId, p.price, p.name

ORDER BY s.productId;

*Результат*:



* 1. Визначити, товари якого постачальник жодного разу не купувались.

*Текст запиту*:

SELECT \* FROM suppliers

WHERE supplierId NOT IN

(SELECT DISTINCT

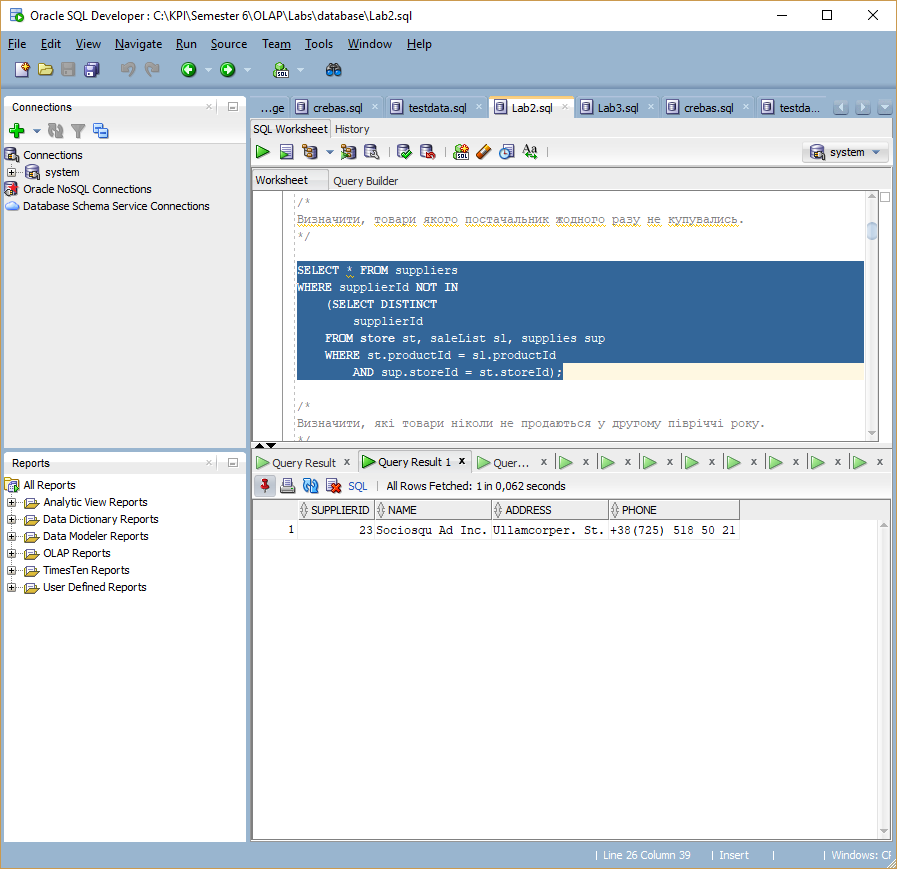
supplierId

FROM store st, saleList sl, supplies sup

WHERE st.productId = sl.productId

AND sup.storeId = st.storeId);

*Результат*:



* 1. Визначити, які товари ніколи не продаються у другому півріччі року.

*Текст запиту*:

