Soal Praktikum *Practicum Case*	
ISYS6028 Database Systems	
Teknik Informatika *Computer Science*	O221-ISYS6028-DW01-09
Periode Berlaku Mulai Semester Ganjil 2021/2022 ***Valid on*** *Odd Semester Year* 2021/2022	**Revisi 00** *Revision 00*

Learning Outcomes

- Apply database language and SQL Programming language

Topic

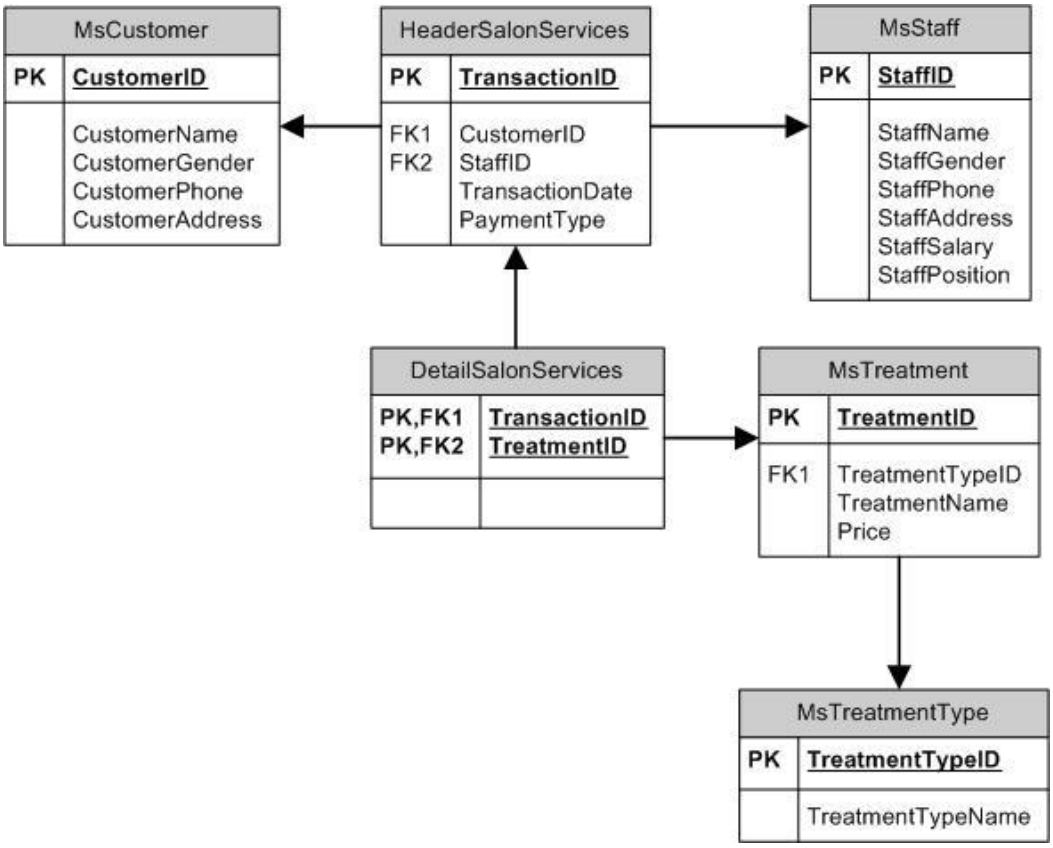
- Session 09 - Advanced SQL

Sub Topics

- Cursors
- Store Procedure
- Trigger
- Function

Tabel Relasional

Relational Table



Sintaks
Syntax

Create Store Procedure
CREATE {PROC | PROCEDURE} **procedure_name** [@parameter datatype (length) [, …]]
AS query

Alter Store Procedure
ALTER {PROC | PROCEDURE} **procedure_name** [@parameter datatype (length) [, …]]
AS query

Use '@' for variable name, e.g.: @customerName

Execute Store Procedure
[EXEC | EXECUTE] **procedure_name** [parameter [, …]]

