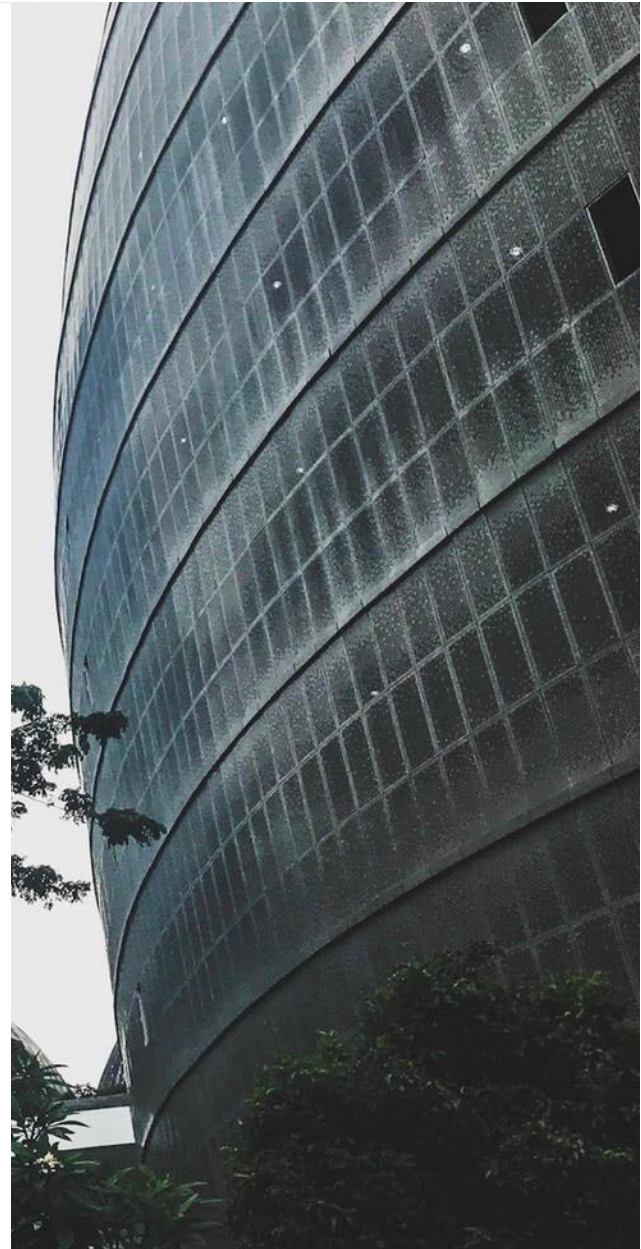


MODUL PRAKTIKUM

IS341 – DATABASE SYSTEM
PROGRAM SARJANA S1 SISTEM INFORMASI
FAKULTAS TEKNIK DAN INFORMATIKA



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN INFORMATIKA
UNIVERSITAS MULTIMEDIA NUSANTARA**

Gedung B Lantai 5, Kampus UMN
Jl. Scientia Boulevard, Gading Serpong, Tangerang, Banten-15811 Indonesia
Telp: +62-21.5422.0808 (ext. 1803), email: ict.lab@umn.ac.id, web: umn.ac.id

DAFTAR ISI

CAPAIAN PEMBELAJARAN PRAKTIKUM.....	3
MODULE 1 INTRODUCTION TO SQL.....	4
MODULE 2 DATABASE ENVIRONMENT.....	8
MODULE 3 SQL DATA DEFINITION I.....	32
MODULE 4 SQL DATA DEFINITION II.....	36
MODULE 5 SECURITY AND ADMINISTRATION	41
MODULE 6 DATA MANIPULATION I.....	47
MODULE 7 CASE STUDY 1	52
MODULE 8 DATA MANIPULATION II.....	53
MODULE 9 DATA MANIPULATION III.....	55
MODULE 10 DATA MANIPULATION IV	60
MODULE 11 DATA MANIPULATION V	66
MODULE 12 NORMALIZATION	70
MODULE 13 ER MODELING	78
MODULE 14 CASE STUDY 2	90

MODULE 1

INTRODUCTION TO SQL



DESKRIPSI TEMA

Describe database systems, terminology, environment, and new concept of database

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to introduction to SQL: Introduction To Database Introduction To SQL Server, Database (Diagram, Table, View, Stored Procedure, Users, Roles), Attach & Detach Database, Import & Export Data, Backup & Restore Database, Query Execution

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. Textbook: These module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 1

LANGKAH-LANGKAH PRAKTIKUM

1. Introduction to Database

Database atau basis data merupakan bagian sangat penting dalam sebuah proses pengolahan data. Basis data adalah koleksi data yang saling berhubungan dan memiliki arti dan terorganisasi secara rapi. Data tersebut harus dapat diakses dengan urutan yang berbeda-beda secara logis dengan cara yang relatif mudah.

Suatu sistem basis data terdiri dari empat komponen yaitu data, yang secara fisik menyimpan informasi-informasi, Database Management System (DBMS) yaitu software yang mengelola basis data, Data Definition language (DDL) dan Data Manipulation Language (DML) yaitu basis data yang berfungsi untuk mendeskripsikan data ke database management system (DBMS) dan juga member fasilitas untuk melakukan perubahan, pemeliharaan, dan pengelolaan basis data, dan program aplikasi yang memudahkan pengguna akhir untuk menggunakan data dan mendapatkannya sebagai informasi yang sesuai.

2. Introduction to SQL Server

Microsoft SQL Server adalah Relational Database Management System (RDBMS) yang dirancang untuk dapat berjalan pada platform mulai dari laptop hingga server multiprosesor besar. SQL Server biasanya digunakan sebagai sistem backend untuk website dan CRMS perusahaan dan dapat membantu ribuan pengguna secara bersamaan.

SQL Server hadir dengan sejumlah alat untuk membantu administrasi database dan tugas-tugas pemrograman. SQL Server jauh lebih kuat dan terukur daripada sistem manajemen database desktop seperti Microsoft Access. Penggunaan Access sebagai database pada website umumnya menghasilkan error yang lebih banyak dibanding SQL Server.

Meskipun SQL Server juga dapat dijalankan sebagai sebuah sistem database desktop, SQL Server paling sering digunakan sebagai sistem database server.

Server database System

Server berbasis sistem database merupakan database yang dirancang untuk berjalan pada server pusat, sehingga beberapa pengguna dapat mengakses data yang sama secara bersamaan. Para pengguna biasanya mengakses database melalui aplikasi.

Sebagai contoh, sebuah website dapat menyimpan semua konten di database. Setiap kali sebuah artikel dibaca pengunjung, mereka mengambil data dari database. Seperti yang Anda ketahui, situs web biasanya tidak terbatas hanya pada satu pengguna. Jadi, pada suatu saat tertentu, sebuah website bisa melayani sampai ratusan, atau bahkan ribuan artikel untuk pengunjung website. Pada saat yang sama, pengguna lain dapat memperbarui profil pribadi mereka pada menu anggota, atau berlangganan newsletter, atau apapun yang pengguna situs web lakukan.

3. Database (Diagram, Table, View, Stored Procure, User, Roles)

Diagram

Database Diagram adalah alat sebuah visual yang memungkinkan untuk merancang dan memvisualisasikan database-database yang berhubungan. Ketika merancang database, Anda dapat menggunakan Database Diagram untuk membuat, mengedit, atau menghapus tables, columns, keys, indexes, relationships, and constraints. Untuk memvisualisasikan database, Anda dapat membuat satu atau lebih diagram yang menggambarkan sebagian atau semua tables, columns, keys, dan hubungan di dalamnya.

Table

Fondasi dari setiap Relational Database Management System adalah objek database yang disebut table. Setiap database terdiri dari satu atau lebih tabel, yang menyimpan data / informasi database. Setiap tabel memiliki nama yang unik dan terdiri dari kolom dan baris. Setiap tabel terdiri dari serangkaian kolom yang berisi informasi tentang jenis data yang tersimpan dalam tabel. Setiap kolom harus diberi nama, tipe data, dan panjang. Tabel A juga dapat memiliki satu set foreign key, key constraint, rules dan index. Kolom, index, constraint, dan trigger pada suatu tabel akan dianggap properti dari tabel dalam database diagram.

View

Data pada SQL Server disimpan dalam bentuk tabel dan Anda dapat melakukan pengolahan data secara fisik pada tabel tersebut. Begitu juga dengan View, View dapat juga dipakai untuk pemanggilan data. Namun view tidak menyimpan data secara fisik pada disk. View adalah salah satu objek database yang merupakan turunan dari satu atau lebih tabel.

Dalam SQL Server, View merupakan tabel virtual. Sama seperti tabel sebenarnya, View terdiri dari baris dan kolom, dan Anda dapat mengambil data dari View (kadang-kadang bahkan update data dalam sebuah View). Field dalam tabel virtual View adalah bidang dari satu atau lebih tabel nyata dalam database.

Stored Procedur

Stored Procedure adalah sekelompok pernyataan Transact-SQL yang dikompilasi menjadi rencana eksekusi tunggal. Prosedur yang disimpan sangat membantu dalam mencapai implementasi yang konsisten dari logika di seluruh aplikasi.

Roles

Ketika membuat login user baru di SQL Server, Anda dapat memilih untuk menempatkan user baru ini ke satu atau lebih Server Roles(peran) pada server.

Server Roles (jangan dikelirukan dengan peran database) tersedia untuk melakukan berbagai tugas administrasi dalam database. Tidak semua orang bisa mendapatkan server role. Pada kenyataannya, hanya pengguna seperti database administrator yang diberikan server role.

4. Attach & Detach Database

File-file log data dan transaksi database dapat dilepaskan(Detach) dan kemudian disambungkan(Attach) ke instance yang sama atau beda dalam SQL Server. Attach dan Detach database berguna jika Anda ingin mengubah database pada instance yang berbeda dari SQL Server pada komputer yang sama atau untuk memindahkan database.

5. Import & Export Data

Import dan Export data digunakan agar dapat menggunakan data yang dihasilkan oleh aplikasi lain. Kemampuan untuk mengimpor data sangat penting dalam aplikasi perangkat lunak karena itu berarti bahwa satu aplikasi dapat melengkapi yang lain. Banyak program, misalnya, dirancang untuk dapat mengimpor grafis dalam berbagai format.

Kebalikan dari mengimpor adalah ekspor, yang mengacu pada kemampuan satu aplikasi agar dapat dipakai sesuai format data untuk aplikasi lain.

6. Backup & Restore Database

Salinan dari sebuah file atau direktori pada perangkat penyimpanan terpisah dapat kita buat backup jadi jika sewaktu-waktu data yang original rusak atau terhapus, masih bisa digunakan salinannya. Proses backup mengacu pada pembuatan salinan data sehingga salinan ini dapat digunakan untuk mengembalikan data semula setelah peristiwa kehilangan data.

Terdapat dua tujuan utama dalam backup. Yang pertama adalah untuk memulihkan keadaan setelah terjadi bencana (disaster recover). Yang kedua adalah untuk mengembalikan sejumlah file kecil setelah secara sengaja terhapus atau rusak. Kehilangan data merupakan hal sangat umum, 66% dari pengguna internet telah mengalami kehilangan data yang serius.

Restore berguna untuk mengembalikan ke dalam keadaan semula dan membuat restitusi. Backup dan Restore adalah proses menyalin data terlebih dahulu dengan tujuan spesifik untuk mengembalikan data yang sama setelah suatu peristiwa yang dapat mengakibatkan hilangnya data.

7. Query Execution

Apa yang sebenarnya membuat SQL Server berbeda dari bahasa pemrograman lain adalah cara SQL Server memproses kodenya. Umumnya, kebanyakan bahasa pemrograman memproses statement dari atas ke bawah. Sebaliknya, SQL Server memproses dalam urutan yang unik yang dikenal sebagai Logical Tahap Pengolahan Query. Fase ini menghasilkan serangkaian tabel-tabel virtual dengan masing-masing tabel virtual ke tahap berikutnya (tabel virtual tidak dapat dilihat). Fase dan perintah yang diberikan adalah sebagai berikut:

1. **SELECT**
2. **FROM**
3. **WHERE**
4. **GROUP BY**
5. **ORDER BY**
6. **HAVING**
7. **CUBE | ROLLUP**
8. **ON**
9. **DISTINCH**
10. **OUTER**
11. **TOP**

ASSESSMENT

Discuss each of the following terms:

- (a) data
- (b) database
- (c) database management system
- (d) database application program (e) data independence
- (f) security
- (g) integrity
- (h) views

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 1

MODULE 2

DATABASE ENVIRONMENT

DESKRIPSI TEMA

Describe concepts, terminologies, environment, and relational model in database system

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

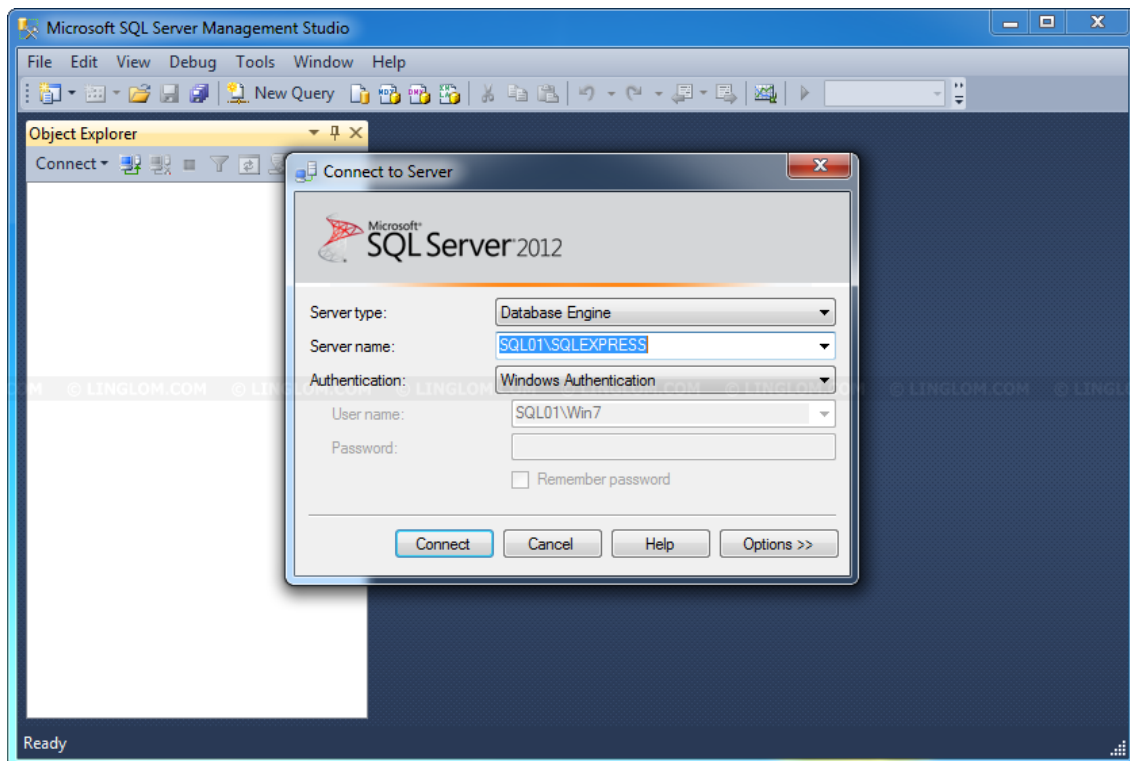
Student are able to Database Environment: Introduction general environment of SQL Server, New Server Registration & New Group, Attach Database, Import & Export Data, Generate SQL Script, SQL Query Analyzer, Execution SQL Statement

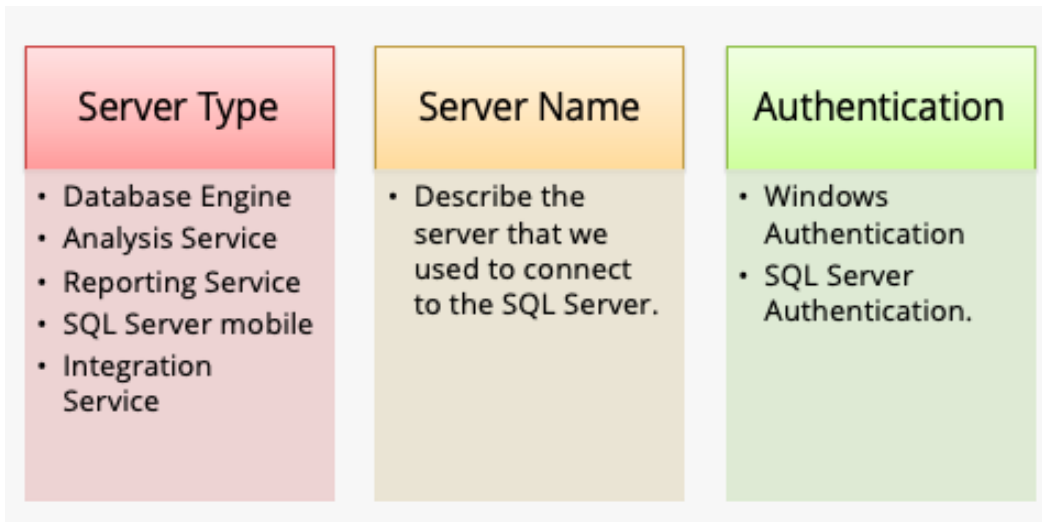
PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. Textbook: These module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 2

