
FACULTY OF COMPUTING
UTM Johor Bahru

**LAB 3**

**DATA MANIPULATION LANGUAGE (DML 2)**

SECD2523-Database

SEMESTER I, SESSION 2023/2024

Lecturer: Dr. Noor Hidayah

| Name | Matric No |
|---|---|
| LEE XIN HUI | A22EC0066 |

Section: 08

## DML 2 PART 1

### Part 1: Retrieving all columns from a table.

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

1. customers.

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

6 rows returned in 0.01 seconds Download

2. teams.

```
SELECT *
FROM teams;
```

Results | Explain | Describe | Saved SQL | History

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

4 rows returned in 0.02 seconds Download

3. items

```
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 |
| im01101047 | game top | team shirt with emblem | clothing | range | m | il010230127 |
| im01101048 | premium bat | high quaity basball bat | equipment | - | - | il010230128 |
| im01101046 | socks | team socks with emblem | clothing | range | l | il010230126 |

5 rows returned in 0.02 seconds Download

## Part 2: Selecting Specific Columns

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

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

6 rows returned in 0.02 seconds Download

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

```
SELECT name, number_of_players
FROM teams;
```

Results | Explain | Describe | Saved SQL | History

| NAME | NUMBER_OF_PLAYERS |
|---|---|
| Rovers | 8 |
| Jets | 10 |
| Rockets | 25 |
| Celtics | 42 |

4 rows returned in 0.00 seconds Download

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

```
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 |
| game top | team shirt with emblem | clothing |
| premium bat | high quaity basball bat | equipment |
| socks | team socks with emblem | clothing |

5 rows returned in 0.01 seconds Download

**DML 2 PART 2**

## 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
FROM customers;
```

Results | Explain | Describe | Saved SQL | History

| FIRST_NAME | LAST_NAME | CURRENT_BALANCE | CURRENT_BALANCE/12 |
|---|---|---|---|
| Andrew | Murcia | 85 | 7.083333333333333333333333333333333333 |
| Maria | Galant | 125.65 | 10.470833333333333333333333333333333333 |
| Jennifer | Jones | 0 | 0 |
| Robert | Thornberry | 150 | 12.5 |
| John | Doe | 987.5 | 82.291666666666666666666666666666666667 |
| 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
FROM customers;
```

Results | Explain | Describe | Saved SQL | History

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

6 rows returned in 0.01 seconds Download

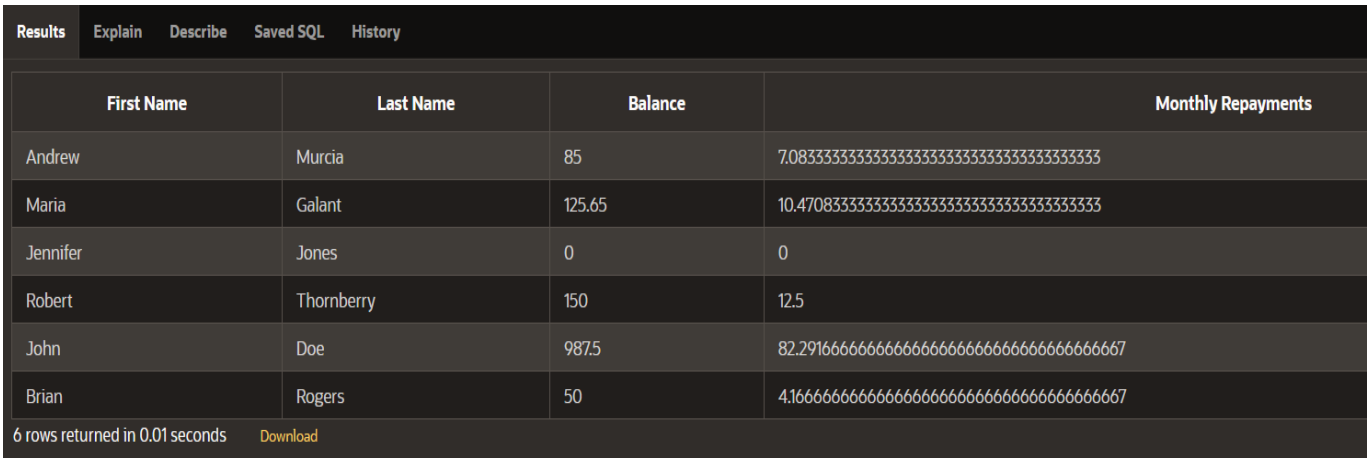
3. What would be the problem with implementing this scheme?

