Ryerson University

Lab Report Cover Page

Course Number	CPS510
Course Title	Database Systems I
Semester/Year	F2020
Instructor	Dr. A. Abhari
TA Name	Haytham Qushtom

Assignment/Lab Report No.	5
Report Title	Demonstration of adv. queries by Unix shell Implementation

Section No.	8
Group No.	9
Submission Date	October 27, 2020
Due Date	October 27, 2020

Student Name	Student ID	Initial
Tanvir Billah	500829695	T.B.
James Choi	500931983	J.C.
Mahamudul Islam	500963051	M.I.

Dropping tables:

```
================================================================
| Oracle All Inclusive Tool|
| Main Menu - Select Desired Operation(s):|
| <CTRL-Z Anytime to Enter Interactive CMD Prompt>|
----------------------------------------------------------------
  1) Drop Tables
  2) Create Tables
  3) Populate Tables
  4) Query Tables

  E) End/Exit
Choose:
1

SQL*Plus: Release 12.1.0.2.0 Production on Mon Oct 26 15:25:16 2020

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.
```

Creating tables:

```
==================================================================
| Oracle All Inclusive Tool|
| Main Menu - Select Desired Operation(s):|
| <CTRL-Z Anytime to Enter Interactive CMD Prompt>|
------------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Query Tables

 E) End/Exit
Choose:
2

SQL*Plus: Release 12.1.0.2.0 Production on Mon Oct 26 15:28:49 2020

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4    5    6
Table created.

SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4
Table created.

SQL> SQL>   2    3    4
Table created.

SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4    5    6    7    8    9
Table created.

SQL> SQL>   2    3    4    5    6    7    8    9
Table created.

SQL> SQL>   2    3    4    5    6
Table created.

SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0
```

Populating tables:

```
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

Queries:

/*advanced queries for assignment 5 */
--Advanced join query
SELECT DISTINCT product_name, brand_name, category_name, seller_name
FROM product, product_category, brand, seller
WHERE product.brand_id = brand.brand_id
AND product.category_id = product_category.category_id
AND product.seller_id = seller.seller_id
ORDER BY product_name;

--Customers who bought Vintage 80s OR bought coffee table
SELECT customer_name
FROM customer, product, contains_product, cart, payment
WHERE contains_product.product_id = product.product_id
AND contains_product.cart_id = cart.cart_id
AND cart.payment_id = payment.payment_id
AND payment.customer_id = customer.customer_id
AND product_name = 'Vintage 80s Military Jacket M'
UNION
(SELECT customer_name
FROM customer, product, contains_product, cart, payment
WHERE contains_product.product_id = product.product_id
AND contains_product.cart_id = cart.cart_id
AND cart.payment_id = payment.payment_id
AND payment.customer_id = customer.customer_id
AND product_name = 'Coffee Table Book');

--Aggregate function, # of items sold
SELECT product_name, COUNT(*)
FROM contains_product, product
WHERE contains_product.product_id = product.product_id
GROUP BY product_name;

--Products that cost above $20 and $70 or under
SELECT product_name
FROM product
WHERE price BETWEEN 20 AND 70;

--Customers who have their products shipped by couriers other than Canpost
SELECT customer_name, cart_status
FROM customer, cart, payment
WHERE cart.payment_id = payment.payment_id
AND payment.customer_id = customer.customer_id

```sql
AND cart_status = 'Shipped'
AND NOT EXISTS
(SELECT *
FROM courier
WHERE courier_id = 1
AND cart.courier_id = courier.courier_id);
--Products sold by ToysRUs
SELECT product_name
FROM product
WHERE NOT EXISTS
(SELECT *
FROM seller
WHERE seller_id <> 4
AND product.seller_id = seller.seller_id);
```

```
----------
Microsoft Keyboard
         1

Hot Wheels Track Set
         1

Zelda
         1


PRODUCT_NAME
--------------------------------------------------------------------------------
  COUNT(*)
----------
Coffee Table Book
         1

Apple Monitor
         1

Vintage 80s Military Jacket M
         1


6 rows selected.

SQL> SQL> SQL>    2    3
PRODUCT_NAME
--------------------------------------------------------------------------------
Hot Wheels Track Set
Zelda
Vintage 80s Military Jacket M

SQL> SQL> SQL>    2    3    4    5    6    7    8    9   10
CUSTOMER_NAME
--------------------------------------------------------------------------------
CART_STATUS
---------------
Mr Saad
Shipped


SQL> SQL> SQL>    2    3    4    5    6    7
PRODUCT_NAME
--------------------------------------------------------------------------------
Hot Wheels Track Set
Zelda

SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```