LANGKAH-LANGKAH PRAKTIKUM

1. Main screen of SQL Server 2012





2. New Server Registration

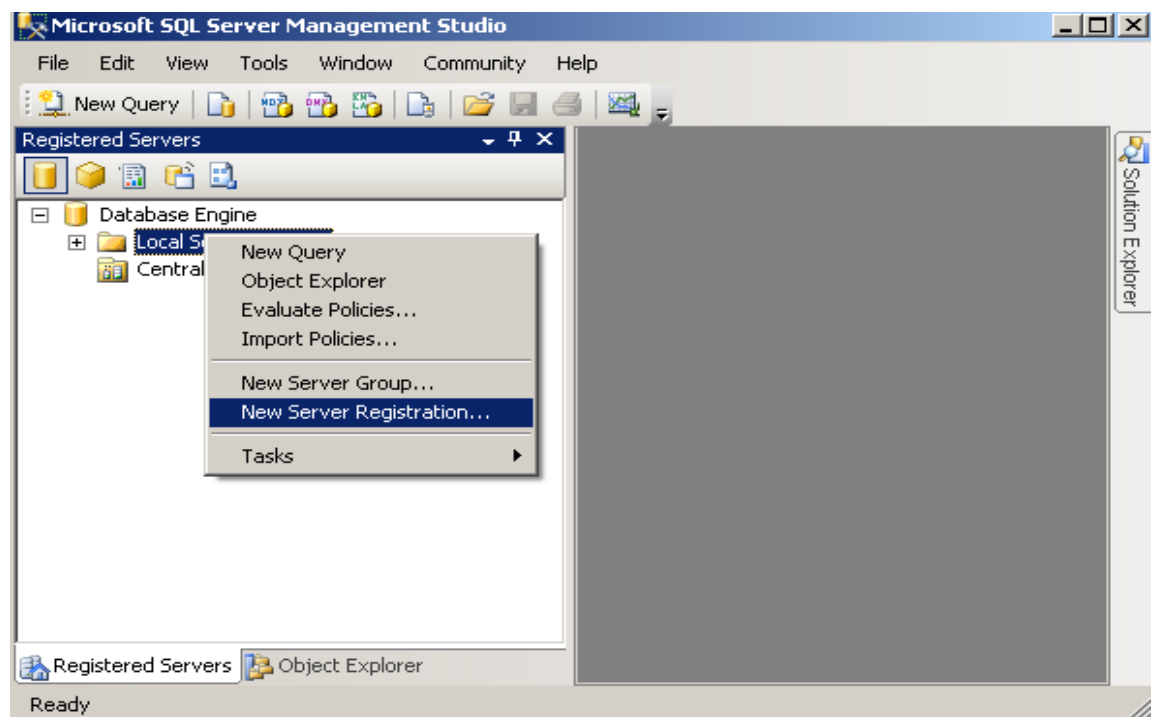
Two types of registered server:

- Local Server Groups
- Central Management Server

In *SQL Management Studio* you can create, modify or remove registered server.

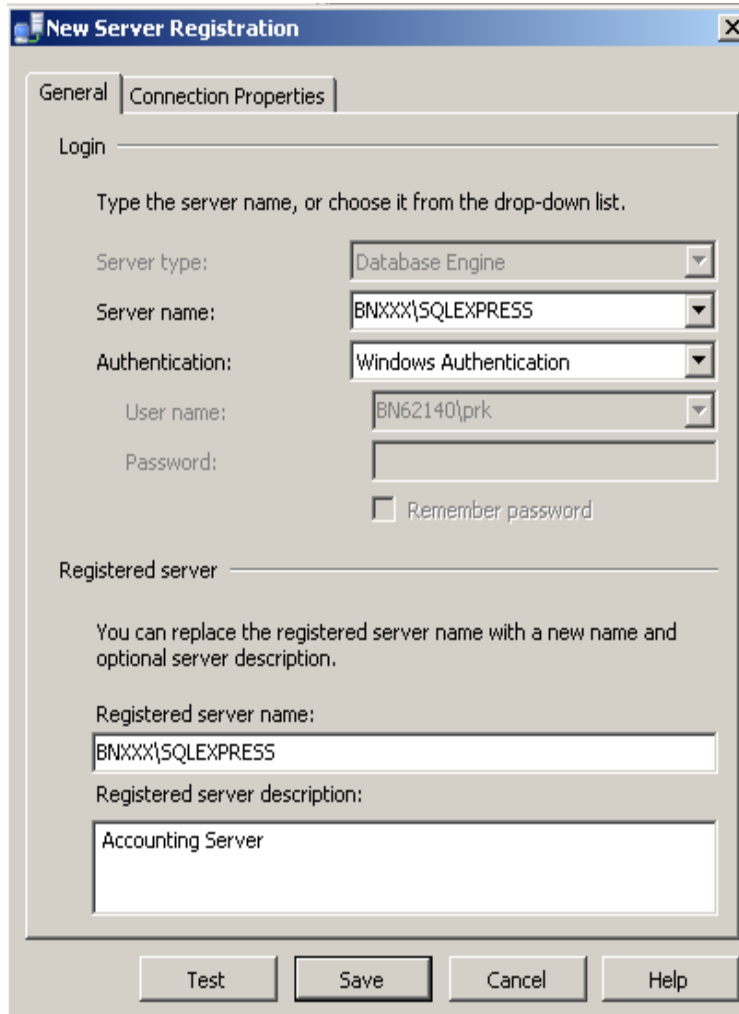
Registered server can be view on the **View** menu, and then click **Registered Servers**

- Click **Database Engine** on Registered Server toolbar.
- Right click on the **Local Server Groups** tree, and then click **New Server Registration**.

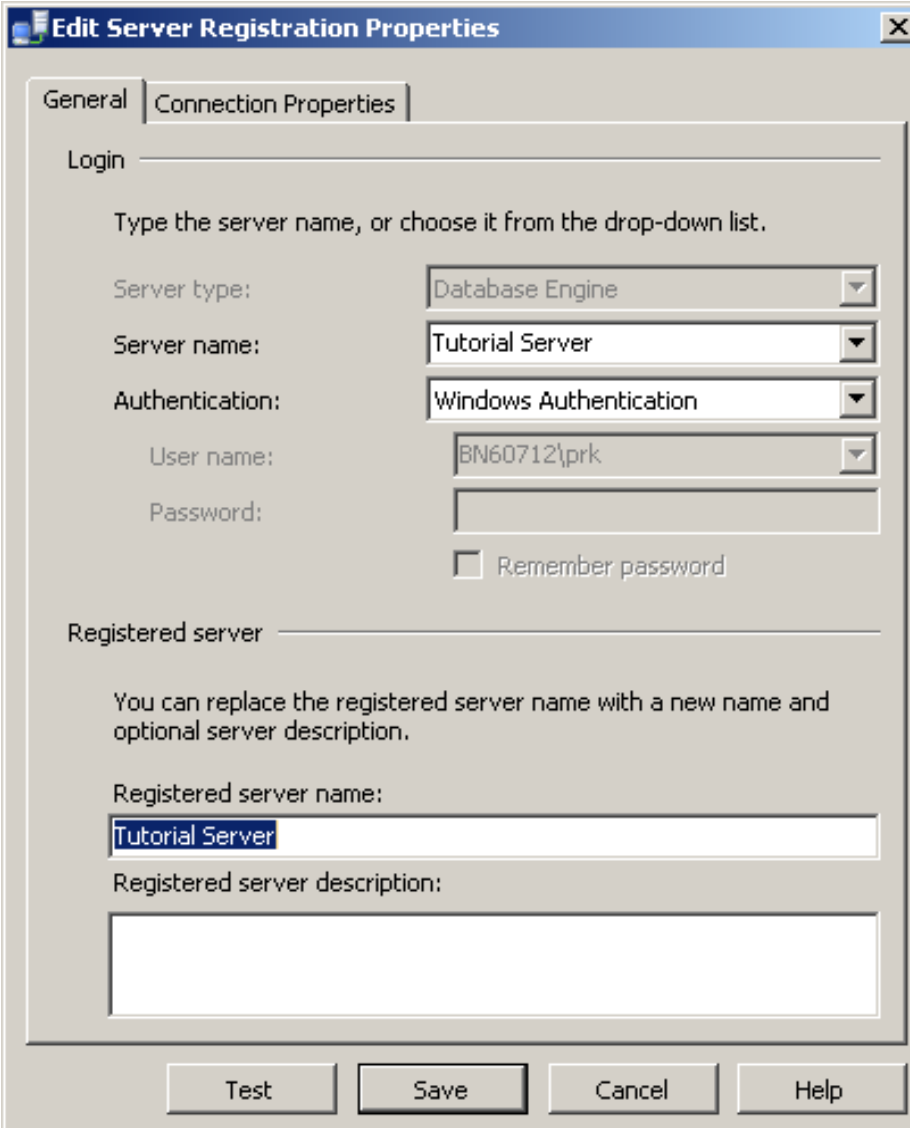


- On the **New Server Registration** dialog box there are :

- **Server name**, select the server instance to register in the format <servername>[<instancename>].
- **Authentication (Window Authentication and SQL Server Authentication)**
- **Username**
- **Login**
- **Password**
- **Registered server name**
- **Registered server description**
- To test the connection to the server selected in server name, click **Test**. To save the registered server settings, click **Save**.



- To **remove** the registered server, right click on the selected server and then click **Delete**, if confirmation dialog box appear click **Yes** to remove the server.
- To **modify** the server name, right click on the selected server and then click **Properties**, you can enter the new server name in the **Registered server name** textbox



Edit Server Registration Properties

General | **Connection Properties**

Login

Type the server name, or choose it from the drop-down list.

Server type: Database Engine

Server name: Tutorial Server

Authentication: Windows Authentication

User name: BN60712\prk

Password:

☐ Remember password

Registered server

You can replace the registered server name with a new name and optional server description.

Registered server name: Tutorial Server

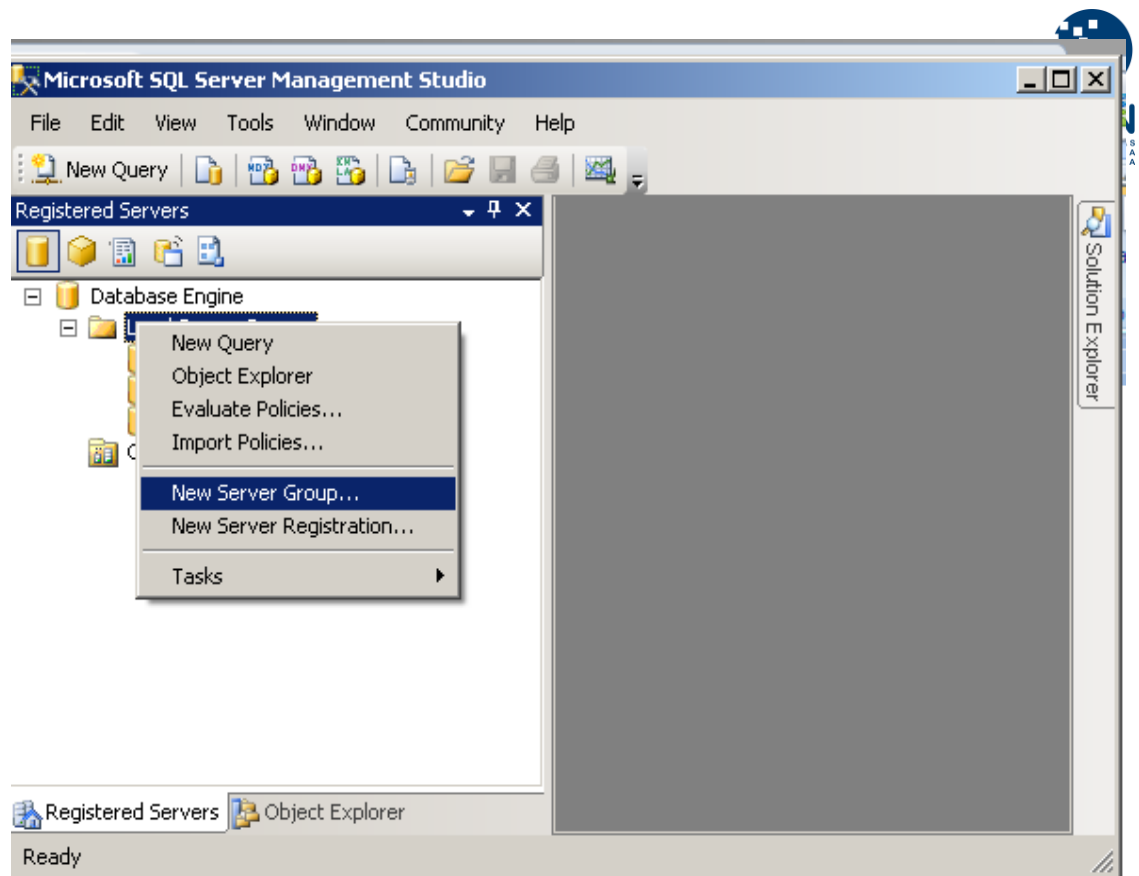
Registered server description:

Test Save Cancel Help

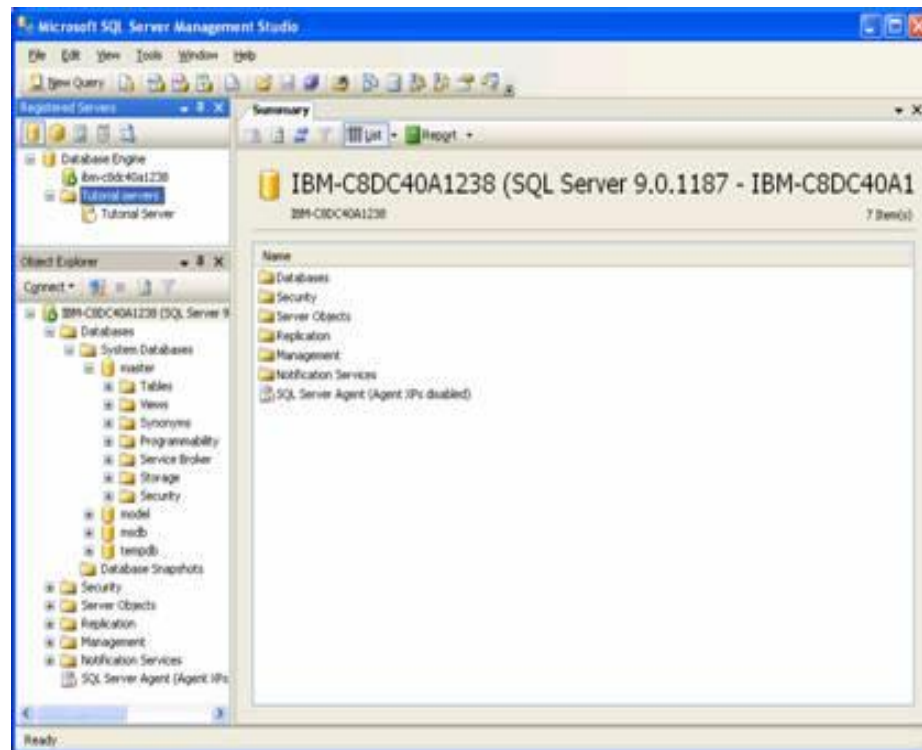
3. New Group

SQL Server Group is used to organize sets of computers running SQL Server. These groups can be organized by function or department. You can create subgroups within a group.

- Right click on the selected server or server group, and then click **New Server Group**

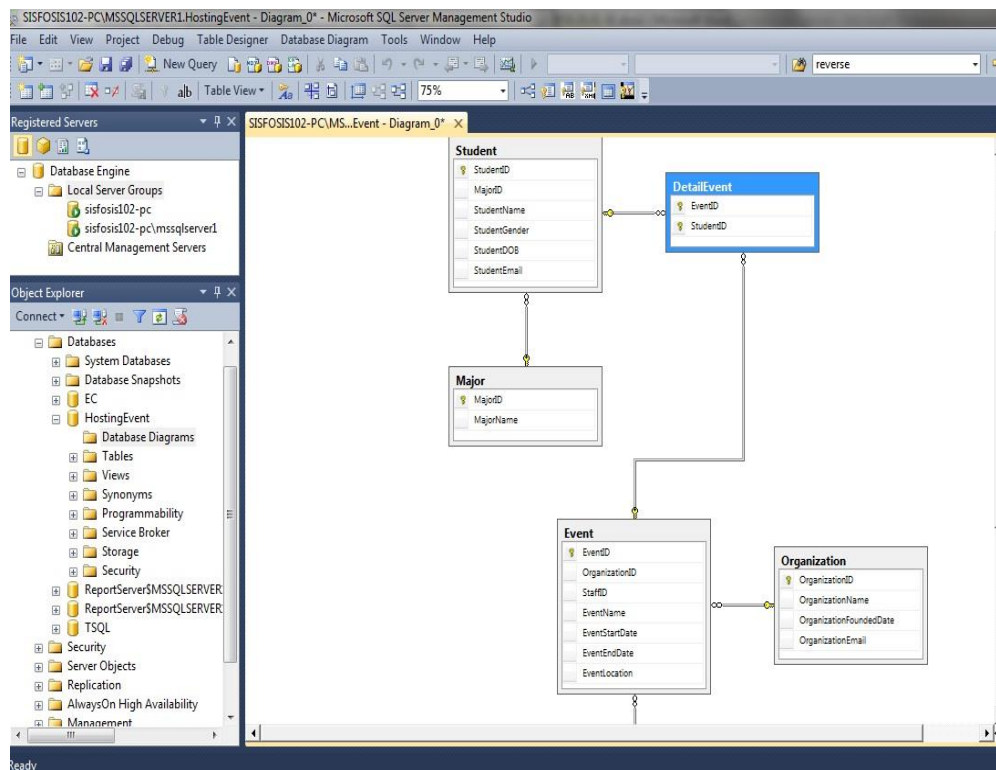


- In the New Server Group dialog box, type a name and the description for the new group. Click **Save**.
- You can see the new server group in the Registered Server toolbar.
- To **remove** a server group or a server, right click on the selected server name or server group name, click **Delete**, and in the confirmation dialog box click **Yes**.