Drop Store Procedure
DROP {PROC | PROCEDURE} **procedure_ name**

Create Trigger
CREATE TRIGGER **trigger_name** ON {table_name | view_name}
{ FOR | AFTER | INSTEAD OF } { [INSERT] [,] [UPDATE] [,] [DELETE] }
AS query

After trigger not supported for views

Drop Trigger
DROP TRIGGER **trigger_name**

Cursor
DECLARE **cursor_name** CURSOR
[LOCAL | GLOBAL]
[FORWARD_ONLY | SCROLL]
[STATIC | KEYSET | DYNAMIC | FAST_FORWARD]
[READ_ONLY | SCROLL_LOCKS | OPTIMISTIC]
[TYPE_WARNING]
FOR select_query

OPEN **cursor_name**

```
FETCH [ NEXT | PRIOR | FIRST | LAST
            | ABSOLUTE { row_number }
            | RELATIVE { row_number } ]
FROM [GLOBAL] cursor_name
WHILE @@FETCH_STATUS = 0
        BEGIN
                query
                FETCH NEXT FROM cursor_name
        END
```

CLOSE [GLOBAL] **cursor_name**

DEALLOCATE [GLOBAL] **cursor_name**

Soal
Case

1. Create a store procedure with named 'sp1' to display CustomerId, CustomerName, CustomerGender, and CustomerAddress for every Customer with Id based on user's input.
(**create procedure**)

EXEC sp1 'CU001'

	CustomerId	CustomerName	CustomerGender	CustomerAddress
1	CU001	Franky	Male	Daan mogot baru Street no 6

2. Create a store procedure with named 'sp2' that receives CustomerName as input from user with the following specification:
   - If the length of CustomerName is odd then procedure will give output '**Character Length of Mentor Name is an Odd Number'**.
   - If the length of CustomerName is even then procedure will display CustomerId, CustomerName, CustomerGender, TransactionId, and TransactionDate for every transaction with customer whose name contains the name that was inputted by user.

(**create procedure**, **len**, **like**)

EXEC sp2 'Elysia Chen'

```
Character Length of Customer Name is an Odd Number
```

EXEC sp2 'Fran'

	CustomerId	CustomerName	CustomerGender	TransactionId	TransactionDate
1	CU001	Franky	Male	TR001	2012-12-20
2	CU001	Franky	Male	TR006	2012-12-21

3. Create a store procedure named 'sp3' to update StaffId, StaffName, StaffGender, and StaffPhone on **MsStaff** table based on StaffId, StaffName, StaffGender, and StaffPhone that was inputted by user. Then display the updated data if the StaffId exists in **MsStaff** table. Otherwise show message '**Staff does not exists**'.
(**create procedure**, **update**, **exists**)

Before EXEC sp3 'SF005', 'Ryan Nixon', 'M', '08567756123'

	StaffId	StaffName	StaffGender	StaffPhone	StaffAddress	StaffSalary	StaffPosition
1	SF001	Dian Felita Tanoto	Female	085265442222	Palmerah Street no 56	15000000.00	Top Stylist
2	SF002	Mellisa Pratiwi	Female	085755552011	Kebon Jeruk Street no 151	10000000.00	Top Stylist
3	SF003	Livia Ashianti	Female	085218542222	Kebon Jeruk Street no 19	7000000.00	Stylist
4	SF004	Indra Saswita	Male	085564223311	Sunter Street no 91	7000000.00	Stylist
5	SF005	Ryan Nixon Salim	Male	085710255522	Kebon Jeruk Street no 123	3000000.00	Stylist

After EXEC sp3 'SF005', 'Ryan Nixon', 'M', '08567756123'

	StaffId	StaffName	StaffGender	StaffPhone	StaffAddress	StaffSalary	StaffPosition
1	SF005	Ryan Nixon	M	08567756123	Kebon Jeruk Street no 123	3000000.00	Stylist

```
EXEC sp3 'SF008', 'Ryan Nixon', 'M', '08567756123'
Staff does not exists
```