SELECT \* FROM products

WHERE productId NOT IN

(SELECT DISTINCT

sl.productId

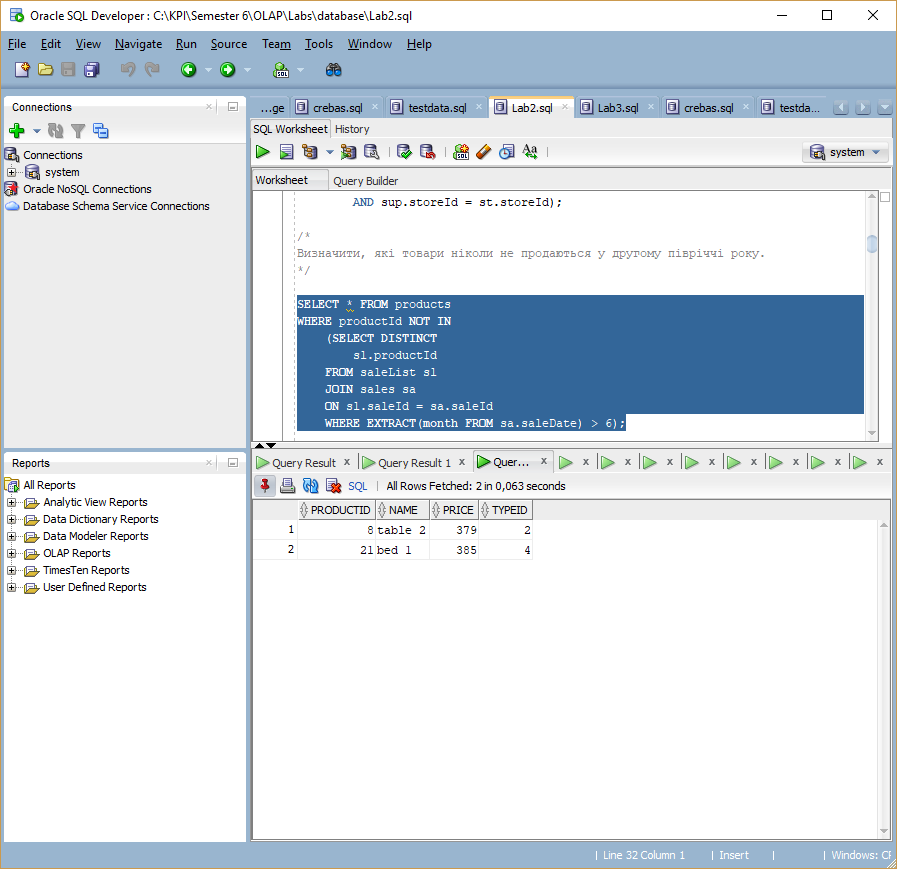
FROM saleList sl

JOIN sales sa

ON sl.saleId = sa.saleId

WHERE EXTRACT(month FROM sa.saleDate) > 6);

*Результат виконання*:



* 1. Визначити 5 покупців, що найчастіше відвідують магазин.

*Текст запиту*:

SELECT \* FROM

(SELECT

c.customerId,

c.name,

c.surname,

count(saleId) AS "Count"