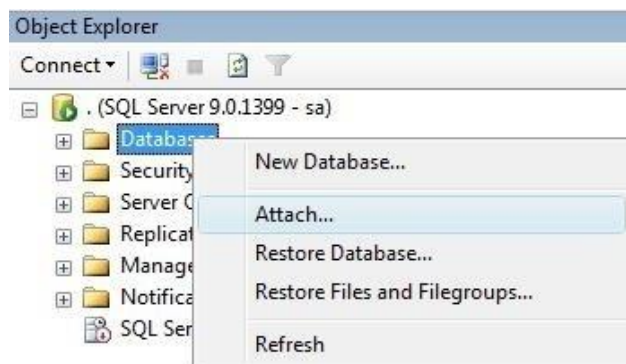
4. Diagram

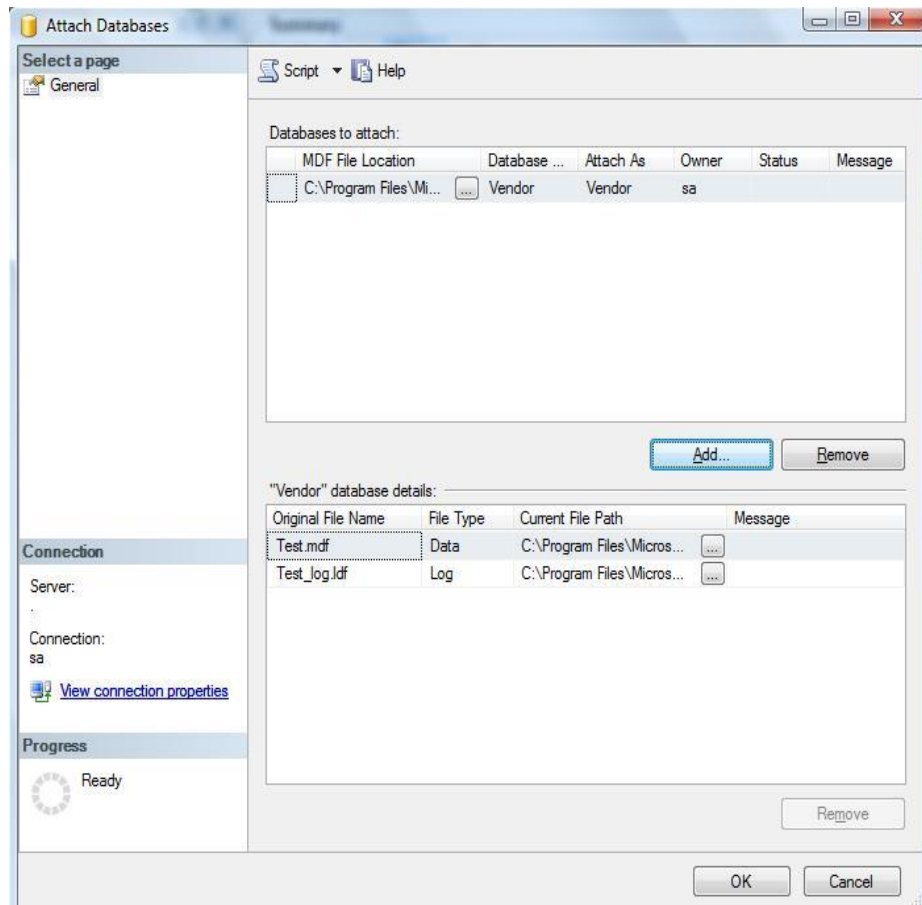
- Database Diagram is a visual tools that can visualized relational database. You can design database by using database diagram to create, edit, or delete tables, columns, keys, indexes, relationships, and constraints.
- To visualized database, you can create one or more database diagram that represent part or all columns, keys, and relations.
- Each diagram will asaved into related database.



5. Attach Database

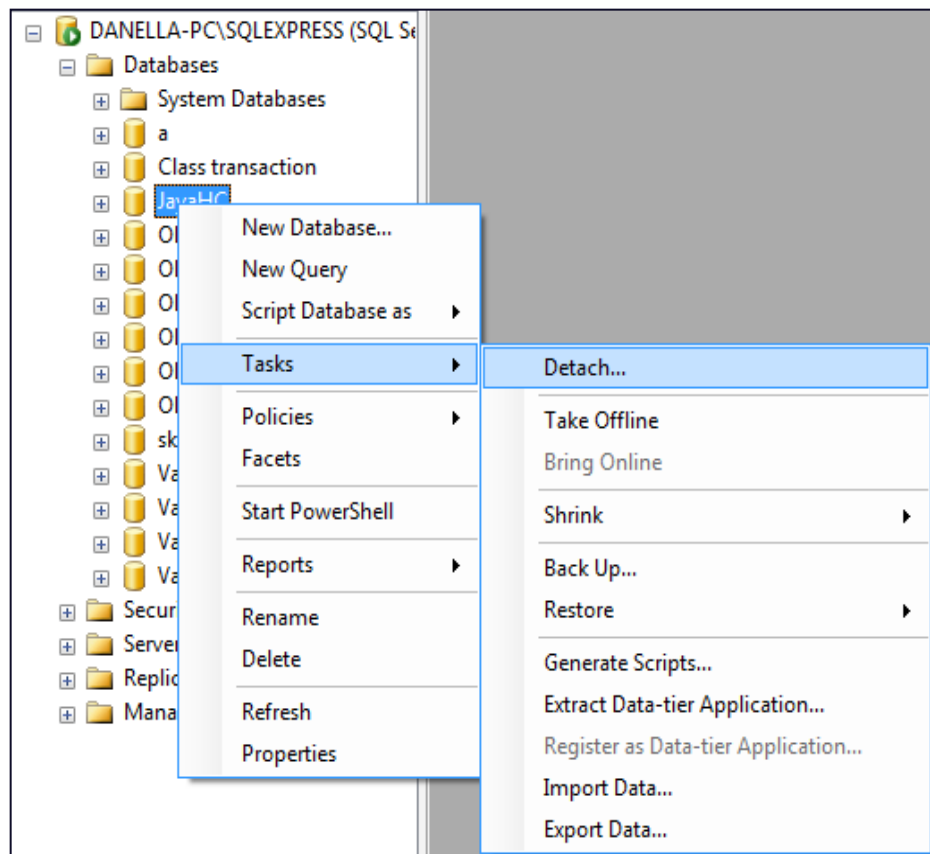
- Right click on Database tree and click Attach
- Click Add button to choose (.mdf) file from a database. This action will automatically added databases' log file on the detail panel. After you complete this steps, click OK button.





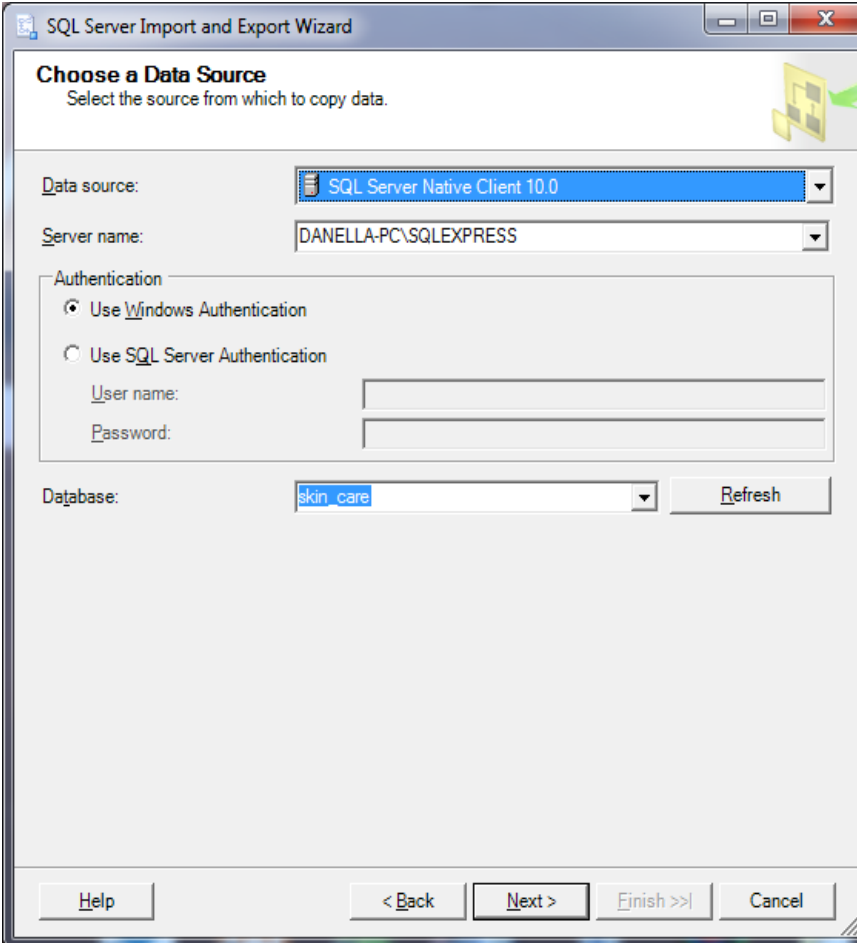
6. Detach Database

- Right click on selected database tree, choose Task menu and then click Detach. On Detach Database dialog box click OK button



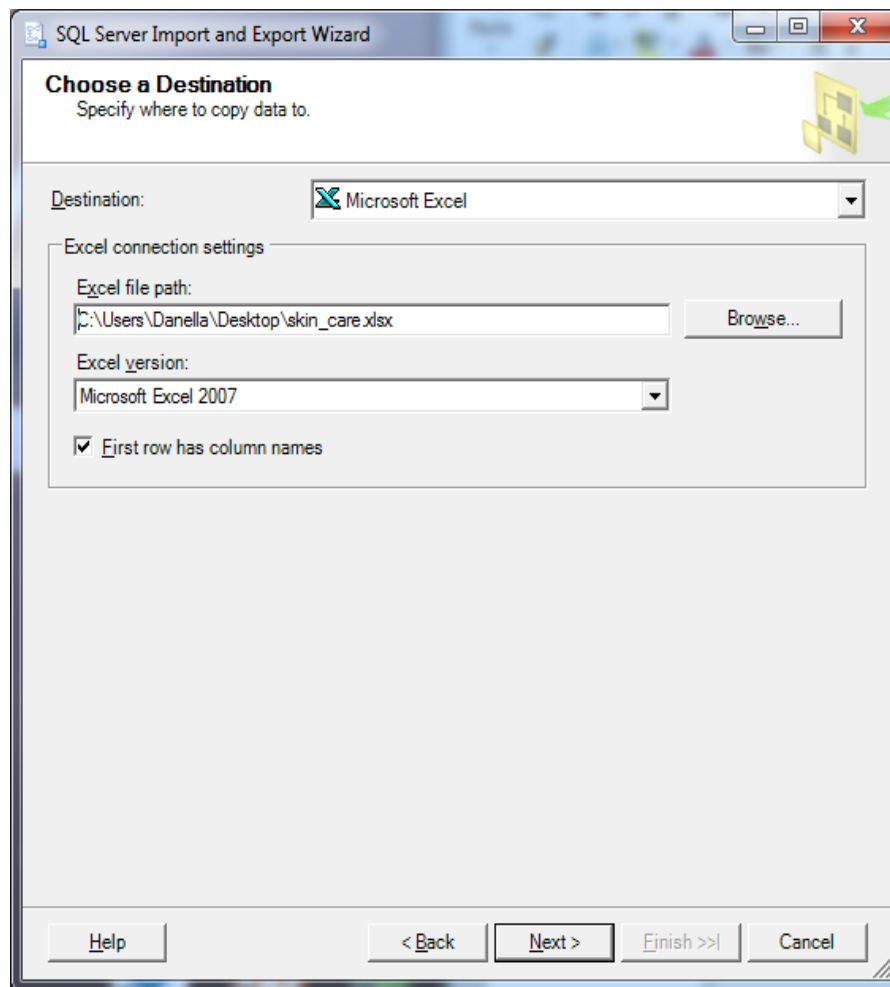
7. Export Data

- Right click on the selected database tree, choose Tasks menu, and then click Export Data.
- In SQL Server Import and Export Wizard welcome dialog box, click Next button
- In data source properties, set Data Source, Server Name, Authentication, and choose database that you want to export. After completing this setting, click Next button.

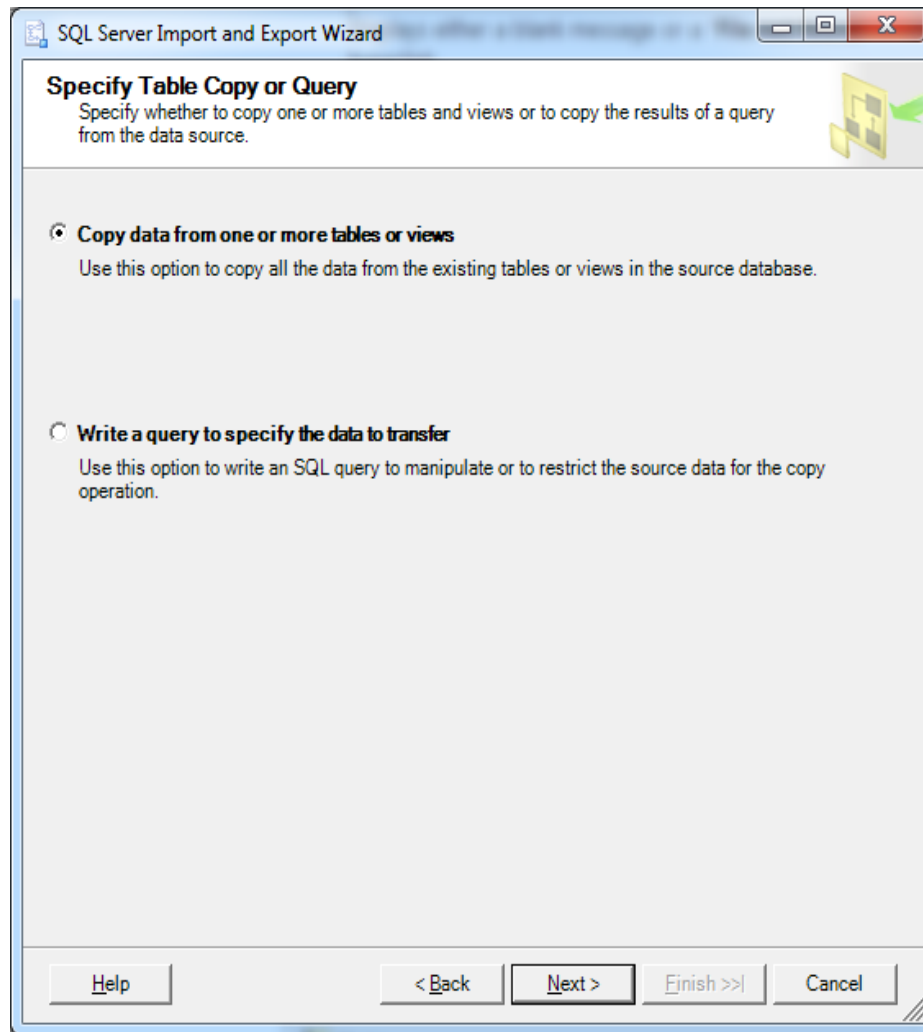


The screenshot shows the 'SQL Server Import and Export Wizard' window, specifically the 'Choose a Data Source' step. The window title is 'SQL Server Import and Export Wizard'. Below the title bar, the text 'Choose a Data Source' is displayed, followed by the instruction 'Select the source from which to copy data.' The main area contains several fields: 'Data source:' with a dropdown menu showing 'SQL Server Native Client 10.0'; 'Server name:' with a dropdown menu showing 'DANELLA-PC\SQLEXPRESS'; an 'Authentication' section with two radio buttons, 'Use Windows Authentication' (selected) and 'Use SQL Server Authentication'; 'User name:' and 'Password:' text boxes; and 'Database:' with a dropdown menu showing 'skin_care' and a 'Refresh' button. At the bottom, there are buttons for 'Help', '< Back', 'Next >', 'Finish >>', and 'Cancel'.