The current balance cannot go below to zero. If the current balance is smaller than the gift value being subtracted, the output of the calculated column will indeed be negative.

**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 "First Name" , last_name "Last Name", current_balance "Balance",
current_balance/12 "Monthly Repayments"
FROM customers;
```



Results | Explain | Describe | Saved SQL | History

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

6 rows returned in 0.01 seconds Download

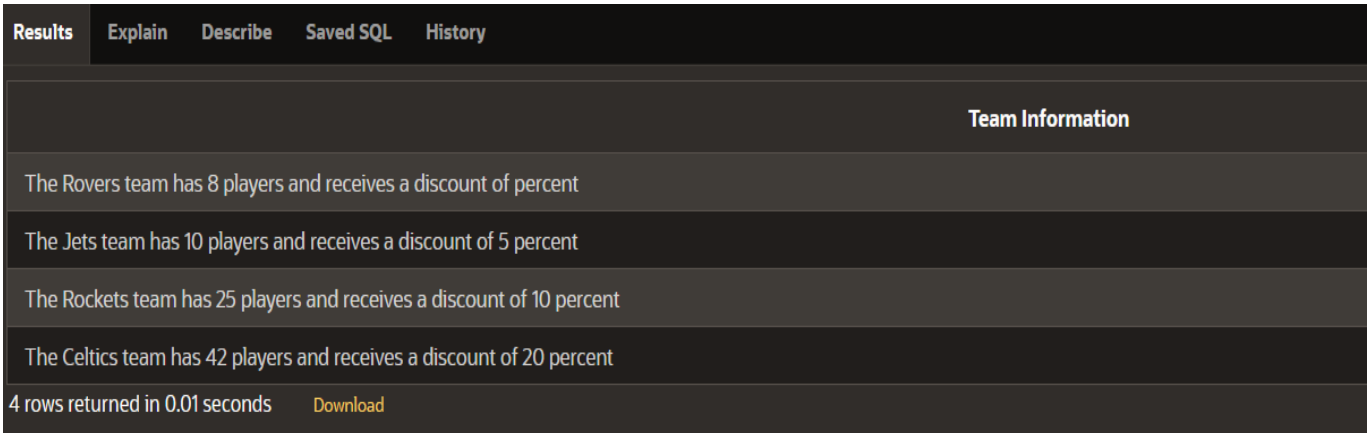
**Part 3: Using Literal Character Strings**

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;
```




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

   The discount values inserted into the teams table where name = 'Rovers' are indeed NULL, hence that specific team does not show a discount due to the absence of a defined discount value for that particular team.

**DML 2 PART 3**

**Part 1: Using the WHERE Clause.**

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

```
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 0.02 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.

```
SELECT first_name "First Name" , last_name "Last Name" , ctr_number "Customer Number"
FROM customers
WHERE current_balance > 100;
```

Results | Explain | Describe | Saved SQL | History

| First Name | Last Name | Customer Number |
|---|---|---|
| Maria | Galant | c01986 |
| Robert | Thornberry | c00001 |
| John | Doe | c00101 |

3 rows returned in 0.03 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.

