UTM

UNIVERSITI TEKNOLOGI MALAYSIA

SECD2523 DATABASE

SEMESTER 1 SESSION 2023/2024

LAB 3

DATA MANIPULATION LANGUANGE 2 (DML2)

Name	:	Adam Imran Bin Alwi
Matric Number	:	B23CS0017
Section	:	08
Lecturer Name	:	Dr. Noor Hidayah Zakaria

Database Design Project

Oracle Baseball League Store Database

Project Scenario:

You are a small consulting company specializing in database development. You have just been awarded the contract to develop a data model for a database application system for a small retail store called Oracle Baseball League (OBL).

The Oracle Baseball League store serves the entire surrounding community selling baseball kit. The OBL has two types of customer, there are individuals who purchase items like balls, cleats, gloves, shirts, screen printed t-shirts, and shorts. Additionally customers can represent a team when they purchase uniforms and equipment on behalf of the team.

Teams and individual customers are free to purchase any item from the inventory list, but teams get a discount on the list price depending on the number of players. When a customer places an order we record the order items for that order in our database.

OBL has a team of three sales representatives that officially only call on teams but have been known to handle individual customer complaints.

Section 6 Lesson 6 Exercise 1: Retrieving Data Using SELECT

Write and Execute SELECT statements (S6L6 Objective 2)

In this exercise you will retrieve data that is stored in the database system by using a SELECT statement.

Part 1: Retrieving all columns from a table.

Using the SELECT * statement show all data stored in the following tables:

1. customers.

```
1  SELECT * FROM customers
```

Results | Explain | Describe | Saved SQL | History

CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER
c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	150	sr01	t001	-
c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	0	-	-	lc1015
c00101	unknown@here.com	John	Doe	03216547808	987.5	sr01	t002	-
c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	85	-	-	lc2341
c01986	margal87@delphiview.com	Maria	Galant	01442736589	125.65	sr03	t003	-
c02001	brianrog@hootech.com	Brian	Rogers	01654564898	50	-	-	lc4587

2. teams.

```
1  SELECT * FROM teams
```

Results | Explain | Describe | Saved SQL | History

ID	NAME	NUMBER_OF_PLAYERS	DISCOUNT
t001	Rockets	25	10
t002	Celtics	42	20
t003	Rovers	8	-
t004	Jets	10	5

3. items

```
1  SELECT * FROM items
```

Results | Explain | Describe | Saved SQL | History

ITM_NUMBER	NAME	DESCRIPTION	CATEGORY	COLOR	Size	ILT_ID
im01101044	gloves	catcher mitt	clothing	brown	m	il010230124
im01101045	under shirt	top worn under the game top	clothing	white	s	il010230125
im01101046	socks	team socks with emblem	clothing	range	l	il010230126
im01101047	game top	team shirt with emblem	clothing	range	m	il010230127
im01101048	premium bat	high quaity basball bat	equipment	-	-	il010230128

5 rows returned in 0.01 seconds Download

Part 2: Selecting Specific Columns

1. Display the customer number, first name, last name, email and phone number of the customers.

```
1  SELECT ctr_number, first_name, last_name, email, phone_number FROM customers
```

Results | Explain | Describe | Saved SQL | History

CTR_NUMBER	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER
c00001	Robert	Thornberry	bob.thornberry@heatmail.com	01234567898
c00012	Jennifer	Jones	Jjones@freemail.com	01505214598
c00101	John	Doe	unknown@here.com	03216547808
c00103	Andrew	Murcia	MurciaA@globaltech.com	07715246890
c01986	Maria	Galant	margal87@delphiview.com	01442736589
c02001	Brian	Rogers	brianrog@hootech.com	01654564898

6 rows returned in 0.01 seconds Download

2. Display the name and number of players for each team.

```
1  SELECT name, number_of_players FROM teams
```

Results | Explain | Describe | Saved SQL | History

NAME	NUMBER_OF_PLAYERS
Rockets	25
Celtics	42
Rovers	8
Jets	10

4 rows returned in 0.01 seconds Download

3. Display the name, description and category for every item in the table.

```
1  SELECT name, description, category FROM items
```

Results | Explain | Describe | Saved SQL | History

NAME	DESCRIPTION	CATEGORY
gloves	catcher mitt	clothing
under shirt	top worn under the game top	clothing
socks	team socks with emblem	clothing
game top	team shirt with emblem	clothing
premium bat	high quaity basball bat	equipment

5 rows returned in 0.00 seconds Download

Section 6 Lesson 6 Exercise 2: Retrieving Data Using SELECT

Write and Execute SELECT statements (S6L6 Objective 2)

In this exercise you will retrieve data that is stored in the database system by using a SELECT statement.