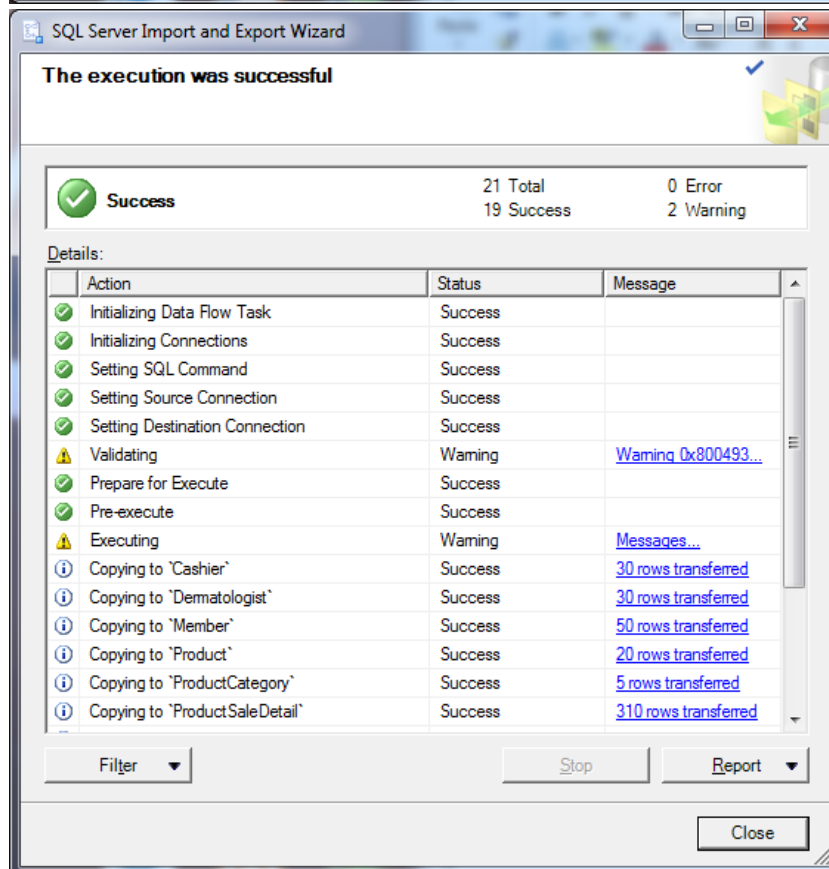
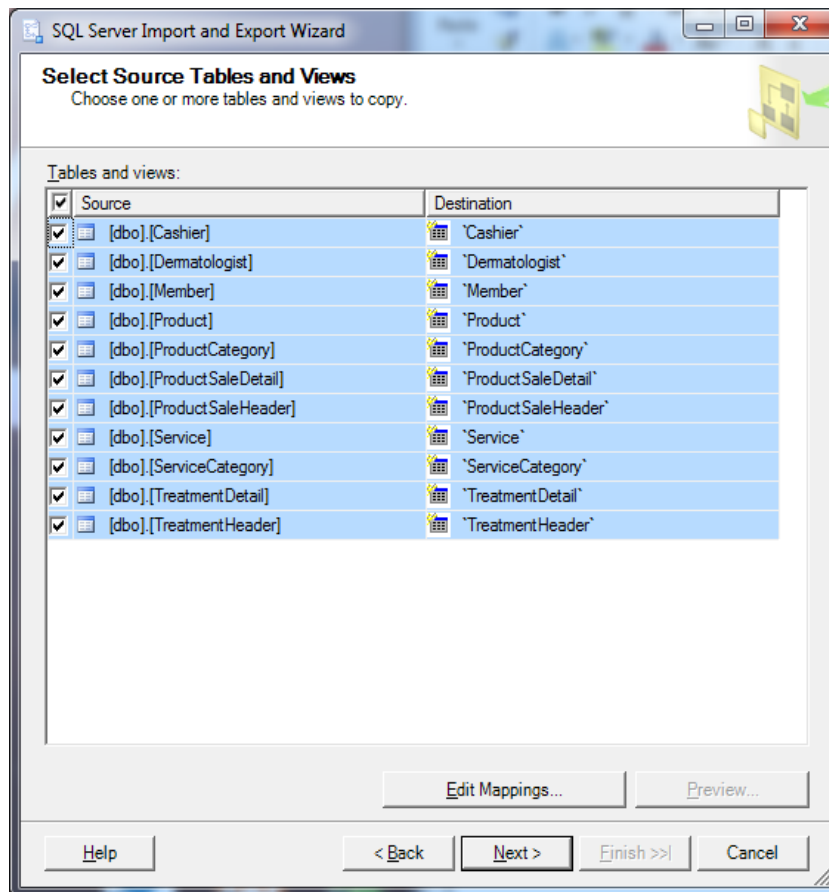
- In destination properties, set Destination (as an example chooses Microsoft Excel), choose where the Excel file resides by click Browse button, choose Excel version, and then click Next button.



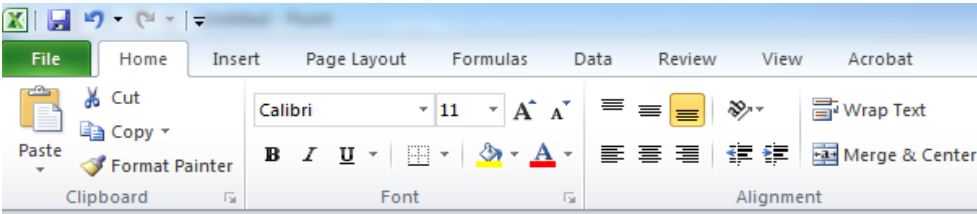
- In the Specify Table Copy or Query page, there are two types of copy operation: copy data from one or more tables or views and write a query to specify the data to transfer. After choosing one of those options, click Next button.



- In the Source Tables and Views page, choose table that you want to export. After completing this steps, click Next button.
- In Review Data Type Mapping page, change On Error and On Truncation option into Ignore. After completing this steps, click Finish button.
- To complete export these database, click Close button.

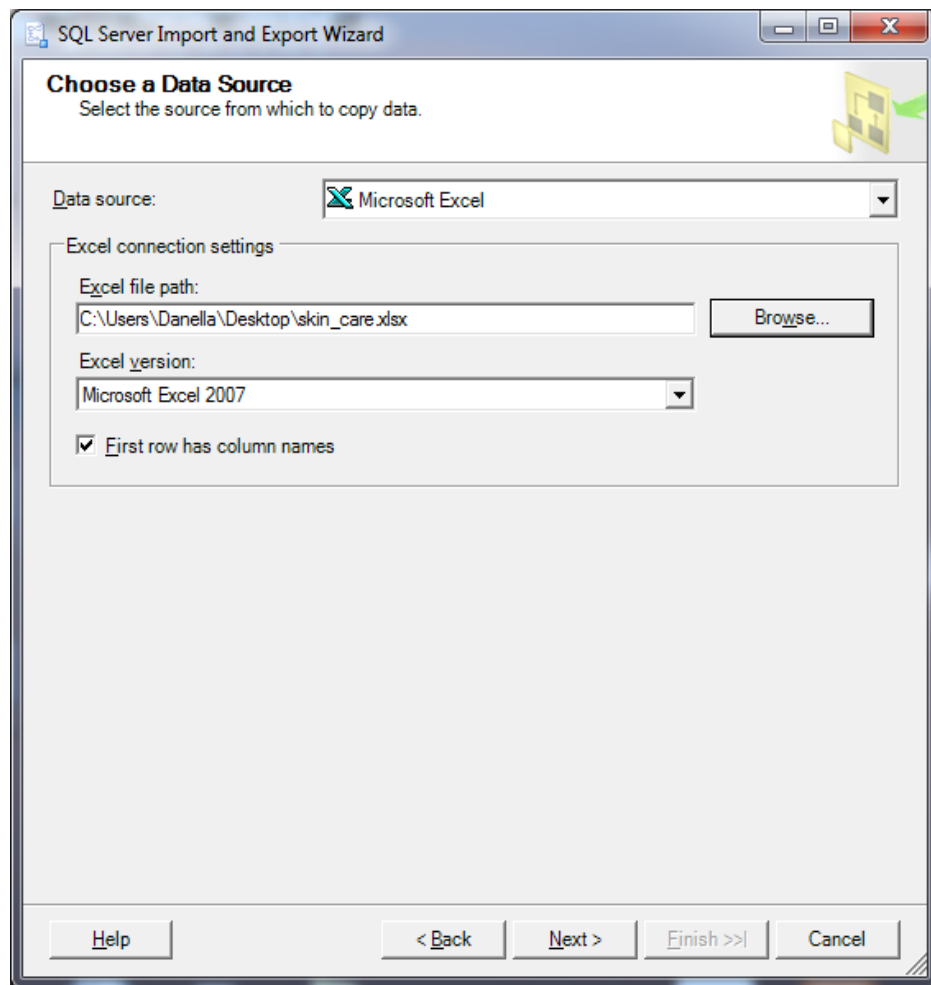


- After data exported successfully, you will be able to see all data in excel.

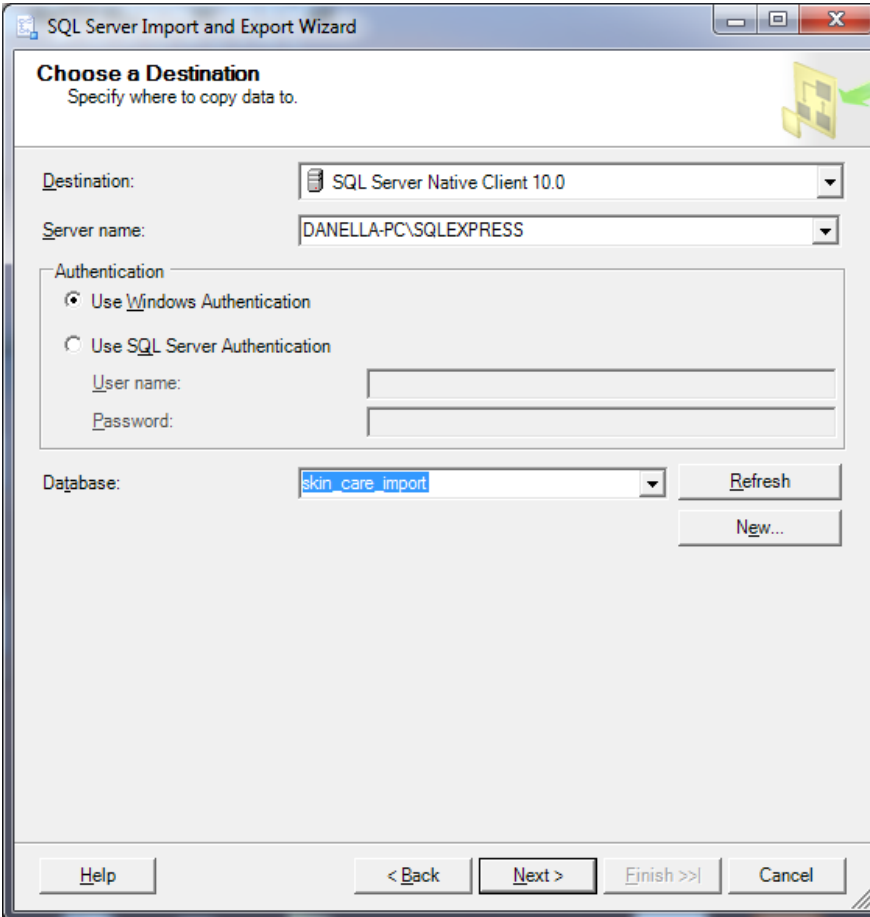


	A	B	C	D	E
1	CashierId	CashierName	CashierAddress	CashierPhone	CashierGender
2	CS000001	Nissim Duke	6174 Vitae Rd.	62 21-474-0809	Male
3	CS000002	Elmo Castillo	Ap #103-558 Dapibus Avenue	62 21-511-3311	Female
4	CS000003	Courtney Walsh	P.O. Box 799, 1864 Ut Av.	62 21-653-5531	Male
5	CS000004	Wendy Stokes	258 Ut, St.	62 819-6045-9392	Female
6	CS000005	Scott Huber	P.O. Box 611, 6366 Nonummy Street	62 21-530-2377	Female
7	CS000006	Teagan Sellers	P.O. Box 385, 7868 Imperdiet Road	62 21-157-9367	Female
8	CS000007	Cassandra Skinner	433-9522 Amet St.	62 21-414-9034	Female
9	CS000008	Oprah Ingram	Ap #768-2704 Maecenas Ave	62 21-485-7021	Female
10	CS000009	Travis Burks	4904 Vitae Rd.	62 21-511-1758	Female
11	CS000010	Benedict Clayton	Ap #398-9623 Eu Road	62 21-516-0018	Female
12	CS000011	Tatum Kirk	446-6446 Vestibulum, Rd.	62 815-3068-3842	Male
13	CS000012	Sharon Raymond	368-8325 Tellus, Ave	62 813-4681-3157	Female
14	CS000013	Giacomo Whitney	268-5858 A Avenue	62 814-2346-4179	Female
15	CS000014	Rhea Fisher	2316 Donec Ave	62 21-971-2886	Female
16	CS000015	Mona Powell	137-6863 Lacus, Ave	62 21-877-7147	Male
17	CS000016	Abdul Bowman	P.O. Box 581, 4889 Sapien. Street	62 812-8279-6683	Male
18	CS000017	Adria Santiago	1493 Euismod Ave	62 819-6930-0191	Male
19	CS000018	Abigail York	6482 Donec Ave	62 812-3201-5280	Female
20	CS000019	Leroy Maxwell	Ap #594-1988 Phasellus Avenue	62 819-6424-1447	Male
21	CS000020	Nevada Lester	P.O. Box 289, 8425 Fermentum Avenue	62 812-7109-6048	Male
22	CS000021	Gay Charles	1343 Vitae Avenue	62 21-613-1385	Male
23	CS000022	Deirdre Nichols	Ap #670-9261 Eget Rd.	62 813-1944-8157	Female
24	CS000023	Larissa Peck	1304 Netus Street	62 819-5347-6030	Male
25	CS000024	Vaughan Dunlap	P.O. Box 322, 3360 Vehicula St.	62 815-8580-8088	Female
26	CS000025	Aristotle Hall	2862 Purus. Ave	62 810-5797-5370	Male
27	CS000026	Jena Copeland	Ap #383-1466 Dolor, Rd.	62 21-540-9941	Male
28	CS000027	Dominique Sellers	958-2987 Duis Rd.	62 21-544-6464	Male
29	CS000028	Callie Nicholson	6778 Parturient Avenue	62 21-137-2799	Male
30	CS000029	Grady Collins	445-2893 Risus, Road	62 21-685-0373	Male

- Right click on the selected database tree, choose Tasks menu, and then click Import Data.
- In the Data Source properties, set Data Source (as an example chooses Microsoft Excel), choose where the Excel file resides by click Browse button, choose Excel version, and then click Next button.



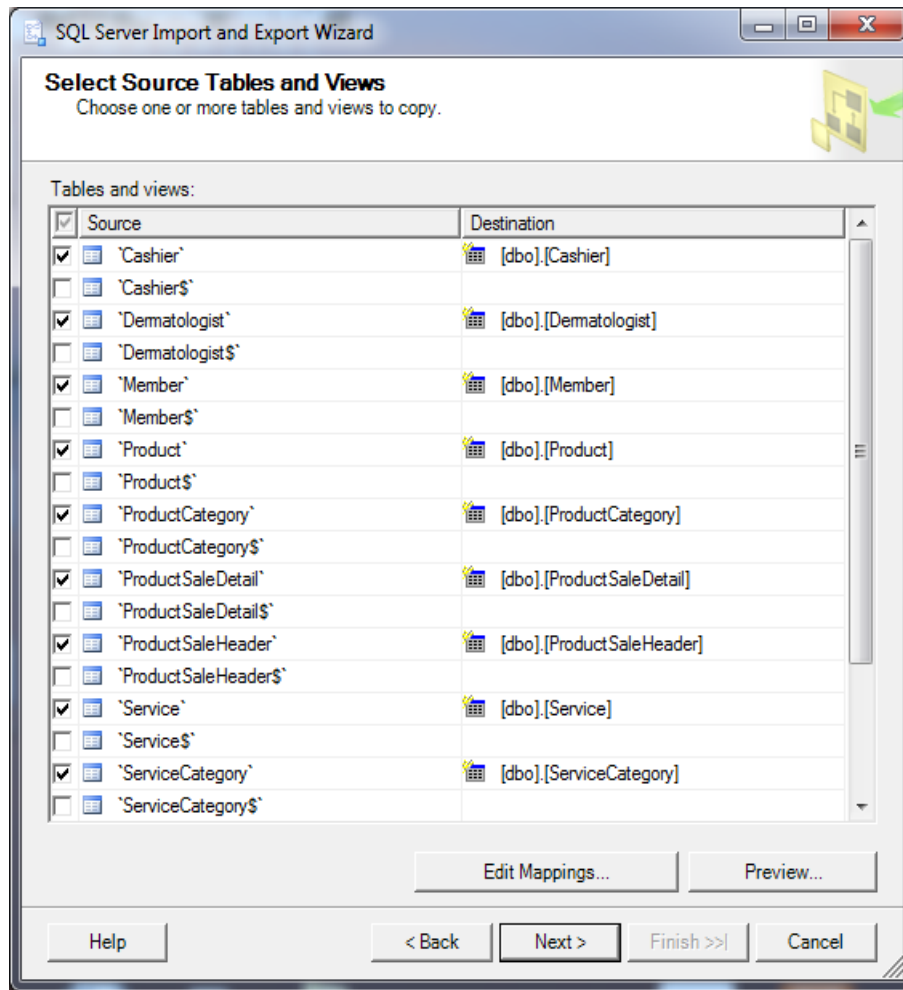
- In the destination properties, set Destination (as an example chooses SQL Server Native Client 10.0), Server Name, Authentication, and Database.
- In the Specify Table Copy or Query page, there are two types of copy operation: copy data from one or more tables or views and write a query to specify the data to transfer. After choosing one of those options, click Next button.