```
SELECT id "Order Id", odr_date "Order Date", odr_time "Order Time"
FROM orders
WHERE odr_date < TO_DATE('28-May-2019', 'DD-MM-YYYY');
```

Results | Explain | Describe | Saved SQL | History

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

5 rows returned in 0.03 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 "Inventory Id", cost "Cost", units "Number of Units"
FROM inventory_list
WHERE cost BETWEEN 3 AND 15;
```

Results | Explain | Describe | Saved SQL | History

| Inventory Id | 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 "Inventory Id", cost "Cost", units "Number of Units"
FROM inventory_list
WHERE units IN (50, 100, 150, 200);
```

Results | Explain | Describe | Saved SQL | History

| Inventory Id | 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 "Inventory Id", cost "Cost", units "Number of Units"
FROM inventory_list
WHERE units NOT IN (50, 100, 150, 200);
```

Results | Explain | Describe | Saved SQL | History

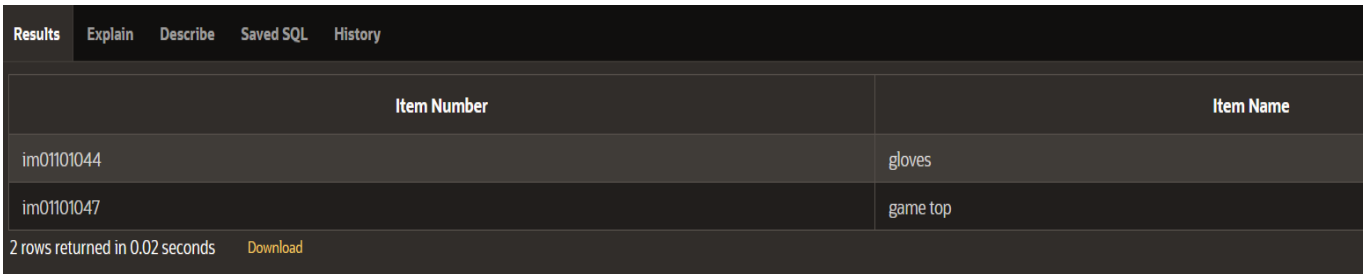
| Inventory Id | Cost | Number of Units |
|---|---|---|
| il010230128 | 97.46 | 8 |
| il010230125 | 7.99 | 250 |
| il010230126 | 5.24 | 87 |
| il010230127 | 18.95 | 65 |

4 rows returned in 0.01 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 "Item Number", name "Item Name"
FROM items
WHERE name LIKE 'g%';
```



Results | Explain | Describe | Saved SQL | History

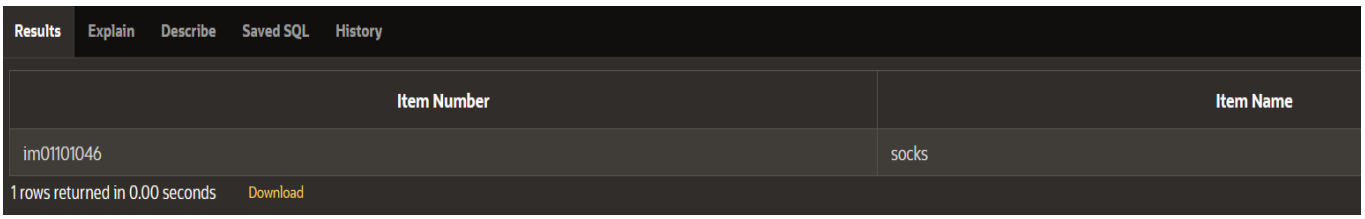
| Item Number | Item Name |
|---|---|
| im01101044 | gloves |
| im01101047 | game top |

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 "Item Number", name "Item Name"
FROM items
WHERE name LIKE '_o%';
```