FROM sales s

JOIN customers c

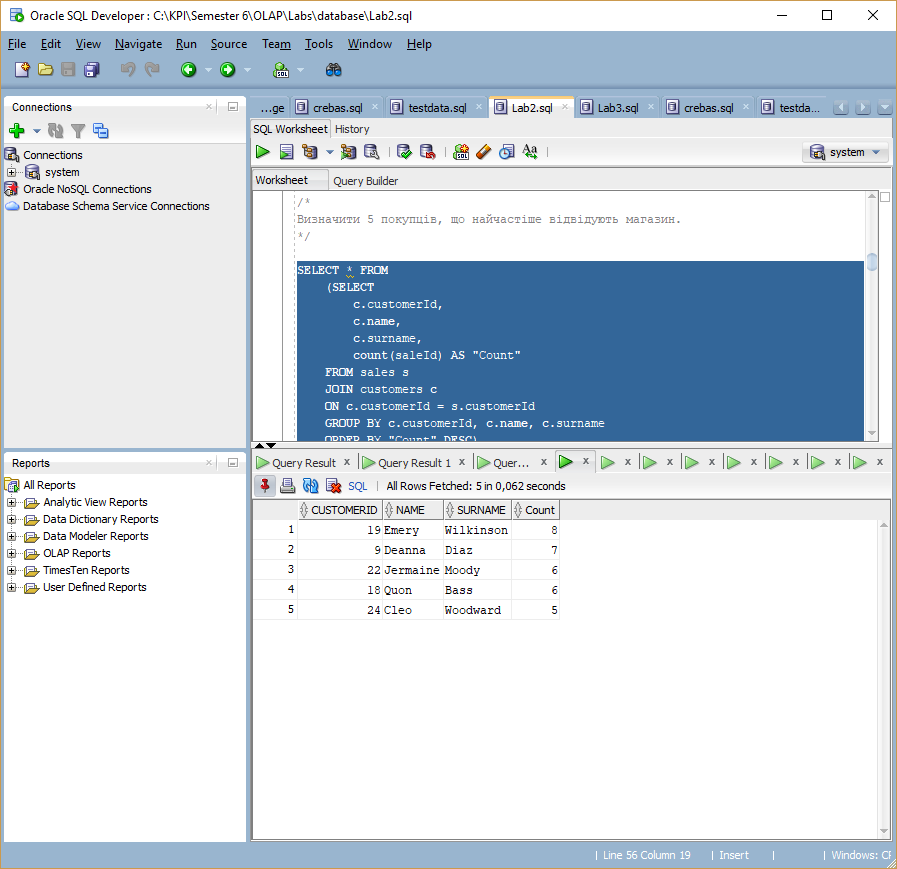
ON c.customerId = s.customerId

GROUP BY c.customerId, c.name, c.surname

ORDER BY "Count" DESC)

WHERE ROWNUM <= 5;

*Результат виконання*:



* 1. Визначити працівника, що продає найбільше товарів.

*Текст запиту*:

SELECT \* FROM

(SELECT

st.staffId,

st.name,

st.surname,

p.name position,

SUM(sl.quantity) Quantity

FROM staff st

JOIN sales sa

ON st.staffId = sa.staffId

JOIN saleList sl

ON sa.saleId = sl.saleId

JOIN positions p

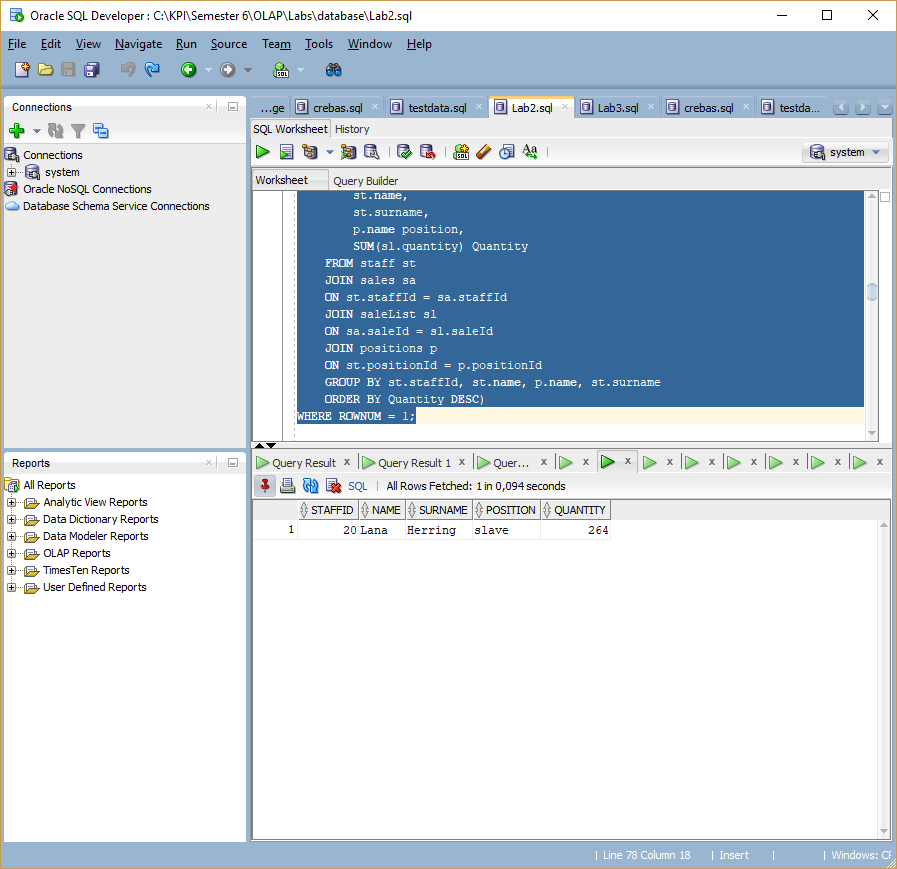
ON st.positionId = p.positionId

GROUP BY st.staffId, st.name, p.name, st.surname

ORDER BY Quantity DESC)

WHERE ROWNUM = 1;

*Результат виконання*:



* 1. Визначити 3 працівника, що оформили найбільше чеків.