Part 1: Using Arithmetic Operators

1. Every customer has been told they can pay off their current balance over a 12 month period. Display the customer's first name, last name, current balance and monthly payment.

```
SELECT first_name, last_name, current_balance, current_balance / 12 AS monthly_payment
    FROM customers;
```

Results | Explain | Describe | Saved SQL | History

FIRST_NAME	LAST_NAME	CURRENT_BALANCE	MONTHLY_PAYMENT
Robert	Thornberry	150	12.5
Jennifer	Jones	0	0
John	Doe	987.5	82.291666666666666666666666666666666667
Andrew	Murcia	85	7.083333333333333333333333333333333333
Maria	Galant	125.65	10.470833333333333333333333333333333333
Brian	Rogers	50	4.166666666666666666666666666666666667

6 rows returned in 0.01 seconds Download

2. Obl is considering giving a gift card to all its customers of 5.00 that can be used to reduce their current balance. Write a query that will show the customers first name, last name, customer number, current balance and the value of their balance minus the gift value.

```
SELECT first_name, last_name, ctr_number, current_balance, (current_balance - 5.00) AS discount_balance
    FROM customers;
```

Results | Explain | Describe | Saved SQL | History

FIRST_NAME	LAST_NAME	CTR_NUMBER	CURRENT_BALANCE	DISCOUNT_BALANCE
Robert	Thornberry	c00001	150	145
Jennifer	Jones	c00012	0	-5
John	Doe	c00101	987.5	982.5
Andrew	Murcia	c00103	85	80
Maria	Galant	c01986	125.65	120.65
Brian	Rogers	c02001	50	45

6 rows returned in 0.01 seconds Download

3. What would be the problem with implementing this scheme?

   The customer can have a current value that is less than 0, so the check constraint became useless.

Part 2 : Using Column Aliases

1. You previously wrote a query that display the customer's first name, last name, current balance and monthly payment. Rewrite the query to use First Name, Last Name, Balance and Monthly Repayments as the column aliases. The aliases are to be shown exactly as described (case sensitive).

```
SELECT first_name as "First Name", last_name as "Last Name",  current_balance as "Balance", current_balance / 12 AS "Monthly Repayment"
    FROM customers;
```

Results | Explain | Describe | Saved SQL | History

First Name	Last Name	Balance	Monthly Repayment
Robert	Thornberry	150	12.5
Jennifer	Jones	0	0
John	Doe	987.5	82.291666666666666666666666666666666667
Andrew	Murcia	85	7.083333333333333333333333333333333333
Maria	Galant	125.65	10.470833333333333333333333333333333333
Brian	Rogers	50	4.166666666666666666666666666666666667

6 rows returned in 0.01 seconds Download

Part 3: Using Literal Character Strings

1. Write a query that will display the team information in the following format:

   The Rockets team has 25 players and receives a discount of 10 percent.

   Use **Team Information** as the column alias.

```
SELECT
    'The ' || name || ' team has ' || number_of_players || ' players and receives a discount of ' || discount || ' percent.' AS "Team Information"
FROM teams;
```

Results | Explain | Describe | Saved SQL | History

Team Information
The Rockets team has 25 players and receives a discount of 10 percent.
The Celtics team has 42 players and receives a discount of 20 percent.
The Rovers team has 8 players and receives a discount of percent.
The Jets team has 10 players and receives a discount of 5 percent.