Results | Explain | Describe | Saved SQL | History

| Item Number | Item Name |
|---|---|
| im01101046 | socks |

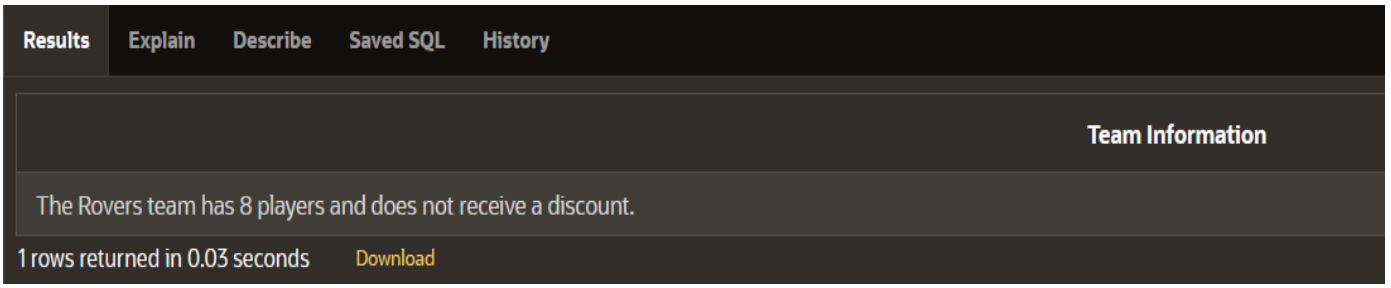
1 rows returned in 0.00 seconds Download

**DML 2 PART 4**

**Part 1: Using the NULL Conditions**

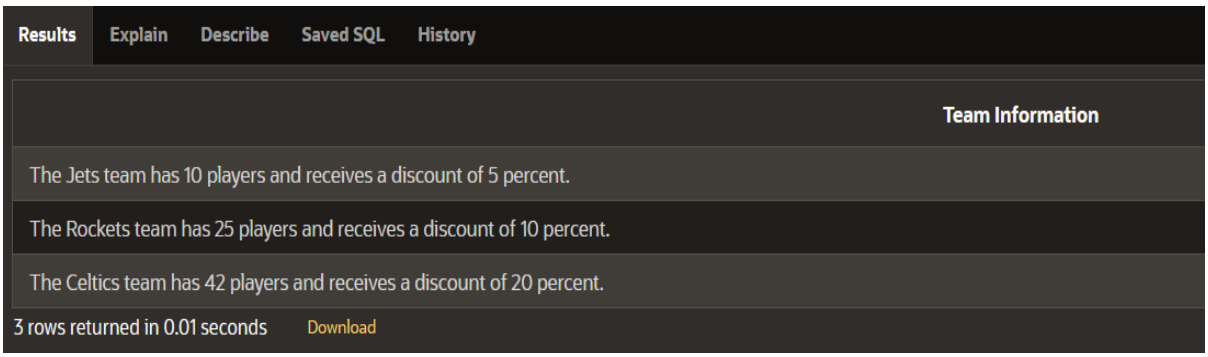
1. Write a query that will display information for teams that don't receive a discount in the following format: The Rovers team has 25 players and does not receive a discount. Use **Team Information** as the column alias.

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




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 receives a
discount of  ' | | discount | | ' percent.' AS "Team Information"
FROM teams
WHERE discount IS NOT NULL;
```



**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 "Customer Number", Address_line_1 "Street Address", zip_code "Postal Code"
FROM customers_addresses
WHERE city = 'Liverpool' AND address_line_2 = 'Starford';
```

Results | Explain | Describe | Saved SQL | History

| Customer Number | Street Address | Postal Code |
|---|---|---|
| c00001 | 17 Gartsquare Road | LP89JHK |