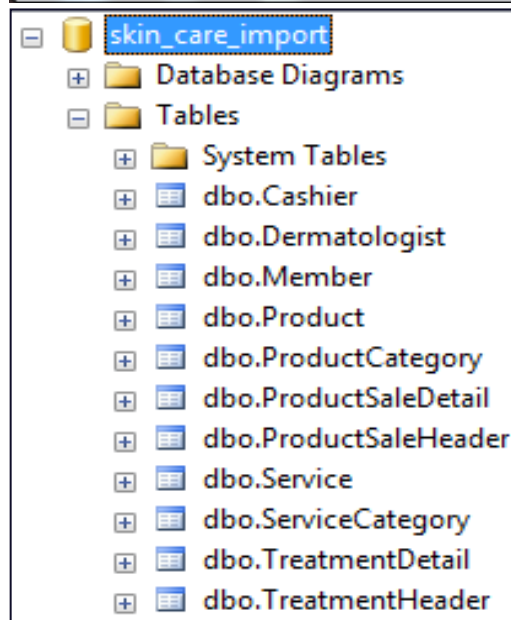
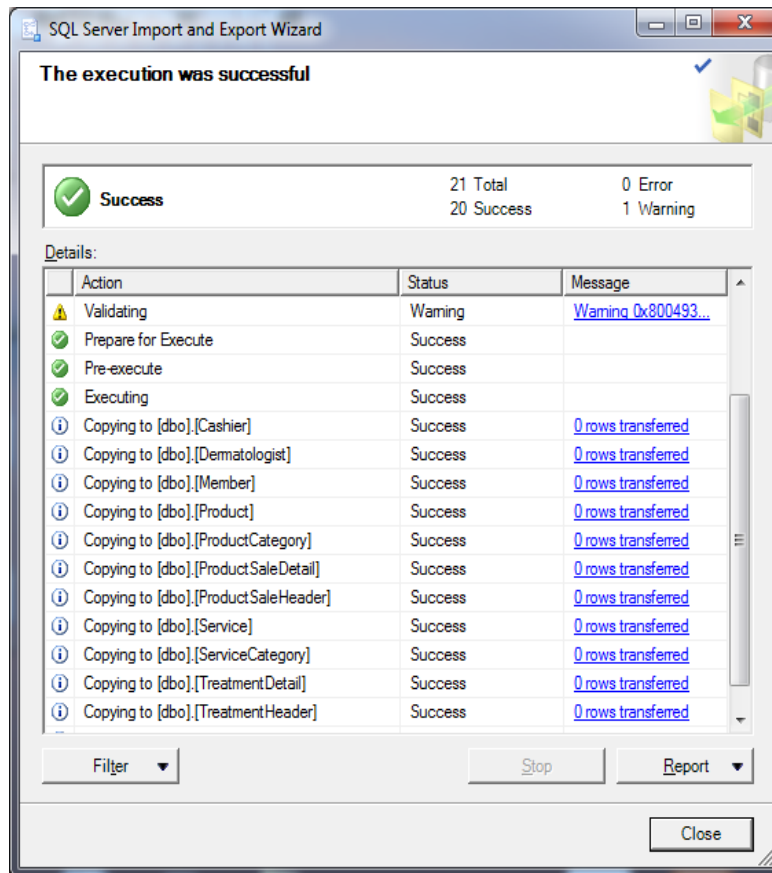
The screenshot shows the 'SQL Server Import and Export Wizard' window, specifically the 'Choose a Destination' step. The window title is 'SQL Server Import and Export Wizard'. The subtitle is 'Choose a Destination' with the instruction 'Specify where to copy data to.' Below this, there are several fields and buttons:

- Destination:** A dropdown menu showing 'SQL Server Native Client 10.0'.
- Server name:** A dropdown menu showing 'DANELLA-PC\SQLEXPRESS'.
- Authentication:** A section with two radio buttons:
 - ☒ Use Windows Authentication
 - ☐ Use SQL Server Authentication
- User name:** An empty text box.
- Password:** An empty text box.
- Database:** A dropdown menu showing 'skin_care_import'.
- Buttons:** 'Refresh' and 'New...' buttons are located to the right of the Database dropdown.
- Navigation:** At the bottom, there are buttons for '< Back', 'Next >', 'Finish >>', and 'Cancel'. A 'Help' button is also present on the left.

- In the Source Tables and Views page, choose table that you want to import. After completing this steps, click Finish button.

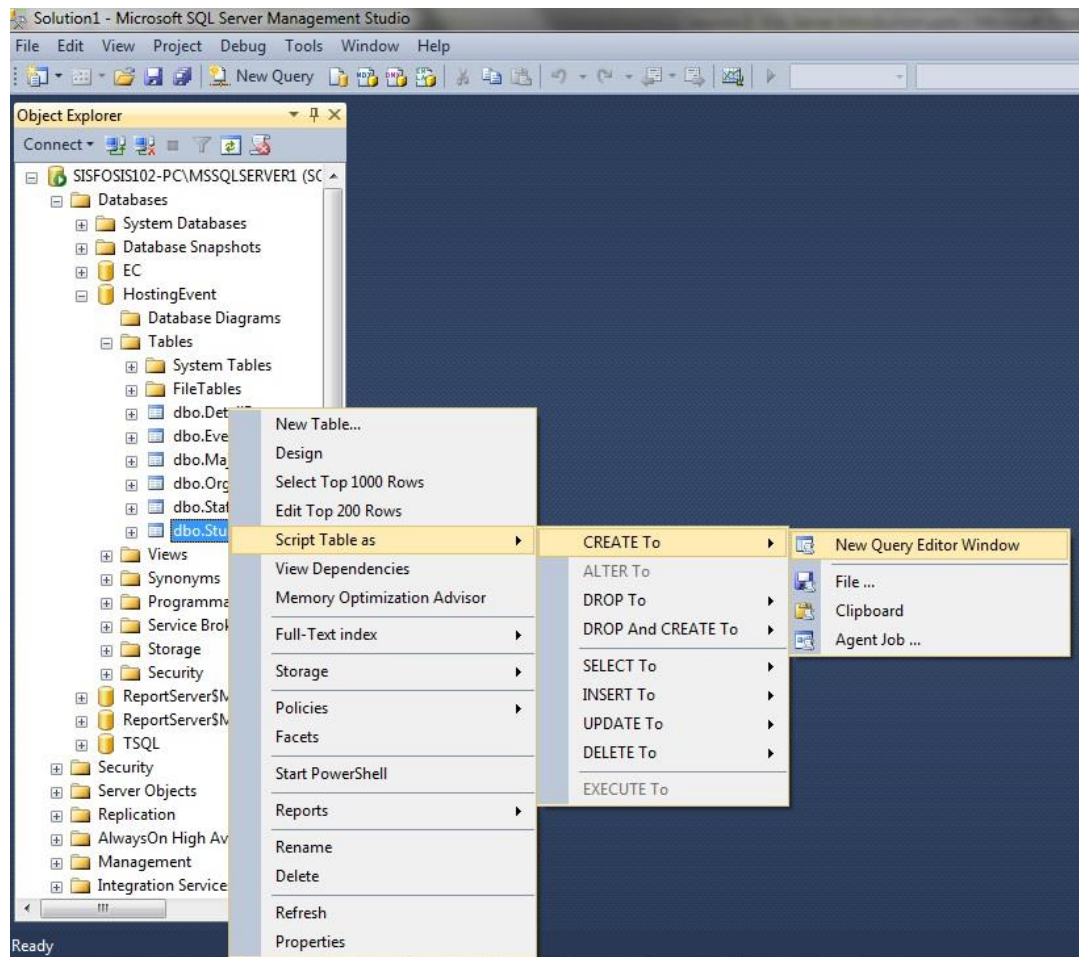


- After excel file imported successfully, now you can see the table in selected database has been create by importing an excel file.

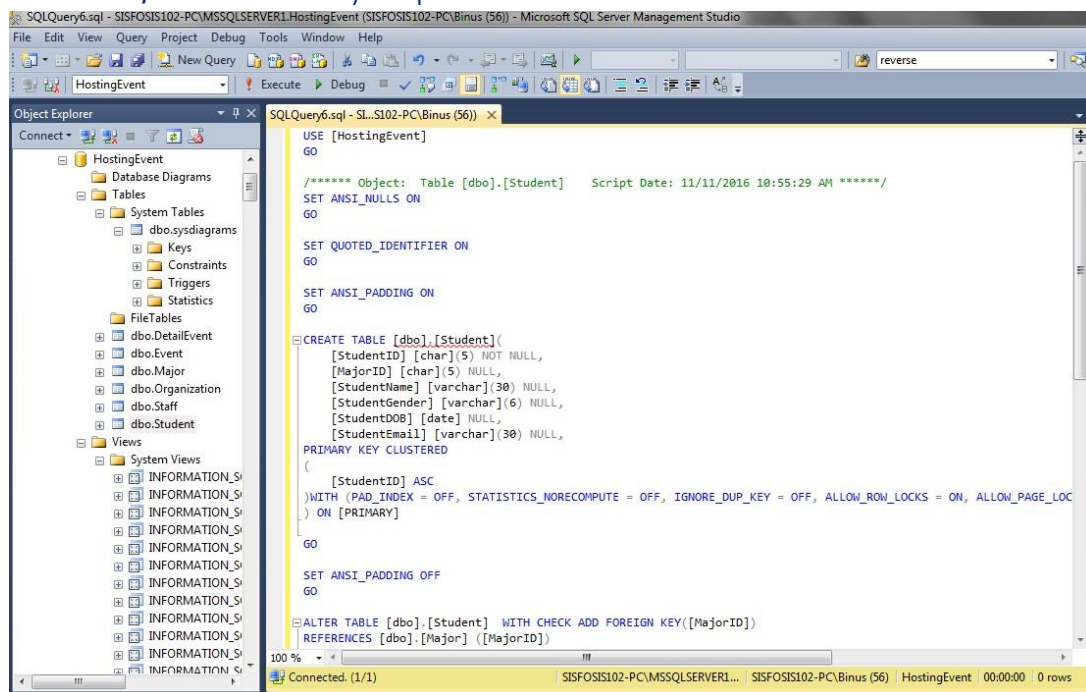


8. Generate SQL Script

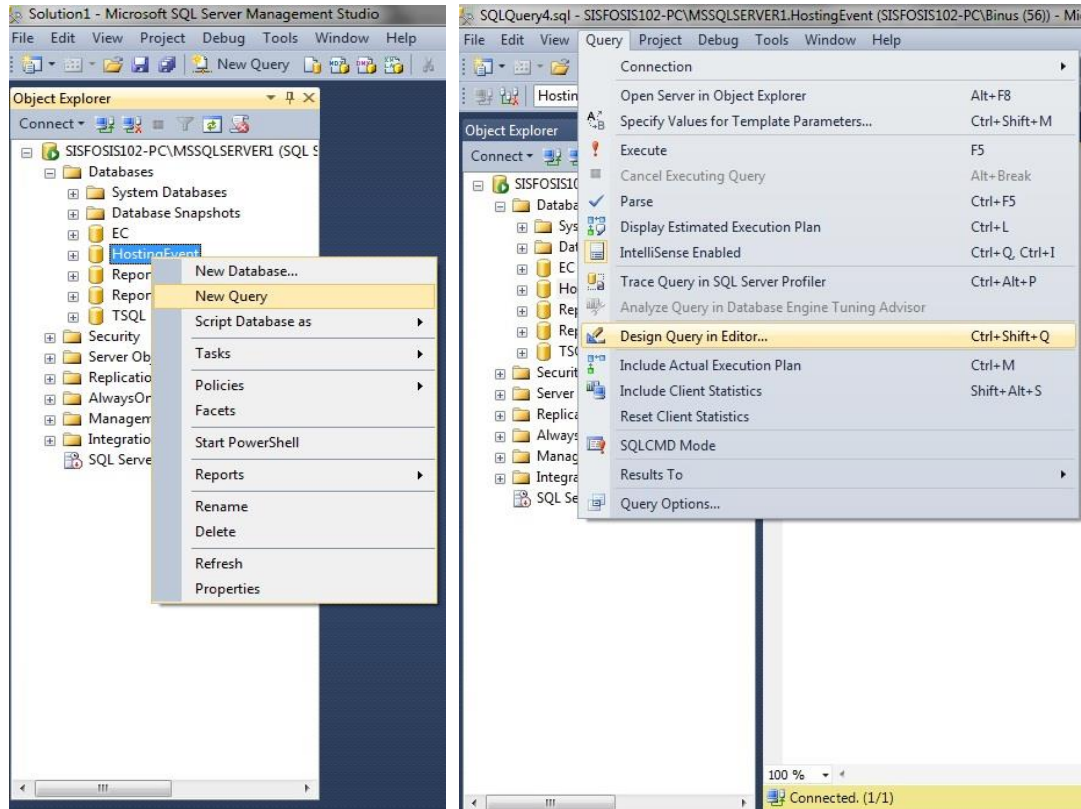
- In SQL Server, you can generate SQL script based on the table or query that you created before
- Right click on the selected database table, choose Script Table as menu, choose the query that you want to script, then click New Query Editor Window.



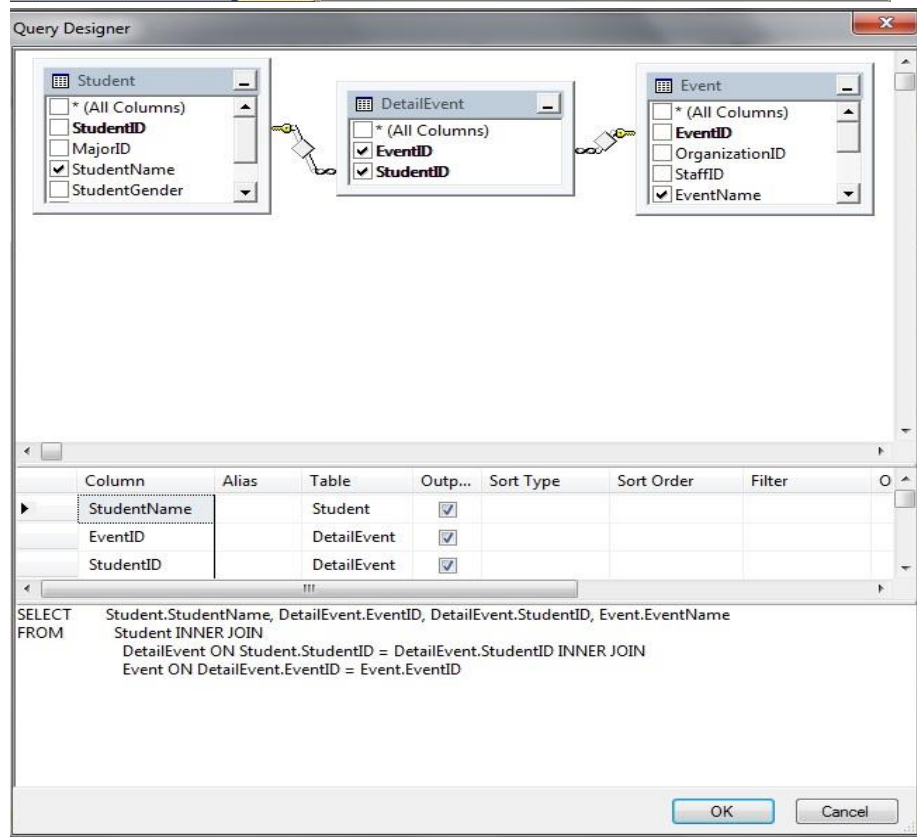
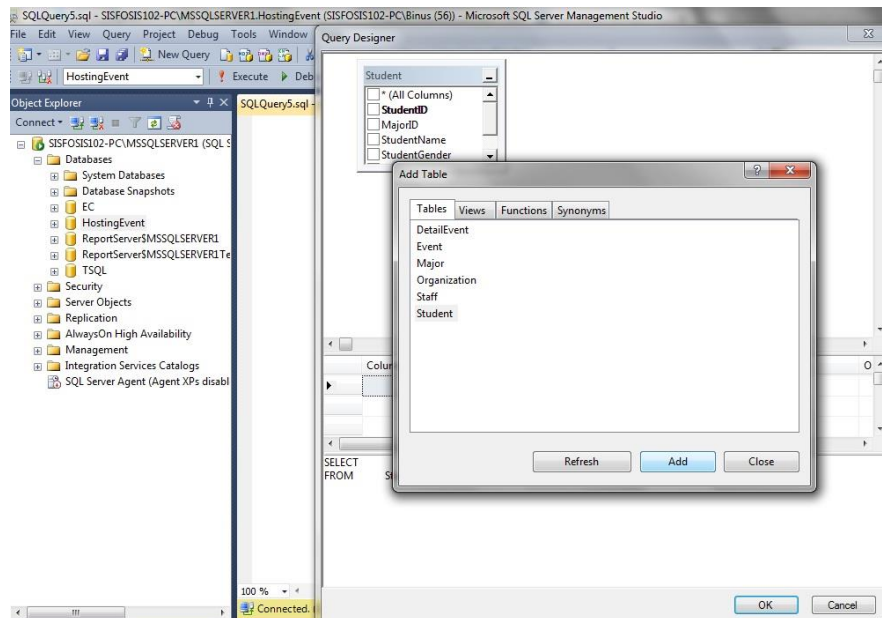
- In this case, we choose Query Script to Create Student table