4 rows returned in 0.03 seconds Download

2. Why does the last team not show a discount?

   Because the Rovers team discount is null value

Section 6 Lesson 7 Exercise 1: Restricting Data Using WHERE

Limit rows using WHERE (S6L7 Objective 1)

In this exercise you will refine the data that is returned in your query by adding a WHERE clause to your SELECT statement.

Part 1: Using the WHERE Clause.

1. Using the unique customer number in the where clause display all columns for Maria Galant.

```
1  SELECT * FROM customers WHERE ctr_number = 'c01986'
```

Results | Explain | Describe | Saved SQL | History

CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER
c01986	margal87@delphiview.com	Maria	Galant	01442736589	125.65	sr03	t003	-

1 rows returned in 0.01 seconds Download

2. Display the first name, last name and customer number for all customers who have a current balance of greater than 100. Use an appropriate alias for your column headings.

```
1  SELECT first_name, last_name, ctr_number FROM customers WHERE current_balance > 100
```

Results | Explain | Describe | Saved SQL | History

FIRST_NAME	LAST_NAME	CTR_NUMBER
Robert	Thornberry	c00001
John	Doe	c00101
Maria	Galant	c01986

3 rows returned in 0.00 seconds Download

3. Display the order id, date and time of all orders that were placed before the 28th of May 2019. Use an appropriate alias for your column headings.

```
1  SELECT id AS "Order ID", odr_date AS "Order Date", odr_time AS "Order Time"
2      FROM orders
3  WHERE odr_date < TO_DATE('2019-05-28', 'YYYY-MM-DD');
```

Results | Explain | Describe | Saved SQL | History

Order ID	Order Date	Order Time
or0101250	04/17/2017	04/17/2017
or0101350	05/24/2017	05/24/2017
or0101425	05/28/2017	05/28/2017
or0101681	06/02/2017	06/02/2017
or0101750	06/18/2017	06/18/2017

5 rows returned in 0.02 seconds Download

Part 2: Range Conditions: BETWEEN Operator

1. Display the inventory id, cost and number of units using appropriate aliases for all items that have a trade cost of between 3.00 and 15.00.

```
SELECT id AS "Inventory ID", cost AS "Item Cost", units AS "Number of Units"
    FROM inventory_list
WHERE cost BETWEEN 3.00 AND 15.00;
```

Results | Explain | Describe | Saved SQL | History

Inventory ID	Item Cost	Number of Units
il010230125	7.99	250
il010230126	5.24	87

2 rows returned in 0.03 seconds Download

Part 3: Membership Conditions: IN Operator

1. Display the inventory id, cost and number of units using appropriate aliases for all items that have 50, 100, 150 or 200 units in stock.

```
SELECT id AS "Inventory ID", cost AS "Item Cost", units AS "Number of Units"
    FROM inventory_list
WHERE units IN (50, 100, 150, 200);
```

Results | Explain | Describe | Saved SQL | History

Inventory ID	Item Cost	Number of Units
il010230124	2.5	100

1 rows returned in 0.03 seconds Download

Part 4: Membership Conditions: NOT IN Operator

1. Display the inventory id, cost and number of units using appropriate aliases for all items that do not have 50, 100, 150 or 200 units in stock.

```
SELECT id AS "Inventory ID", cost AS "Item Cost", units AS "Number of Units"
    FROM inventory_list
WHERE units NOT IN (50, 100, 150, 200);
```

Results | Explain | Describe | Saved SQL | History

Inventory ID	Item Cost	Number of Units
il010230125	7.99	250
il010230126	5.24	87
il010230127	18.95	65
il010230128	97.46	8

4 rows returned in 0.00 seconds Download

Part 5: Pattern Matching: LIKE Operator

1. Display item number and name of all items that have a name that begins with g. Use an appropriate alias for your column headings.

```
SELECT itm_number AS "Item Number", name AS "Item Name"
    FROM items
WHERE name LIKE 'g%';
```

Results | Explain | Describe | Saved SQL | History

Item Number	Item Name
im01101044	gloves
im01101047	game top

2 rows returned in 0.02 seconds Download

Part 6 : Pattern Matching: Combining Wildcard Characters with the LIKE Operator

1. Display item number and name of all items that have a name that contain a lowercase o. Use an appropriate alias for your column headings.

```
SELECT itm_number AS "Item Number", name AS "Item Name"
    FROM items
WHERE name LIKE '%o%';
```