*Текст запиту*:

SELECT \* FROM

(SELECT

st.staffId,

st.name,

st.surname,

p.name position,

COUNT(sa.saleId) SaleCount

FROM staff st

JOIN sales sa

ON st.staffId = sa.staffId

JOIN positions p

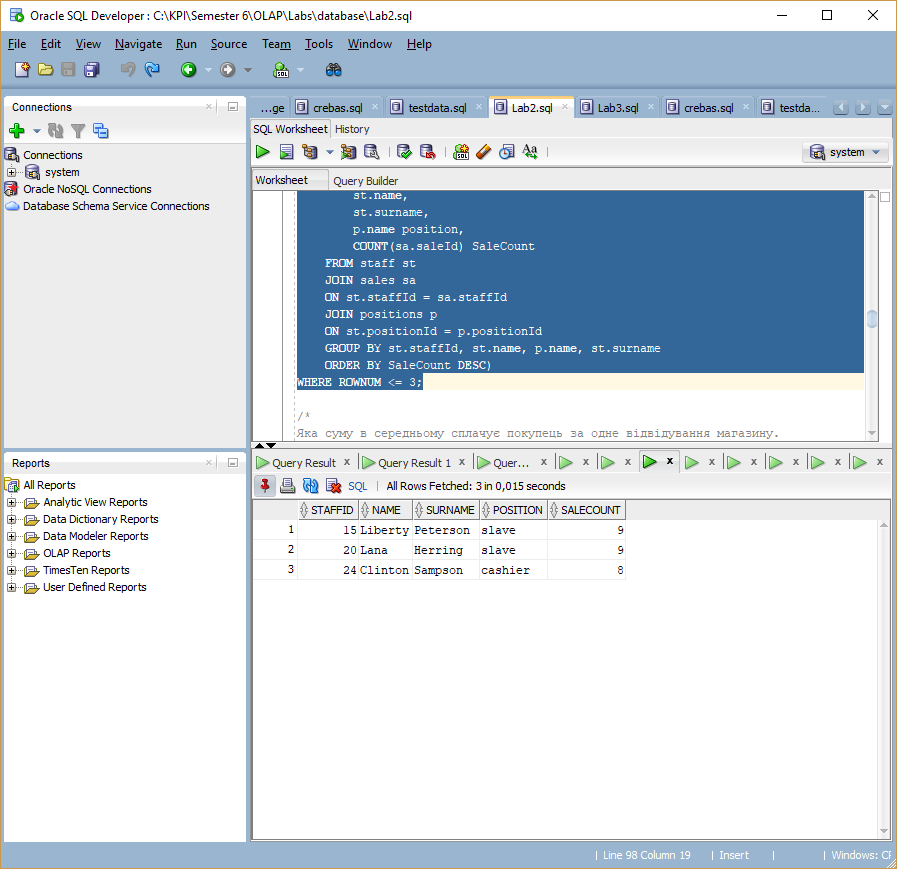
ON st.positionId = p.positionId

GROUP BY st.staffId, st.name, p.name, st.surname

ORDER BY SaleCount DESC)

WHERE ROWNUM <= 3;

*Результат виконання*:



* 1. Яка суму в середньому сплачує покупець за одне відвідування магазину.

*Текст запиту*:

SELECT AVG(AllPrice)