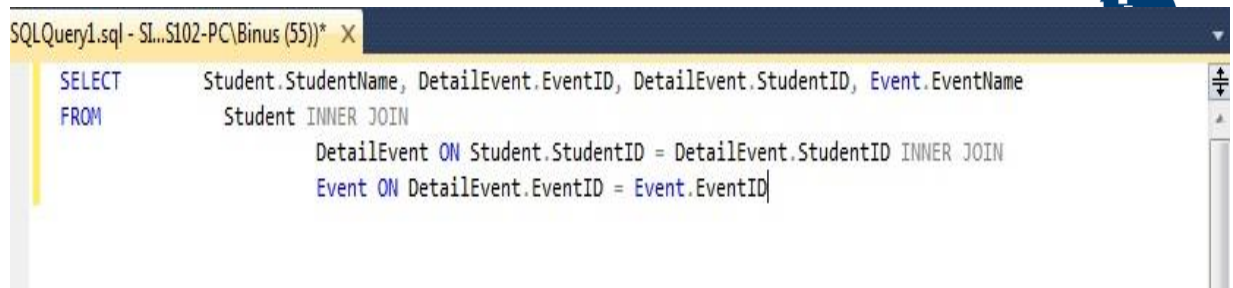
- We also can generate SQL Script from join table by using Query Editor
- Right click on the selected database choose New Query menu
- After that there will be Query menu in the Toolbars. Click on it and choose Design Query in Editor



- Add the tables that you want to draw relations
- After that SQL Server will show you the table relations with SQL statement below the relational tables



- Then click ok, and query statement will be appear as follow :

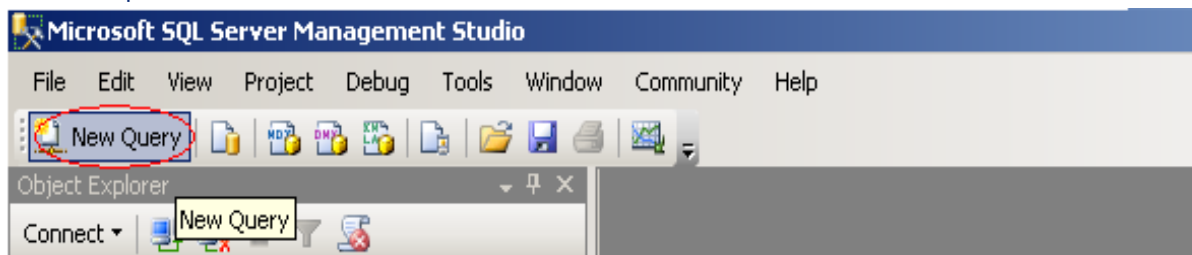


```
SQLQuery1.sql - SI...S102-PC\Binus (55)) * X
SELECT      Student.StudentName, DetailEvent.EventID, DetailEvent.StudentID, Event.EventName
FROM        Student INNER JOIN
            DetailEvent ON Student.StudentID = DetailEvent.StudentID INNER JOIN
            Event ON DetailEvent.EventID = Event.EventID
```

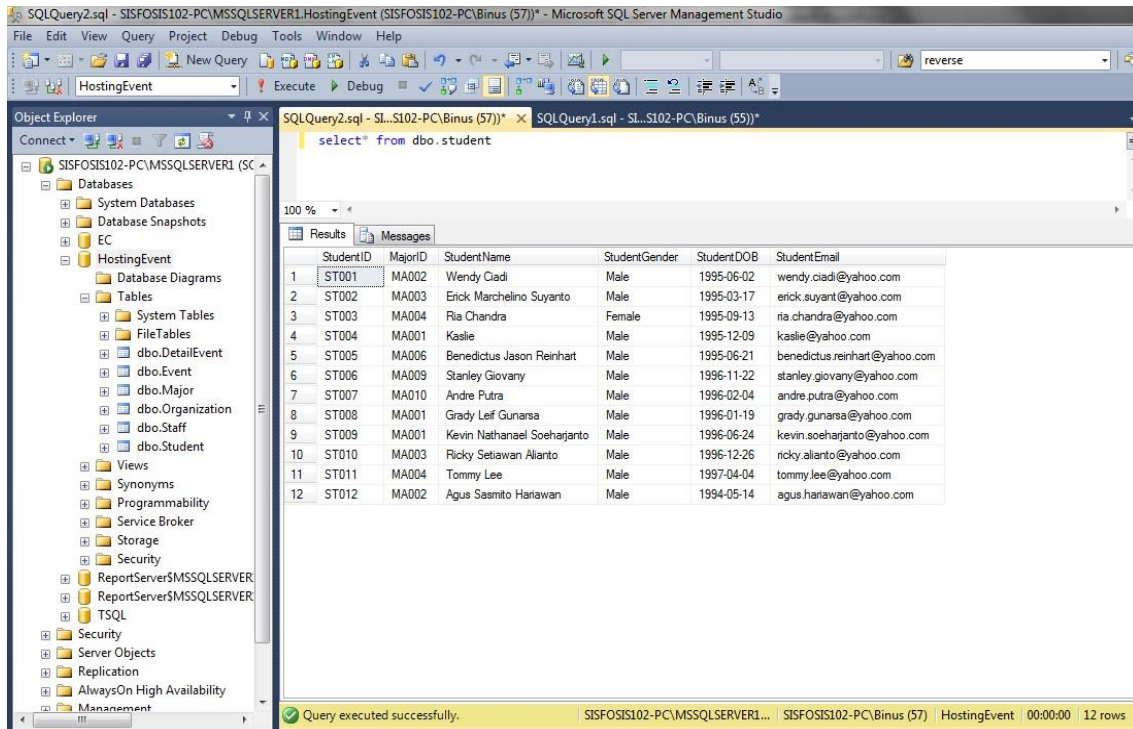
9. SQL Query Analyzer

Query Analyzer is graphical tool on Microsoft SQL Server that can perform below operations :

- a. Create database
- b. Manipulates and connect to other databases
- c. Provide remote data access (RDA) and replication
- d. Create and modify table and index on database
- e. Display query objects using select statement
- f. Insert and delete data on the tables
- g. Modify the existing table
- h. Repair the database



10. Execution SQL Statement



SQLQuery2.sql - SISFOSIS102-PC\MSSQLSERVER1.HostingEvent (SISFOSIS102-PC\Binus (57)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

HostingsEvent

Object Explorer

SISFOSIS102-PC\MSSQLSERVER1 (SC)

Databases

System Databases

Database Snapshots

EC

HostingsEvent

Database Diagrams

Tables

System Tables

FileTables

dbo.DetailEvent

dbo.Event

dbo.Major

dbo.Organization

dbo.Staff

dbo.Student

Views

Synonyms

Programmability

Service Broker

Storage

Security

ReportServer\MSSQLSERVER

ReportServer\MSSQLSERVER

TSQL

Security

Server Objects

Replication

AlwaysOn High Availability

Management

SQLQuery2.sql - SL...S102-PC\Binus (57)) * SQLQuery1.sql - SL...S102-PC\Binus (55)) *

select * from dbo.student

Results

StudentID	MajorID	StudentName	StudentGender	StudentDOB	StudentEmail	
1	ST001	MA002	Wendy Cadi	Male	1995-06-02	wendy.cadi@yahoo.com
2	ST002	MA003	Erick Marchelino Suyanto	Male	1995-03-17	erick.suyant@yahoo.com
3	ST003	MA004	Ria Chandra	Female	1995-09-13	ria.chandra@yahoo.com
4	ST004	MA001	Kaslie	Male	1995-12-09	kaslie@yahoo.com
5	ST005	MA006	Benedictus Jason Reinhart	Male	1995-06-21	benedictus.reinhart@yahoo.com
6	ST006	MA009	Stanley Giovany	Male	1996-11-22	stanley.giovany@yahoo.com
7	ST007	MA010	Andre Putra	Male	1996-02-04	andre.putra@yahoo.com
8	ST008	MA001	Grady Lelf Gunarsa	Male	1996-01-19	grady.gunarsa@yahoo.com
9	ST009	MA001	Kevin Nathanael Soeharjanto	Male	1996-06-24	kevin.soeharjanto@yahoo.com
10	ST010	MA003	Ricky Setiawan Alianto	Male	1996-12-26	ricky.alianto@yahoo.com
11	ST011	MA004	Tommy Lee	Male	1997-04-04	tommy.lee@yahoo.com
12	ST012	MA002	Agus Sasmito Hariawan	Male	1994-05-14	agus.hariawan@yahoo.com

Query executed successfully. SISFOSIS102-PC\MSSQLSERVER1... SISFOSIS102-PC\Binus (57) | HostingsEvent | 00:00:00 | 12 rows

LATIHAN

Silahkan dikumpulkan kegiatan praktik hari ini menggunakan SQL server dengan mengumpulkan berupa screenshot beserta keterangan/ judul dan pada query dikumpulkan .sql!

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 2

MODULE 3

SQL DATA DEFINITION I



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Definition: Create & Drop Database, Create & Drop Table, Create & Drop Index, and Alter Table.

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 7

LANGKAH-LANGKAH PRAKTIKUM

1. Create Database

```
CREATE DATABASE database_name
```

Drop Database

```
DROP DATABASE database_name
```

Create Table

```
CREATE TABLE table_name  
(  
field1_name datatype ( length ),  
field2_name datatype ( length ),  
field3_name datatype ( length ),  
....  
PRIMARY KEY ( field_name ),  
FOREIGN KEY ( field_name ) REFERENCES table_name,  
constraint [constraint_name] check (criteria)  
)
```

Drop Table

```
DROP TABLE table_name
```

Alter Table

```
ALTER TABLE table_name  
ADD PRIMARY KEY (column_name)
```



```
ALTER TABLE table_name
ADD CONSTRAINT constraint_name CHECK ()
```

```
ALTER TABLE table_name
ADD column_name column_type_data
```

```
ALTER TABLE table_name
DROP COLUMN column_name
```

```
ALTER TABLE table_name
ALTER COLUMN column_name new_data_type
```

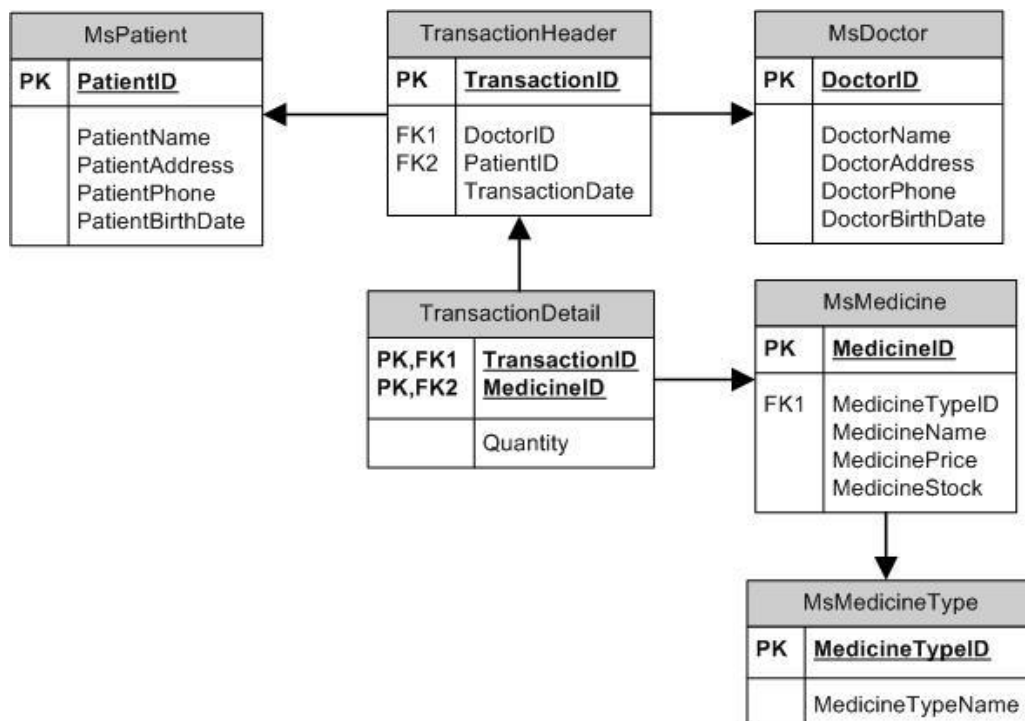
Create Index

```
CREATE INDEX "INDEX_NAME" ON "TABLE_NAME" (COLUMN_NAME)
```

Drop Index

```
DROP INDEX table_name, index_name
```

2. Relational Table



3. Assessment

- Buatlah database baru dengan nama 'TestDatabase', lalu gunakan database tersebut. (create database, use)

b. Buatlah tabel dengan struktur sebagai berikut:

Nama tabel : Msltem

Primary key : ItemID

Nama Field	Tipe Data	Length	Keterangan
ItemID	CHAR	5	Harus diisi.
ItemName	VARCHAR	50	Harus diisi.

(create table)

Msltem	
PK	<u>ItemID</u>
	ItemName

c. Buatlah tabel dengan struktur sebagai berikut:

Nama tabel : TransasctionItem

Nama Field	Tipe Data	Length	Keterangan
TransactionID	INT	-	Harus diisi.
ItemID	CHAR	5	<i>Foreign key.</i>
TransactionDate	DATE	-	Harus diisi.



d. Tambahkan primary key pada tabel TransactionItem pada kolom TransactionID.

(alter table, add primary key)

e. Tambahkan constraint pada tabel Msltem dimana panjang ItemName harus lebih dari 3 karakter, data yang terdahulu tidak perlu di cek.

(alter table, with no check, add constraint)

f. Tambahkan kolom pada tabel Msltem dengan struktur sebagai berikut:

Nama Field	Tipe Data	Length	Keterangan
ItemStock	INT	-	Harus diisi.
ItemPrice	NUMERIC	10,2	Harus diisi.