Results | Explain | Describe | Saved SQL | History

Item Number	Item Name
im01101044	gloves
im01101046	socks
im01101047	game top

3 rows returned in 0.03 seconds Download

Section 6 Lesson 7 Exercise 2: Restricting Data Using WHERE

Limit rows using WHERE (S6L7 Objective 1)

In this exercise you will refine the data that is returned in your query by adding a WHERE clause to your SELECT statement.

Part 1: Using the NULL Conditions

1. Write a query that will display information for teams that don't receive a discount in the following format:

   The Rovers team has 25 players and does not receive a discount.

   Use **Team Information** as the column alias.

```
SELECT 'The ' || name || ' team has ' ||  number_of_players || ' players and does not receive a discount.' AS "Team Information"
    FROM  teams
WHERE discount IS NULL OR discount = 0;
```

Results | Explain | Describe | Saved SQL | History

Team Information
The Rovers team has 8 players and does not receive a discount.

2. Write a query that will display information for only teams that receive a discount in the following format:

   The Rockets team has 25 players and receives a discount of 10 percent.

   Use **Team Information** as the column alias.

```
SELECT 'The ' || name || ' team has ' ||  number_of_players || ' players and does not receive a discount.' AS "Team Information"
    FROM  teams
WHERE discount IS NOT NULL AND discount > 0;
```

Results | Explain | Describe | Saved SQL | History

Team Information
The Rockets team has 25 players and does not receive a discount.
The Celtics team has 42 players and does not receive a discount.
The Jets team has 10 players and does not receive a discount.

3 rows returned in 0.01 seconds Download

Part 2: Logical Operators: AND

1. Write a query that will display the customer number, address line 1 and postal code for customers that live in the starford area of Liverpool. Use Customer Number, Street Address and Postal Code as the column aliases.

```
SELECT ctr_number AS "Customer Number", address_line_1 AS "Street Address", zip_code AS "Postal Code"
    FROM customers_addresses
WHERE address_line_2 = 'Starford' AND city = 'Liverpool';
```

Results | Explain | Describe | Saved SQL | History

Customer Number	Street Address	Postal Code
c00001	17 Gartsquare Road	LP89JHK

1 rows returned in 0.02 seconds Download

Part 3: Logical Operators: OR

1. Write a query that will display the customer number, address line 1 and postal code for customers that live in either starford or Liverpool in general. Use Customer Number, Street Address and Postal Code as the column aliases.

```
SELECT ctr_number AS "Customer Number", address_line_1 AS "Street Address", zip_code AS "Postal Code"
    FROM customers_addresses
WHERE address_line_2 = 'Starford' OR city = 'Liverpool';
```

Results | Explain | Describe | Saved SQL | History

Customer Number	Street Address	Postal Code
c00001	17 Gartsquare Road	LP89JHK
c00001	63 Acacia Drive	LP83JHR

2 rows returned in 0.01 seconds Download

Part 4: Logical Operators: NOT Equal To

1. Write a query that will display the customer number, address line 1 and postal code for customers that do not live in Liverpool. Use Customer Number, Street Address and Postal Code as the column aliases.

```
SELECT ctr_number AS "Customer Number", address_line_1 AS "Street Address", zip_code AS "Postal Code"
    FROM customers_addresses
WHERE city != 'Liverpool';
```

Results | Explain | Describe | Saved SQL | History

Customer Number	Street Address	Postal Code
c00101	54 Ropehill Crescent	ST45AGV
c01986	36 Watercress Lane	JP23YTH

2 rows returned in 0.01 seconds Download

Section 6 Lesson 8 Exercise 1: Sorting Data Using ORDER BY

Use the ORDER BY Clause to Sort SQL Results (S6L8 Objective 1)

In this exercise you will sort the order of the data that is returned in your query by adding an ORDER BY clause to the end of your SELECT statement.