1 rows returned in 0.03 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 "Customer Number", Address_line_1 "Street Address",
zip_code "Postal Code"
FROM customers_addresses
WHERE city = 'Liverpool' OR address_line_2 = 'Starford';
```

Results | Explain | Describe | Saved SQL | History

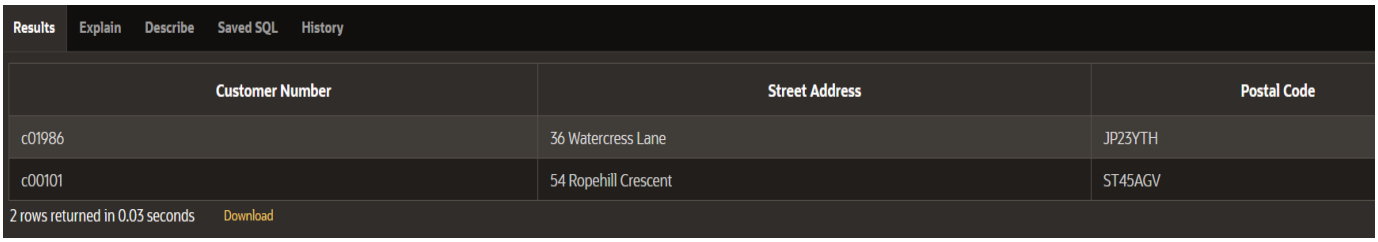
| Customer Number | Street Address | Postal Code |
|---|---|---|
| c00001 | 17 Gartsquare Road | LP89JHK |
| c00001 | 63 Acacia Drive | LP83JHR |

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 "Customer Number", Address_line_1 "Street Address", zip_code "Postal Code"
FROM customers_addresses
WHERE city NOT IN ( 'Liverpool' );
```



Results | Explain | Describe | Saved SQL | History

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

2 rows returned in 0.03 seconds Download

**DML 2 PART 5**

**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.

```
SELECT name "Team Name", number_of_players "Number of Players"
FROM teams
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 0.01 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.

```
SELECT name "Team Name", number_of_players "Number of Players"
FROM teams
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.

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

Results | Explain | Describe | Saved SQL | History

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

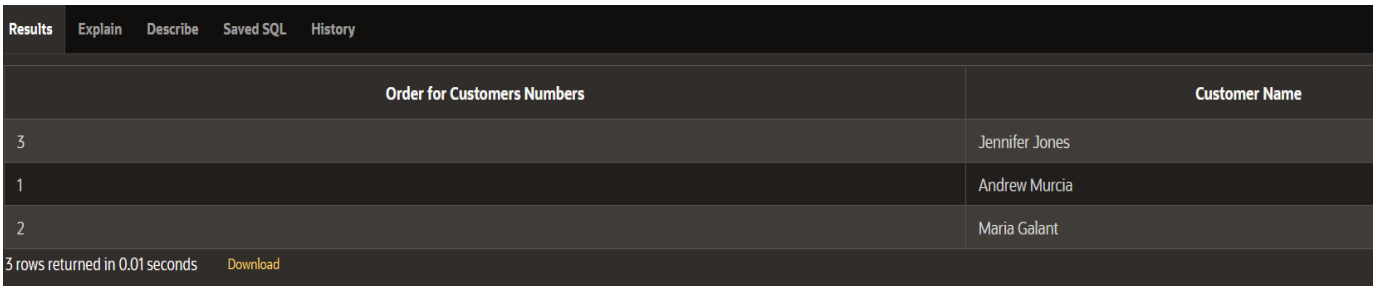
4 rows returned in 0.01 seconds Download

**DML 2 PART 6**

## 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 ROWNUM AS "Order for Customers Numbers", first_name | | ' ' | | last_name "Customer Name"
FROM customers
WHERE ROWNUM<=3
ORDER BY ctr_number;
```



Results | Explain | Describe | Saved SQL | History

| Order for Customers Numbers | Customer Name |
|---|---|
| 3 | Jennifer Jones |
| 1 | Andrew Murcia |
| 2 | Maria Galant |

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 | | ' ' | | last_name "Sales Representative Name"
FROM sales_representatives
WHERE commission_rate = :commission_rate
ORDER BY last_name
```

Submit

| Bind Variable | Value |
|---|---|
| :COMMISSION_RATE | 5 |

Results | Explain | Describe | Saved SQL | History

| Sales Representative Name |
| --- |
| Barry Speed |
| Victoria Wright |

2 rows returned in 0.06 seconds Download