FROM

(SELECT

sa.saleId,

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

FROM sales sa

JOIN saleList sl

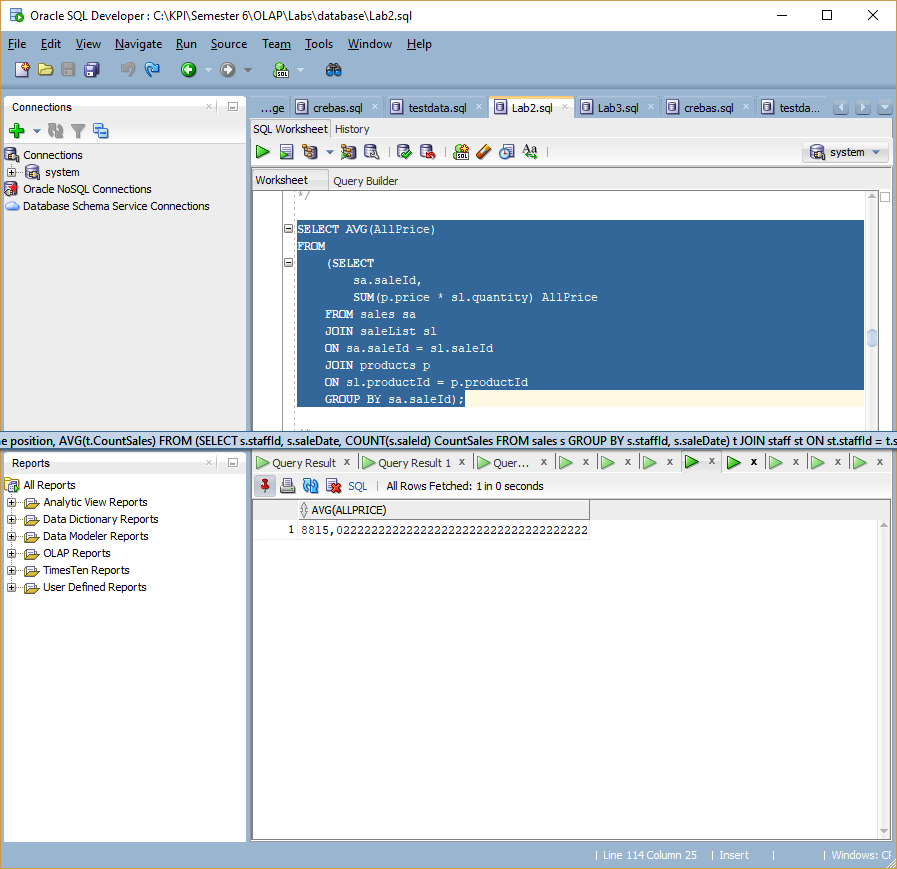
ON sa.saleId = sl.saleId

JOIN products p

ON sl.productId = p.productId

GROUP BY sa.saleId);

*Результат виконання*:



* 1. Скільки покупок оформлює кожен продавець за день.

*Текст запиту*:

SELECT

t.staffId,

st.name,

st.surname,

p.name position,

AVG(t.CountSales)