1. Display the team name and number of players alphabetically in order of team name. Use an appropriate alias for your column headings.

```
1  SELECT  name AS "Team Name", number_of_players AS "Number of Players"
2      FROM teams
3  ORDER BY name
```

Results | Explain | Describe | Saved SQL | History

Team Name	Number of Players
Celtics	42
Jets	10
Rockets	25
Rovers	8

4 rows returned in 0.00 seconds Download

2. Display the team name and number of players in descending order of number of players. Use an appropriate alias for your column headings.

```
1  SELECT  name AS "Team Name", number_of_players AS "Number of Players"
2      FROM teams
3  ORDER BY number_of_players DESC;
```

Results | Explain | Describe | Saved SQL | History

Team Name	Number of Players
Celtics	42
Rockets	25
Jets	10
Rovers	8

4 rows returned in 0.01 seconds Download

3. Display the team name and number of players alphabetically in order of team name. Use Team Name for the name alias and Players for the number of players. Sort the output in descending order of name using the alias in the ORDER BY clause.

```
1  SELECT  name AS "Team Name", number_of_players AS "Number of Players"
2      FROM teams
3  ORDER BY "Team Name" DESC;
```

Results | Explain | Describe | Saved SQL | History

Team Name	Number of Players
Rovers	8
Rockets	25
Jets	10
Celtics	42

4 rows returned in 0.01 seconds Download

Section 6 Lesson 8 Exercise 2: Sorting Data Using ORDER BY

Part 1 : TOP-N-ANALYSIS (S6L8 Objective 3)

1. The customers are numbered sequentially with each new customer being assigned a higher customer number. Use TOP-N-ANALYSIS to only show the First and last name of the first three customers. Show the customers first and last name in the same column using Customer Name as the column alias.

```
SELECT first_name || ' ' || last_name AS "Customer Name"
    FROM customers
ORDER BY ctr_number
    FETCH FIRST 3 ROWS ONLY;
```

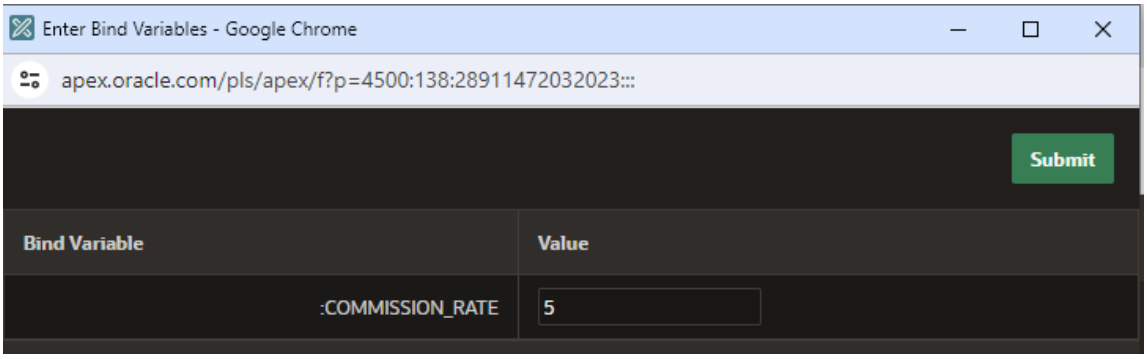
Results | Explain | Describe | Saved SQL | History

Customer Name
Robert Thornberry
Jennifer Jones
John Doe

3 rows returned in 0.01 seconds Download

Part 2 : Using a Substitution Variable (S6L8 Objective 4)

1. Use a substitution variable that will allow you to enter the commission rate for the sales representatives. The first and last names should be displayed to screen for any sales representatives that earn that commission rate and the output should be ordered by their last name. Use an appropriate alias for your column headings.




```
SELECT first_name AS "First Name", last_name AS "Last Name", commission_rate AS "Commission Rate"
    FROM sales_representatives
WHERE commission_rate = :COMMISSION_RATE
    ORDER BY last_name;
```

Results | Explain | Describe | Saved SQL | History

First Name	Last Name	Commission Rate
Barry	Speed	5
Victoria	Wright	5

2 rows returned in 0.01 seconds Download