4. Create trigger named ‘trig1’ for **MsCustomer** table to validate if there are any data which had been updated, it will display before and after updated data on **MsCustomer** table.
(**create trigger**, **union**)

Update MsCustomer SET CustomerName = 'Franky Quo' WHERE CustomerId = 'CU001'

	CustomerId	CustomerName	CustomerGender	CustomerPhone	CustomerAddress
1	CU001	Franky	Male	08566543338	Daan mogot baru Street no 6
2	CU001	Franky Quo	Male	08566543338	Daan mogot baru Street no 6

5. Create trigger with name ‘trig2’ for **MsCustomer** table to validate if there are any new inserted data, then the first data on **MsCustomer** will be deleted.
(**create trigger**, **top**, **delete**)

Before INSERT INTO MsCustomer VALUES('CU006','Yogie soesanto', 'Male', '085562133000', 'Pelsakih Street no 52')

	CustomerId	CustomerName	CustomerGender	CustomerPhone	CustomerAddress
1	CU001	Franky	Male	08566543338	Daan mogot baru Street no 6
2	CU002	Emalia Dewi	Female	085264782135	Tanjung Duren Street no 185
3	CU003	Elysia Chen	Female	085754206611	Kebon Jeruk Street no 120
4	CU004	Brando Kartawijaya	Male	081170225561	Greenvil Street no 88
5	CU005	Andy Putra	Male	087751321421	Sunter Street no 42

After INSERT INTO MsCustomer VALUES('CU006','Yogie soesanto', 'Male', '085562133000', 'Pelsakih Street no 52')

	CustomerId	CustomerName	CustomerGender	CustomerPhone	CustomerAddress
1	CU002	Emalia Dewi	Female	085264782135	Tanjung Duren Street no 185
2	CU003	Elysia Chen	Female	085754206611	Kebon Jeruk Street no 120
3	CU004	Brando Kartawijaya	Male	081170225561	Greenvil Street no 88
4	CU005	Andy Putra	Male	087751321421	Sunter Street no 42
5	CU006	Yogie soesanto	Male	085562133000	Pelsakih Street no 52

6. Create trigger with name 'trig3' on **MsCustomer** table to validate if the data on **MsCustomer** table is deleted, then the deleted data will be insert into **Removed** table. If **Removed** table hasn't been created, then create the **Removed** table and insert the deleted data to **Removed** table.
(**create trigger**, **object_id**, **is not null**, **insert**, **select into**)

```
DELETE FROM MsCustomer WHERE CustomerId = 'CU002'
```

	CustomerId	CustomerName	CustomerGender	CustomerPhone	CustomerAddress
1	CU002	Emalia Dewi	Female	085264782135	Tanjung Duren Street no 185

7. Create cursor with name 'cur1' to validate whether the length of StaffName is odd or even then show the message about result.
(**declare cursor**, **len**)

```
The length from StaffName Dian Felita Tanoto is an odd number
The length from StaffName Mellisa Pratiwi is an even number
The length from StaffName Livia Ashianti is an odd number
The length from StaffName Indra Saswita is an even number
The length from StaffName Ryan Nixon Salim is an odd number
```

8. Create procedure named 'sp4' that receive StaffName from user's input to display StaffName and StaffPosition for every staff which name contains the word that has been inputted by user.
(**create procedure**, **declare cursor**, **like**)

```
EXEC sp4 'a'
StaffName : Dian Felita Tanoto Position : Top Stylist
StaffName : Mellisa Pratiwi Position : Top Stylist
StaffName : Livia Ashianti Position : Stylist
StaffName : Indra Saswita Position : Stylist
StaffName : Ryan Nixon Salim Position : Stylist
```

9. Create procedure with name 'sp5' that receive CustomerId from user's input to display CustomerName, and TransactionDate for every customer which Id has been inputted by user and did treatment which ID is an even number.
(**create procedure**, **declare cursor**, **in**, **right**)

```
EXEC sp5 'CU001'
Customer Name : Franky Transaction Date is 2012-12-21
```

10. Delete all procedure and trigger that has been made.

(**drop proc**, **drop trigger**)