- (alter table, add column)
- g. Buatlah sebuah index dengan nama index1 untuk ItemID dan ItemName pada tabel MsItem.
(create index)
 - h. Hapuslah index dengan nama index1 yang telah dibuat.
(drop index)
 - i. Hapuslah tabel MsItem dan TransactionItem yang telah dibuat.
(drop table)
 - j. Hapuslah database dengan nama TestDatabase yang telah dibuat.
(drop database)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 7

MODULE 4

SQL DATA DEFINITION II



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Definition: View

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 7

LANGKAH-LANGKAH PRAKTIKUM

1. Syntax

Create View

```
CREATE VIEW view_name
```

```
[ ( column , ..... ) ]
```

As subquery

```
[ with check option ]
```

Drop View

```
DROP VIEW view_name
```

Create Procedure

```
CREATE PROCEDURE proc_name [(parameter_name parameter_type_data,...)]
```

```
[ ( column , ..... ) ]
```

As subquery

```
[ with check option ]
```

Execute Procedure

```
EXEC proc_name [(parameter_name parameter_type_data,...)]
```

Drop Procedure

```
DROP PROCEDURE proc_name
```

Create Function

```
CREATE FUNCTION function_name [(parameter_name parameter_type_data,...)]
```

```
RETURNS return_type_data
```

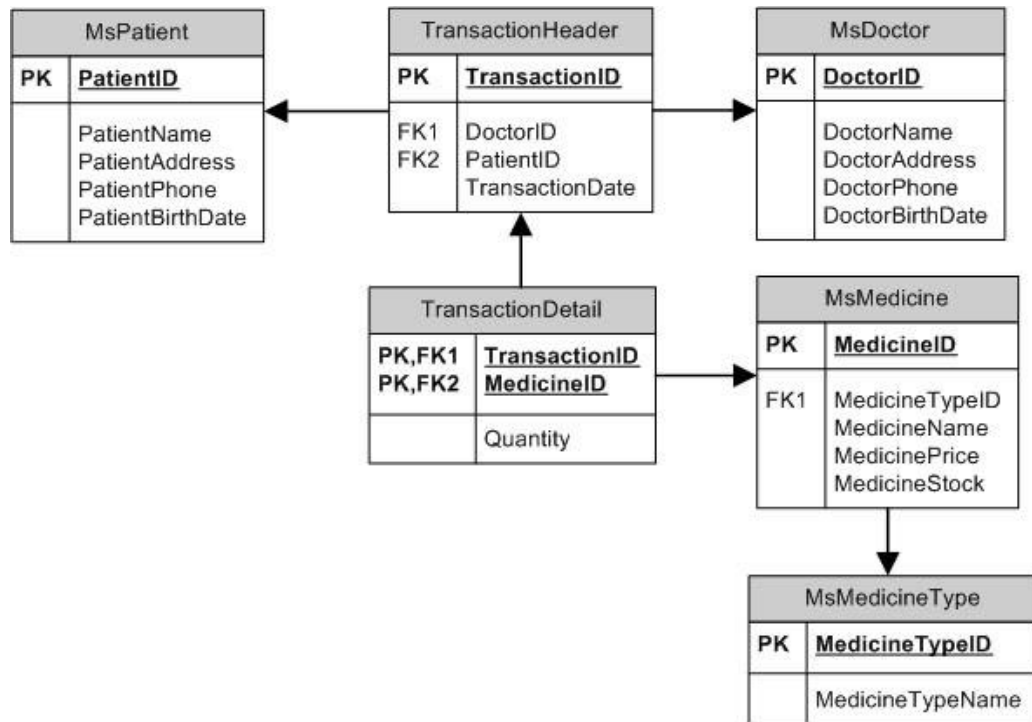
```
BEGIN
```

Query
RETURN return_data

END

Drop Function
DROP FUNCTION function_name

2. Relational Table



3. Assessment

- Buatlah sebuah view dengan nama 'view1' untuk menampilkan semua data dari tabel MsMedicine, lalu tampilkan view tersebut.

(create view)

	MedicineID	MedicineTypeID	MedicineName	MedicinePrice	MedicineStock
1	ME001	MT002	OBH Combi Plus	18000	50
2	ME002	MT001	Aspirin	5000	200
3	ME003	MT003	Balpirik	15500	30
4	ME004	MT003	Vicks	16000	30
5	ME005	MT001	Panadol	6000	450
6	ME006	MT002	Komix	2000	400
7	ME007	MT001	Clartin	14500	25
8	ME008	MT002	Decadryl	15000	25
9	ME009	MT002	Bisolvon	25000	30
10	ME010	MT002	Woods	18000	40

- b. Buatlah sebuah view dengan nama 'view2' untuk menampilkan MedicineName, MedicinePrice, MedicineStock, dan Total (didapat dari hasil perkalian MedicinePrice dan MedicineStock), lalu tampilkan view tersebut.

(create view)

	MedicineName	MedicinePrice	MedicineStock	Total
1	OBH Combi Plus	18000	50	900000
2	Aspirin	5000	200	1000000
3	Balpirik	15500	30	465000
4	Vicks	16000	30	480000
5	Panadol	6000	450	2700000
6	Komix	2000	400	800000
7	Clantin	14500	25	362500
8	Decadryl	15000	25	375000
9	Bisolvon	25000	30	750000
10	Woods	18000	40	720000

- c. Buatlah sebuah view dengan nama 'view3' untuk menampilkan TransactionDate (dengan format 'dd mon yyyy'), DoctorName, dan PatientName dimana dokter yang melayani bernama 'Mellisa Pratiwi', lalu tampilkan view tersebut.

(create view, convert)

	TransactionDate	DoctorName	PatientName
1	31 Jul 2012	Mellisa Pratiwi	Ffilia

- d. Buatlah sebuah view dengan nama 'view4' untuk menampilkan PatientName, QueueCode (didapat dari 3 digit terakhir PatientID) dimana QueueCode merupakan bilangan genap, lalu tampilkan view tersebut.

(create view, right)

	PatientName	QueueCode
1	Calvin	002
2	Falerian Jonathan	004
3	Hianto Hendry	006
4	Andi Juardi	008
5	Davin Timothy	010

- e. Buatlah sebuah view dengan nama 'view5' untuk menampilkan DoctorName, DoctorAddress (didapat dari DoctorAddress dengan mengganti kata 'Street' menjadi 'St.'), dan PatientName dimana PatientName hanya terdiri dari 1 kata, lalu tampilkan view tersebut.

(create view, replace, not like)

	DoctorName	DoctorAddress	PatientName
1	Nico Natalie	Anggrek St.	Ffilia
2	Franky Quo	Crox St.	Junius
3	Yudy Pumama	Brother St.	Ffilia
4	Yudy Pumama	Brother St.	Calvin
5	Mellisa Pratiwi	Bowl St.	Ffilia

- f. Buatlah sebuah view dengan nama 'view6' untuk menampilkan DoctorID dan DoctorName dimana dokter tersebut pernah melayani pasien pada hari 'Kamis', jika data yang ditampilkan berulang cukup tampilkan sekali saja, lalu tampilkan view tersebut.
 (create view, distinct, datename, weekday)

	DoctorID	DoctorName
1	DC003	Julian Wesley
2	DC005	Indra Saswita

- g. Buatlah sebuah user-defined function dengan nama 'Multiply' yang menerima parameter number1 dan number2 dengan tipe data int yang berfungsi untuk mengembalikan hasil perkalian dari number1 dan number2.
 (create function)

SELECT dbo.Multiply(5,4)	
	(No column name)
1	20

- h. Buatlah sebuah stored-procedure dengan nama 'proc1' untuk menampilkan semua data dari tabel MsPatient, lalu jalankan stored-procedure tersebut. (create procedure)

	PatientID	PatientName	PatientAddress	PatientPhone	PatientBirthDate
1	PA001	Ffilia	Hard Stone Street	08991828752	1991-12-24
2	PA002	Calvin	Syahdan Street	08191231589	2003-12-23
3	PA003	Ferlim Putra	Kapuk Street	08991200912	2001-08-02
4	PA004	Falerian Jonathan	Sunter Street	021921125252	1974-05-30
5	PA005	Gradiyanto Sanjaya	Greenville Street	02198776654	1982-07-31
6	PA006	Hianto Hendry	Syahdan Street	081823358796	1987-04-01
7	PA007	Junius	Kijang Street	08812670794	1998-03-03
8	PA008	Andi Juardi	Google Street	0219783968	2005-11-30
9	PA009	Jeklin Harefa	Web Street	0818231322	1976-02-25
10	PA010	Davin Timothy	Strong Horse St...	0219780642	1955-12-15

- i. Buatlah sebuah stored-procedure dengan nama 'proc2' yang memiliki parameter MedicineID untuk menampilkan TransactionID dan TotalSales (didapat dari hasil perkalian antara Quantity dan MedicinePrice dengan menggunakan function yang telah dibuat pada nomor 7), lalu jalankan stored-procedure tersebut.
 (create procedure)

EXEC proc2 'ME009'		
	TransactionID	TotalSales
1	TR015	50000

- j. Hapuslah semua view, function, dan stored-procedure yang telah dibuat.
(drop view, drop function, drop procedure)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-29206118-4, Chapter 7

MODULE 5

SECURITY AND ADMINISTRATION

DESKRIPSI TEMA

Apply user role and right access to increase security in database system

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to Security and Administration: Backup & Restore Database, Users Administrations & Roles, Data Transformation Services, Grant & Revoke

PENUNJANG PRAKTIKUM

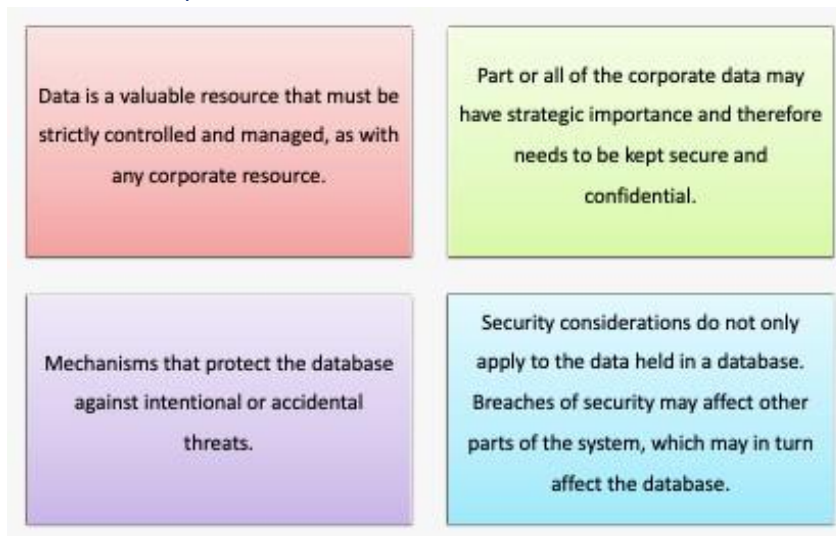
1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 20

LANGKAH-LANGKAH PRAKTIKUM

1. Database Security

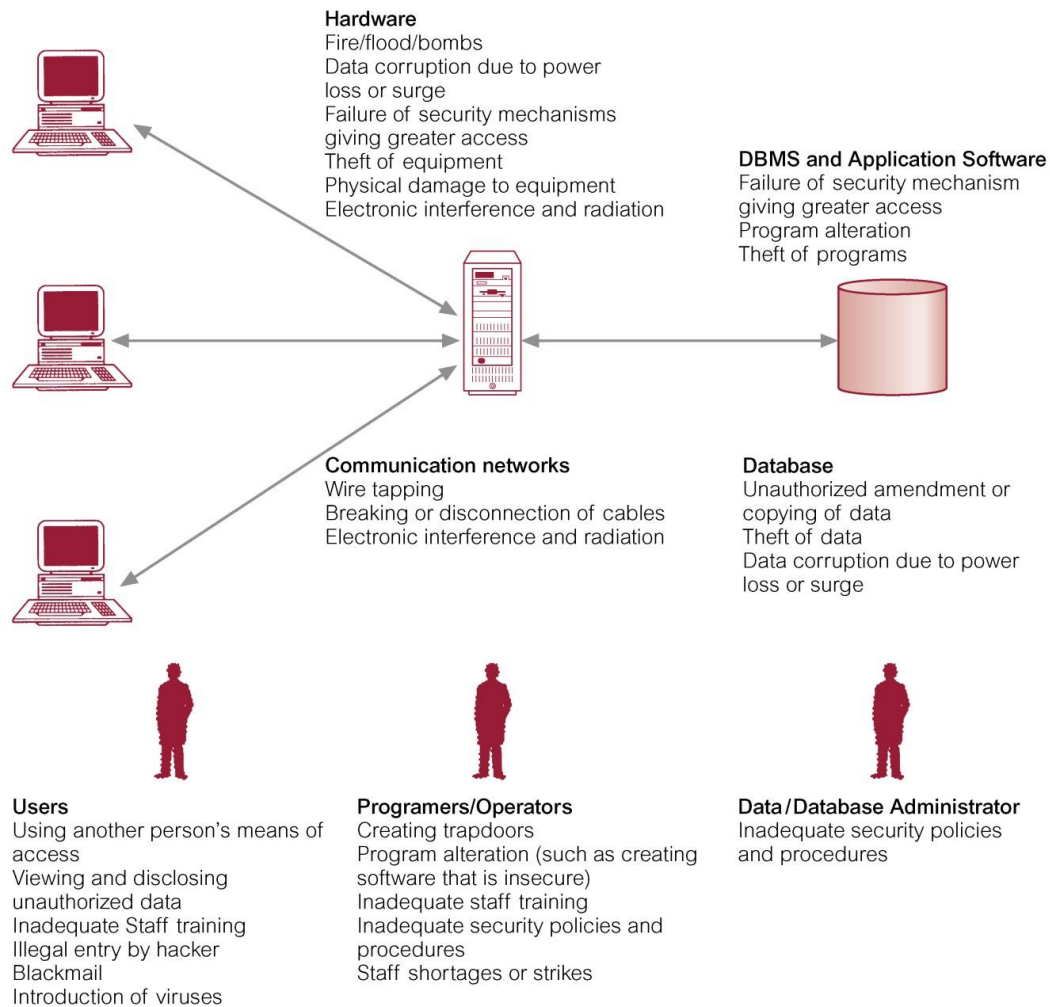
Database Security is the mechanisms that protect the database against intentional or accidental threats Involves measures to avoid:

- a. Theft and fraud
- b. Loss of confidentiality (secrecy): refers to the need to maintain secrecy of data (critical data)
- c. Loss of privacy: refers to the need to protect data about individuals
- d. Loss of integrity: invalid or corrupted data
- e. Loss of availability: cannot be accessed

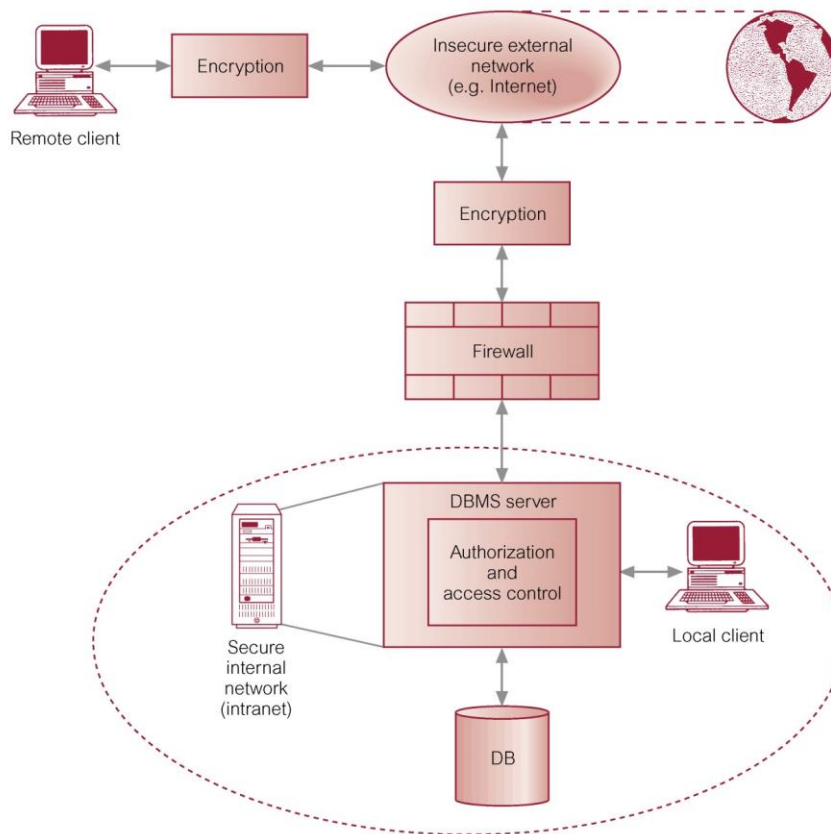


2. Threat

Any situation or event, whether intentional or unintentional, that will adversely affect a system and consequently an organization.



3. Protecting a Computer System



Countermeasures – Computer-Based Controls

Concerned with physical controls to administrative procedures and includes:

1. Authorization
2. Access controls
3. Views
4. Backup and recovery
5. Integrity
6. Encryption
7. RAID technology

4. Access Control: GRANT and REVOKE Command

- Access Control - Authorization Identifiers and Ownership

1. D Authorization identifier is normal SQL identifier used to establish identity of a user. Usually has an associated password.
2. Used to determine which objects user may reference and what operations may be performed on those objects.
3. Each object created in SQL has an owner, as defined in AUTHORIZATION clause of schema to which object belongs.
4. Owner is only person who may know about it.

- Actions user permitted to carry out on given base table or view:

1. **SELECT** Retrieve data from a table.

- | | | |
|----|-------------------|--|
| 2. | INSERT | Insert new rows into a table. |
| 3. | UPDATE | Modify rows of data in a table. |
| 4. | DELETE | Delete rows of data from a table. |
| 5. | REFERENCES | Reference columns of named table in integrity constraints. |
| 6. | USAGE | Use domains, collations, character sets, and translations. |

- Can restrict INSERT/UPDATE/REFERENCES to named columns.
- Owner of table must grant other users the necessary privileges using GRANT statement.
- To create view, user must have SELECT privilege on all tables that make up view and REFERENCES privilege on the named columns.

5. Grant

```
GRANT {PrivilegeList | ALL PRIVILEGES}  
ON ObjectName  
TO {AuthorizationIdList | PUBLIC}  
[WITH GRANT OPTION]
```

- PrivilegeList consists of one or more of above privileges separated by commas.
- ALL PRIVILEGES grants all privileges to a user.
- PUBLIC allows access to be granted to all present and future authorized users.
- ObjectName can be a base table, view, domain, character set, collation or translation.
- WITH GRANT OPTION allows privileges to be passed on.