FROM

(SELECT

s.staffId,

s.saleDate,

COUNT(s.saleId) CountSales

FROM sales s

GROUP BY s.staffId, s.saleDate) t

JOIN staff st

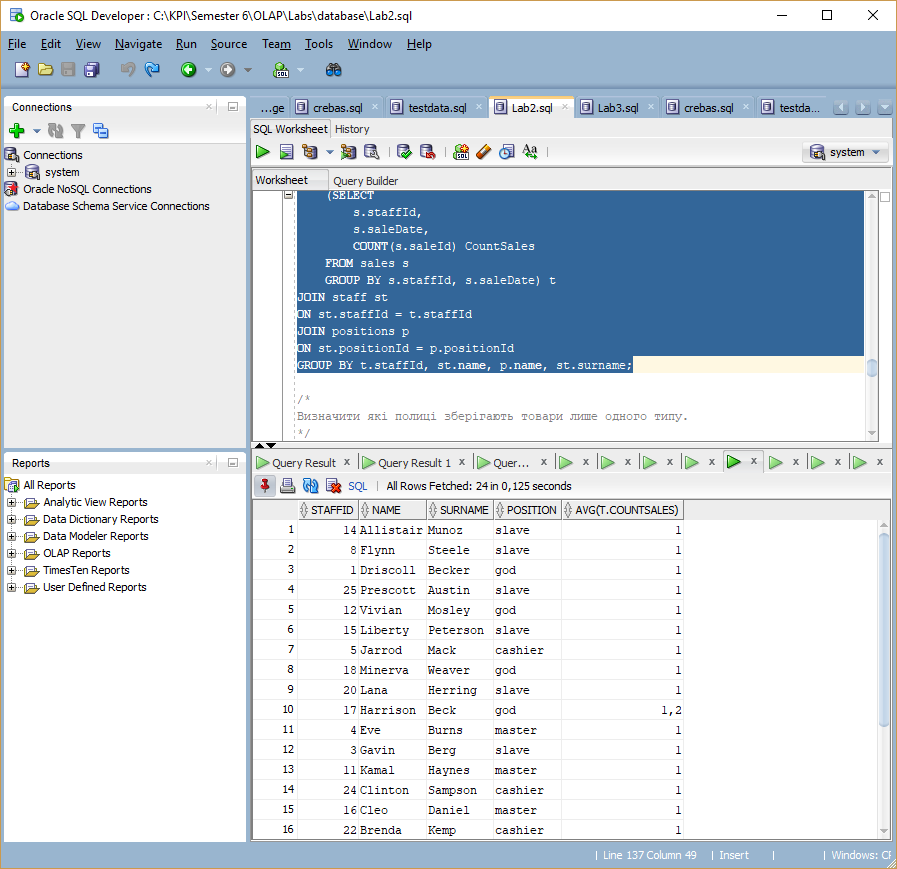
ON st.staffId = t.staffId

JOIN positions p

ON st.positionId = p.positionId

GROUP BY t.staffId, st.name, p.name, st.surname;

*Результат*:



* 1. Визначити які полиці зберігають товари лише одного типу.

*Текст запиту*:

SELECT \* FROM

(SELECT

st.shelf,

COUNT(DISTINCT p.typeId) CountType

FROM store st

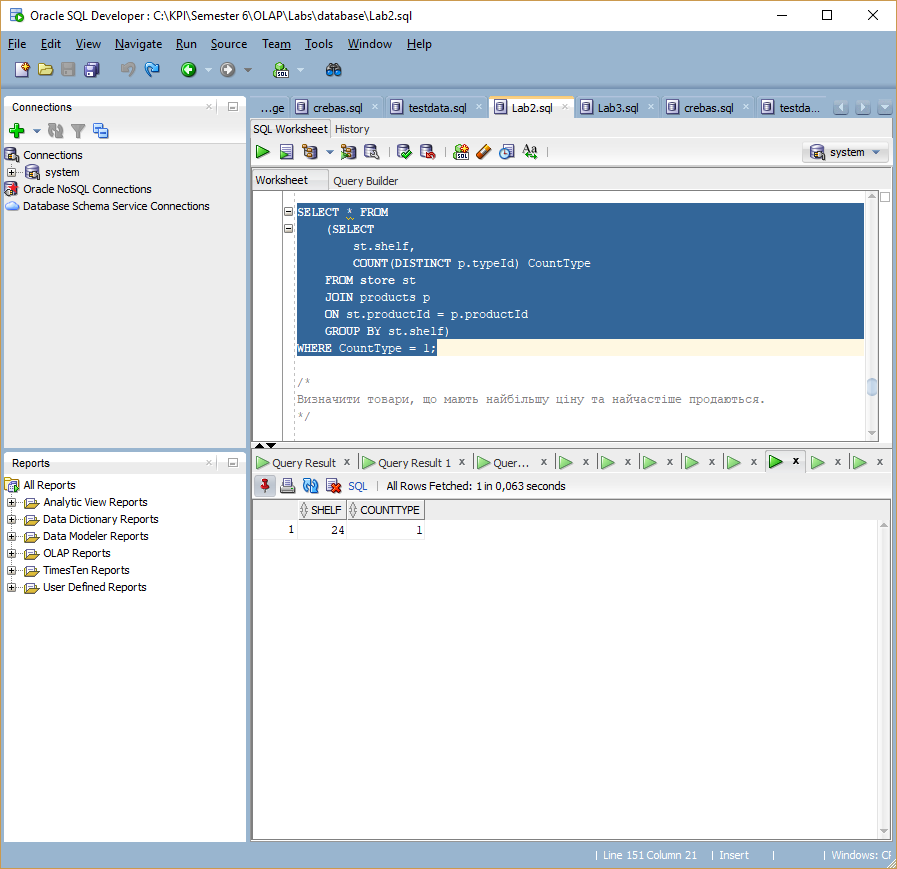
JOIN products p

ON st.productId = p.productId

GROUP BY st.shelf)

WHERE CountType = 1;

*Результат*:



* 1. Визначити товари, що мають найбільшу ціну та найчастіше продаються.

*Текст запиту*:

SELECT \* FROM

(SELECT

sl.productId,

SUM(sl.quantity) ProductQuantity

FROM saleList sl

GROUP BY sl.productId

ORDER BY ProductQuantity DESC) tq

JOIN

(SELECT

p.productId,

p.price

FROM products p

ORDER BY p.price DESC) tp

ON tq.productId = tp.productId;

*Результат*:

