|  |  |
| --- | --- |
| **Practicum Case** |  |
| M0564 | ISYS6123  Introduction to Database Systems |
| **Information Systems** | **E1-ISYS6123-AM01-10** |
| ***Valid on*** *Even Semester Year 2019/2020* | **Revision 00** |

**Learning Outcomes**

* Construct query of SQL that suitable with the problem

## Topic

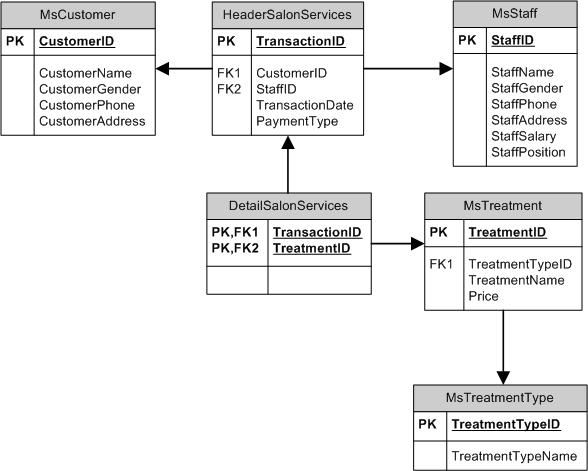
* Session 10 - SQL – Data Manipulation (Multi table Queries)

## Sub Topics

* Join tables together
* Set operations (Union, Intersect, Except)
* Outer Join

**Tabel Relasional**

*Relational Table*



**Sintaks**

*Syntax*

###### **Join**

SELECT { \* | field\_name [, …] }

FROM first\_table [INNER | LEFT | RIGHT | FULL] JOIN second\_table

ON first\_table.keyfield = second\_table.foreign\_keyfield

###### **Union**

select\_query1

{UNION | UNION ALL | INTERSECT | EXCEPT }

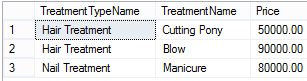
select\_query2

## Soal

*Case*

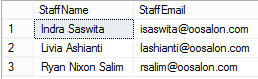
1. Display TreatmentTypeName, TreatmentName, and Price for every treatment which name contains ‘hair’ or start with ‘nail’ word and has price below 100000.

(**join**, **like**)



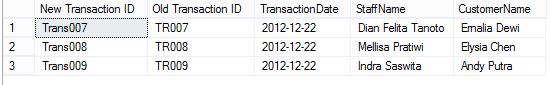
1. Display StaffName and StaffEmail (obtained from the first character of staff’s name in lowercase format and followed with last word of staff’s name and ‘@oosalon.com’ word) for every staff who handle transaction on Thursday.The duplicated data must be displayed only once.

(**distinct**, **lower**, **left**, **reverse**, **left**, **charindex**, **join**, **datename**, **weekday**, **like**)



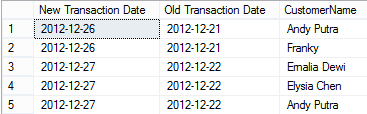
1. Display New Transaction ID (obtained by replacing ‘TR’ in TransactionID with ‘Trans’), Old Transaction ID (obtained from TransactionId), TransactionDate, StaffName, and CustomerName for every transaction which happened 2 days before 24th December 2012.

(**replace**, **join**, **datediff**, **day**)



1. Display New Transaction Date (obtained by adding 5 days to TransactionDate), Old Transaction Date (obtained from TransactionDate), and CustomerName for every transaction which didn’t happen on day 20th.

(**dateadd**, **day**, **join**, **datepart**)



1. Display Day (obtained from the day transaction happened), CustomerName, and TreatmentName for every Customer who was handled by female staff that has position name begin with ‘TOP’ word. Then order the data based on CustomerName in ascending format.

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



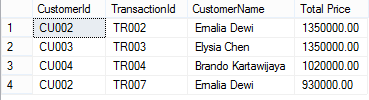
1. Display the first data of CustomerId, CustomerName, TransactionId, Total Treatment (obtained from the total number of treatment). Then sort the data based on Total Treatment in descending format.

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

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\07\new\6.png

1. Display CustomerId, TransactionId, CustomerName, and Total Price (obtained from total amount of price) for every transaction with total price is higher than the average value of treatment price from every transaction. Then sort the data based on Total Price in descending format.

(**sum**, **join**, **alias subquery**,**avg**, **group by**, **having**, **order by**)



1. Display Name (obtained by adding ‘Mr. ’ in front of StaffName), StaffPosition, and StaffSalary for every male staff. The **combine** it with Name (obtained by adding ‘Ms. ’ in front of StaffName), StaffPosition, and StaffSalary for every female staff. Then sort the data based on Name and StaffPosition in ascending format.

(**union**, **order by**)



1. Display TreatmentName, Price (obtained by adding ‘Rp. ’ in front of Price), and Status as ‘Maximum Price’ for every Treatment which price is the highest treatment’s price. Then **combine** it with TreatmentName, Price (obtained by adding ‘Rp. ’ in front of Price), and Status as ‘Minimum Price’ for every Treatment which price is the lowest treatment’s price.

(**cast**, **max**, **alias** **subquery**, **union**, **min**)

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\07\new\9.png

1. Display Longest Name of Staff and Customer (obtained from CustomerName), Length of Name (obtained from length of customer’s name), Status as ‘Customer’ for every customer who has the longest name. Then **combine** it with Longest Name of Staff and Customer (obtained from StaffName), Length of Name (obtained from length of staff’s name), Status as ‘Staff’ for every staff who has the longest name

(**len**, **max**, **alias subquery**, **union**)

C:\Users\AdityaMili\Desktop\BANK SOAL\Bank Case Making\07\new\10.png