|  |  |
| --- | --- |
| **Practicum Case** |  |
| ISYS6084 | J0974  Database |
| **Information Systems** | **E1-J0974-AM01** |
| ***Valid on*** *Even Semester Year 2019/2020* | **Revision 00** |

**Learning Outcomes**

* Construct query of SQL that suitable with the problem

## Topic

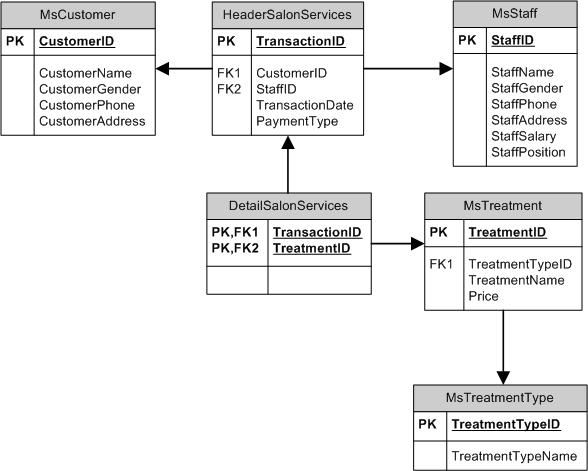
* Session 07 - SQL: Data Manipulation Language (4)

## Sub Topics

* Aggregate Functions
* Group By
* Order By
* Having

**Tabel Relasional**

*Relational Table*



**Sintaks**

*Syntax*

[**Select**](http://www.1keydata.com/sql/sqlselect.html)  
SELECT { \* | field\_name [, …] }  
FROM table\_name [, …]

[**Distinct**](http://www.1keydata.com/sql/sqldistinct.html)  
SELECT DISTINCT { \* | field\_name [, …] }  
FROM table\_name [, …]

[**Where**](http://www.1keydata.com/sql/sqlwhere.html)  
SELECT { \* | field\_name [, …] }  
FROM table\_name [, …]WHERE {condition}

[**Between**](http://www.1keydata.com/sql/sqlbetween.html)  
SELECT { \* | field\_name [, …] }  
FROM table\_name [, …]WHERE field\_name BETWEEN value1 AND value2

[**Like**](http://www.1keydata.com/sql/sqllike.html)  
SELECT { \* | field\_name [, …] }  
FROM table\_name [, …]WHERE field\_name LIKE {PATTERN}

**Aggregate**

These are the aggregate functions:

1. **Sum**(field\_name) = to sum the total content of the field.

2. **Count**(field\_name) = to count the total of rows from the data.

3. **Avg**(field\_name)= to count the average from content of the rows.

4. **Max**(field\_name)= to count the maximum value from content of the rows.

5. **Min**(field\_name)= to count the minimum value from content of the rows.

Additional syntax:

1. **Order by** = to sort the data. The default format is ascending.

2. **Group by** = to group the data that not use the aggregate function.

3. **Having** = to make a condition for aggregate function that we used.

The order of using the syntax:

* + group by
  + having (to use having, must use group by)
  + order by

## Soal

*Case*

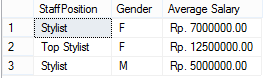
1. Display Maximum Price (obtained from the maximum price of all treatment), Minimum Price (obtained from minimum price of all treatment), and Average Price (obtained by rounding the average value of Price in 2 decimal format).

(**max**, **min**, **cast**, **round**, **avg**)

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\05 -\new\1.png

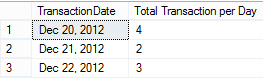
1. Display StaffPosition, Gender (obtained from first character of staff’s gender), and Average Salary (obtained by adding ‘Rp.’ in front of the average of StaffSalary in 2 decimal format).

(**left**, **cast**, **avg**, **group by**)



1. Display TransactionDate (obtained from TransactionDate in ‘Mon dd,yyyy’ format), and Total Transaction per Day (obtained from the total number of transaction).

(**convert**, **count**, **group by**)



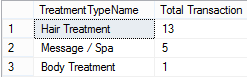
1. Display CustomerGender (obtained from customer’s gender in uppercase format), and Total Transaction (obtained from the total number of transaction).

(**upper**, **count**, **group by**)

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\05 -\new\4.png

1. Display TreatmentTypeName, and Total Transaction (obtained from the total number of transaction). Then sort the data in descending format based on the total of transaction.

(**count**, **group by**, **order by**)



1. Display Date (obtained from TransactionDate in ‘dd mon yyyy’ format), Revenue per Day (obtained by adding ‘Rp. ’ in front of the total of price) for every transaction which Revenue Per Day is between 1000000 and 5000000.

(**convert**, **cast**, **sum**, **group by**, **having**)

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\05 -\new\6.png

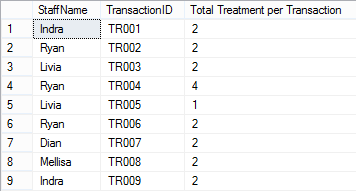
1. Display ID (obtained by replacing ‘TT0’ in TreatmentTypeID with ‘Treatment Type’), TreatmentTypeName, and Total Treatment per Type (obtained from the total number of treatment and ended with ‘ Treatment ’) for treatment type that consists of more than 5 treatments. Then sort the data in descending format based on Total Treatment per Type.

(**replace**, **cast**, **count**, **group by**, **having**, **order by**)



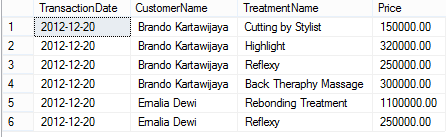
1. Display StaffName (obtained from first character of staff’s name until character before space), TransactionID, and Total Treatment per Transaction (obtained from the total number of treatment).

(**left**, **charindex**, **count**, **group by**)



1. Display TransactionDate, CustomerName, TreatmentName, and Price for every transaction which happened on ‘Thursday’ and handled by Staff whose name contains the word ‘Ryan’. Then order the data based on TransactionDate and CustomerName in ascending format.

(**datename**, **weekday**, **like**, **order by**)



1. Display TransactionDate, CustomerName, and TotalPrice (obtained from the total amount of price) for every transaction that happened after 20th day. Then order the data based on TransactionDate in ascending format.

(**sum**, **day**, **group by**, **order by**)