6. Grant Examples

- Give Manager full privileges to Staff table.
GRANT ALL PRIVILEGES
ON Staff
TO Manager WITH GRANT OPTION;
- Give users Personnel and Director SELECT and UPDATE on column salary of Staff.
GRANT SELECT, UPDATE (salary)
ON Staff
TO Personnel, Director;
- Give all users SELECT on Branch table.
GRANT SELECT
ON Branch
TO PUBLIC;

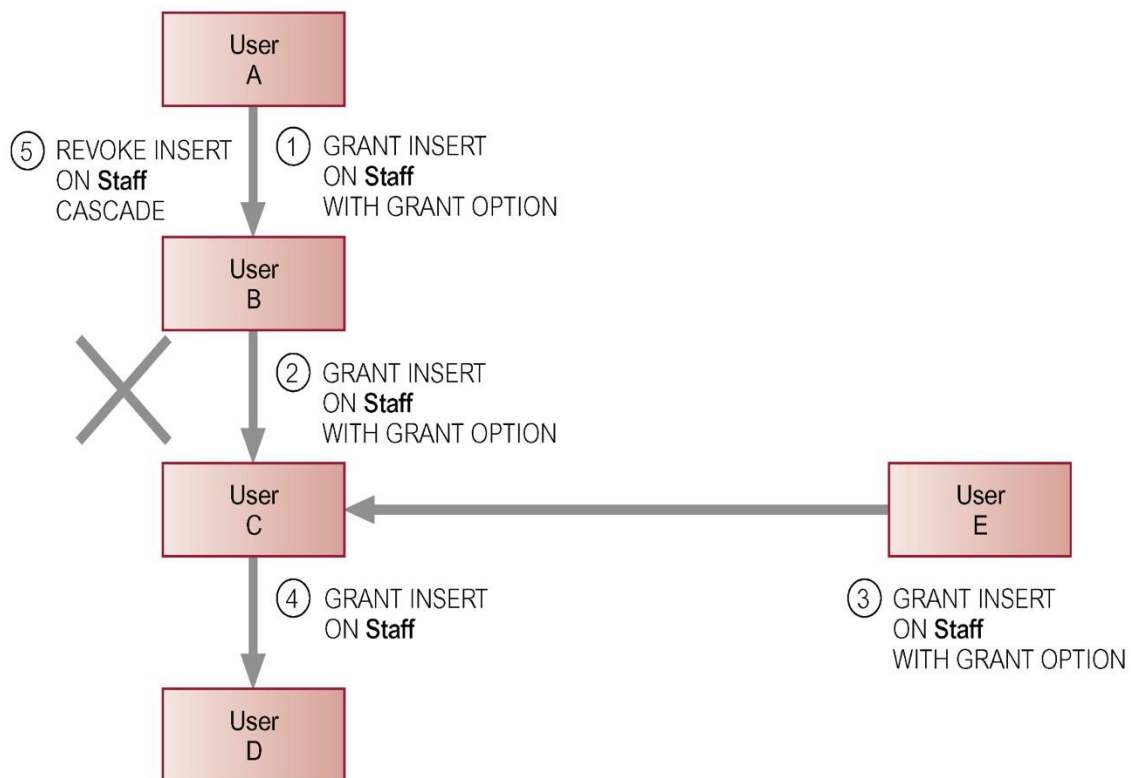
7. Revoke

REVOKE takes away privileges granted with GRANT.

```

REVOKE [GRANT OPTION FOR]
      {PrivilegeList | ALL PRIVILEGES}
ON ObjectName
FROM {AuthorizationIdList | PUBLIC}
      [RESTRICT | CASCADE]
  
```

- ALL PRIVILEGES refers to all privileges granted to a user by user revoking privileges.
- GRANT OPTION FOR allows privileges passed on via WITH GRANT OPTION of GRANT to be revoked separately from the privileges themselves.
- REVOKE fails if it results in an abandoned object, such as a view, unless the CASCADE keyword has been specified.
- Privileges granted to this user by other users are not affected.



- Revoke privilege SELECT on Branch table from all users.

```

REVOKE SELECT
ON Branch
FROM PUBLIC;
  
```

- Revoke all privileges given to Director on Staff table.

```

REVOKE ALL PRIVILEGES
ON Staff
  
```

8. Sintaks GRANT

```
GRANT { ALL [ PRIVILEGES ] }  
    | permission [ ( column [ ,...n ] ) ] [ ,...n ]  
    [ ON [ class :: ] securable ] TO principal [ ,...n ]  
    [ WITH GRANT OPTION ] [ AS principal ]
```

9. REVOKE

```
REVOKE { ALL [ PRIVILEGES ] }  
    | permission [ ( column [ ,...n ] ) ] [ ,...n ]  
    [ ON [ class :: ] securable ] FROM principal [ ,...n ]  
    [ CASCADE ] [ AS principal ]
```

10. Assessment

- a. Lakukan backup database 'dbPracCase' secara full backup ke Drive D:\ dengan nama file 'ContohBackUp1'.
- b. Lakukan restore database dari file 'ContohBackUp1' di Drive D:\ menjadi database baru bernama 'praktikum'.
- c. Buatlah sebuah user bernama 'user1' dengan password 'user1' dan server roles yang dimiliki adalah 'public, sysadmin, serveradmin, setupadmin'.
- d. Buatlah sebuah user bernama 'user2' dengan password 'user2' dengan default database 'dbPracCase' dan user mapping ke database 'dbPracCase' dan 'master'.
- e. Berilah hak akses select kepada user2 terhadap tabel MsPatient.
(grant)
- f. Berilah hak akses insert, update kepada user2 terhadap tabel TransactionDetail.
(grant)
- g. Berilah semua hak akses kepada user2 terhadap tabel MsDoctor.
(grant)
- h. Cabutlah hak akses select pada tabel MsPatient dari user2.
(revoke)
- i. Cabutlah hak akses insert, update pada tabel TransactionDetail dari user2.
(revoke)
- j. Cabutlah semua hak akses pada tabel MsDoctor dari user2.
(revoke)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 20

MODULE 6

DATA MANIPULATION I



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Manipulation: Simple SQL command, Sorting the query results, Date Functions, Character Functions, Operator <, >, =, %, /

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

LANGKAH-LANGKAH PRAKTIKUM

a. Insert, Update, Delete

b. Syntax

Insert

```
INSERT INTO table_name ( field1_name, field2_name, ...) VALUES ( value1, value2, ...)
```

Update

```
UPDATE table_name
```

```
SET field1_name = [new value]
```

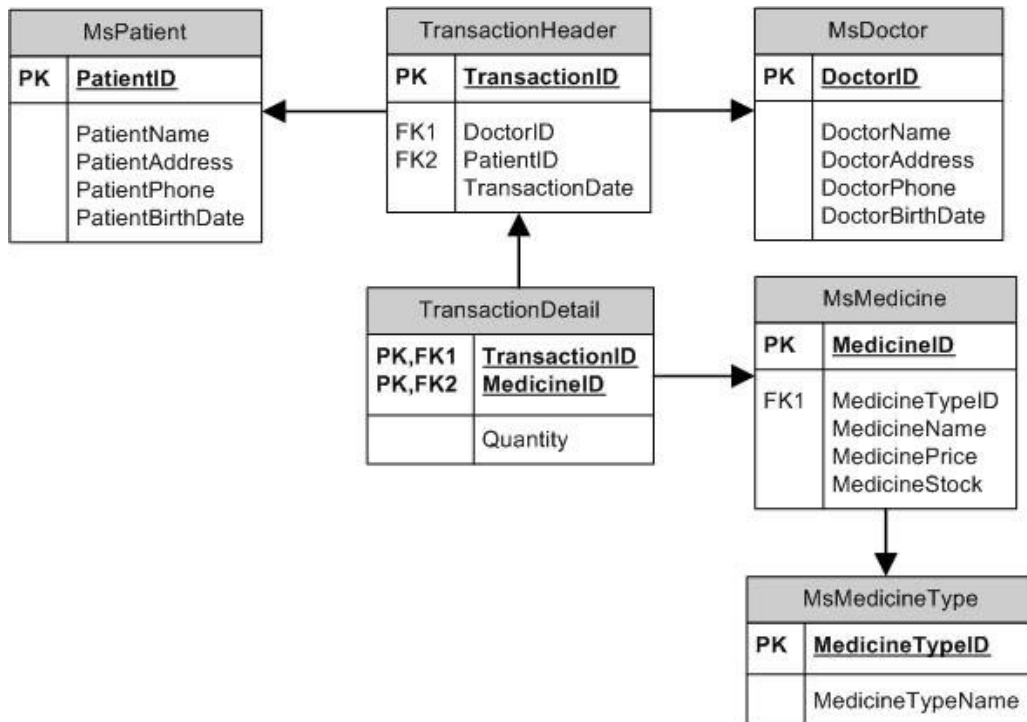
```
WHERE {condition}
```

Delete

```
DELETE FROM table_name
```

```
WHERE {condition}
```

c. Relational Table



d. Assessment

- a. Masukkan data sebagai berikut ke dalam tabel MsPatient.
(insert)

PatientID	PatientName	PatientAddress	PatientPhone	PatientBirthDate
PA011	Ernalina Dewi	Delima Street	021-987987	1954-12-31
PA012	Danny Sadeli	Mandiri Street	021-1234565	1992-04-06
PA013	Indra Susanto	Dermawan Street	021-666554	1988-08-08
PA014	Budianto	Rama Sinta Street	021-554466	1975-04-04
PA015	Flavius	Tugas Mandiri Street	021-132456	2010-05-21

- b. Masukkan data sebagai berikut ke dalam tabel MsDoctor.
(insert)

DoctorID	DoctorName	DoctorAddress	DoctorPhone	DoctorBirthDate
DC011	Calvin	Jelambar Street	021-98212354	1969-12-31
DC012	Jamal	Muara Karang Street	021-11231299	1962-03-20
DC013	Ade Galang	Pluit Indah Street	021-66212588	2000-02-02
DC014	Kevin Pangesa	Muara Karang Street	021-9152678	1945-08-17

DCo15	Imelda	Pantai Mutiara Street	021-554433322	1954-05-04
-------	--------	-----------------------	---------------	------------

- c. Masukkan data sebagai berikut ke dalam tabel MsMedicine.
(insert)

MedicineID	MedicineTypeID	MedicineName	MedicinePrice	MedicineStock
MEo11	MT002	Vitamin C	50000	45
MEo12	MT001	Vitamin D	55000	50

- d. Masukkan data sebagai berikut ke dalam tabel MsMedicineType. (insert)

MedicineTypeID	MedicineTypeName
MT004	Drops

- e. Lakukan *transaction* terlebih dahulu, ubahlah MedicineStock pada tabel MsMedicine menjadi 3, lalu kembalikan isi dari MedicineStock yang telah diubah menjadi semula. (begin tran, update, rollback)

	MedicineID	MedicineTypeID	MedicineName	MedicinePrice	MedicineStock
1	ME001	MT002	OBH Combi Plus	18000	3
2	ME002	MT001	Aspirin	5000	3
3	ME003	MT003	Balpink	15500	3
4	ME004	MT003	Vicks	16000	3
5	ME005	MT001	Panadol	6000	3
6	ME006	MT002	Komix	2000	3
7	ME007	MT001	Claritin	14500	3
8	ME008	MT002	Decadryl	15000	3
9	ME009	MT002	Bisolvon	25000	3
10	ME010	MT002	Woods	18000	3
11	ME011	MT002	Vitamin C	50000	3
12	ME012	MT001	Vitamin D	55000	3

- f. Lakukan *transaction* terlebih dahulu, ubahlah DoctorPhone pada tabel MsDoctor menjadi '0123456' **dimana** DoctorID adalah 'DC003', lalu kembalikan isi dari DoctorPhone yang telah diubah menjadi semula.

(begin tran, update, rollback)

	DoctorID	DoctorName	DoctorAddress	DoctorPhone	DoctorBirthDate
1	DC001	Yogie Soesanto	Binus Street	+62216600123	1985-11-24
2	DC002	Nico Natalie	Anggrek Street	+628181231345	1992-12-15
3	DC003	Julian Wesley	Syahdan Street	021-123456	1964-03-07
4	DC004	Mellisa Pratiwi	Bowl Street	+6221112331	1994-08-08
5	DC005	Indra Saswita	Sunter Street	+62216600127	1962-09-23
6	DC006	Franky Quo	Crox Street	+62816609123	1975-10-01
7	DC007	Livia Ashianti	Syahdan Street	+62816699111	1989-04-03
8	DC008	Yudy Pumama	Brother Street	+6221316623	1965-05-22
9	DC009	Budiman	Kijang Street	+6289991221	1954-02-02
10	DC010	Andy Putra	Angry Street Street	+6287877991	1987-12-06
11	DC011	Calvin	Jelambar Street	021-98212354	1969-12-31
12	DC012	Jamal	Muara Karang Street	021-11231299	1962-03-20
13	DC013	Ade Galang	Pluit Indah Street	021-66212588	2000-02-02
14	DC014	Kevin Pangesa	Muara Karang Street	021-9152678	1945-08-17
15	DC015	Imelda	Pantai Mutiara Street	021-554433322	1954-05-04

- g. Lakukan *transaction* terlebih dahulu, ubahlah MedicineStock pada tabel MsMedicine menjadi bertambah 100 **dimana** obat tersebut pernah terbeli pada transaksi dengan TransactionID = 'TR002', lalu kembalikan isi dari MedicineStock yang telah diubah menjadi semula.

(begin tran, update, rollback)

	MedicineID	MedicineTypeID	MedicineName	MedicinePrice	MedicineStock
1	ME001	MT002	OBH Combi Plus	18000	150
2	ME002	MT001	Aspirin	5000	300
3	ME003	MT003	Balpirik	15500	30
4	ME004	MT003	Vicks	16000	30
5	ME005	MT001	Panadol	6000	550
6	ME006	MT002	Komix	2000	500
7	ME007	MT001	Clantin	14500	25
8	ME008	MT002	Decadryl	15000	25
9	ME009	MT002	Bisolvon	25000	30
10	ME010	MT002	Woods	18000	40
11	ME011	MT002	Vitamin C	50000	45
12	ME012	MT001	Vitamin D	55000	50

- h. Lakukan *transaction* terlebih dahulu, hapuslah semua data pada tabel MsPatient, lalu kembalikan isi tabel MsPatient menjadi semula. (**begin tran, delete, rollback**) Data yang terhapus:

	PatientID	PatientName	PatientAddress	PatientPhone	PatientBirthDate
1	PA001	Fifilia	Hard Stone Street	08991828752	1991-12-24
2	PA002	Calvin	Syahdan Street	08191231589	2003-12-23
3	PA003	Ferlim Putra	Kapuk Street	08991200912	2001-08-02
4	PA004	Falerian Jonathan	Sunter Street	021921125252	1974-05-30
5	PA005	Gradiyanto Sanjaya	Greenville Street	02198776654	1982-07-31
6	PA006	Hianto Hendry	Syahdan Street	081823358796	1987-04-01
7	PA007	Junius	Kjang Street	08812670794	1998-03-03
8	PA008	Andi Juardi	Google Street	0219783968	2005-11-30
9	PA009	Jeklin Harefa	Web Street	0818231322	1976-02-25
10	PA010	Davin Timothy	Strong Horse Street	0219780642	1955-12-15
11	PA011	Emalia Dewi	Delima Street	021-987987	1954-12-31
12	PA012	Danny Sadeli	Mandiri Street	021-1234565	1992-04-06
13	PA013	Indra Susanto	Demawan Street	021-666554	1988-08-08
14	PA014	Budianto	Rama Sinta Street	021-554466	1975-04-04
15	PA015	Flavius	Tugas Mandiri Street	021-132456	2010-05-21

- i. Lakukan *transaction* terlebih dahulu, hapuslah data pada tabel MsPatient dimana pasien tersebut bernama adalah 'Flavius', lalu kembalikan isi tabel MsPatient menjadi semula. (**begin tran, delete, rollback**) Data yang terhapus:

	PatientID	PatientName	PatientAddress	PatientPhone	PatientBirthDate
1	PA015	Flavius	Tugas Mandiri Street	021-132456	2010-05-21

- j. Lakukan *transaction* terlebih dahulu, hapuslah data pada tabel MsDoctor dimana dokter tersebut pernah melayani pemeriksaan pada tanggal '2011-05-01', lalu kembalikan isi tabel MsDoctor menjadi semula. (**begin tran, delete, rollback**) Data yang terhapus:

	DoctorID	DoctorName	DoctorAddress	DoctorPhone	DoctorBirthDate
1	DC010	Andy Putra	Angry Street Street	+6287877991	1987-12-06

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-29206118-4, Chapter 6

MODULE 7

CASE STUDY 1



DESKRIPSI TEMA

(Apa yang akan dipelajari mahasiswa pada minggu ini)

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

(Capaian pembelajaran pada minggu ini)

PENUNJANG PRAKTIKUM

1. ...
 2. ...
 3. dst
- (+ Perlengkapan Apd/Alat Pelindung Diri Yang Harus Digunakan, Jika Ada)

LANGKAH-LANGKAH PRAKTIKUM

Studi Kasus Terlampir

REFERENSI

MODULE 8

DATA MANIPULATION II



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Manipulation: Simple SQL command, Sorting the query results, Date Functions, Character Functions, Operator <, >, =, %, /

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

LANGKAH-LANGKAH PRAKTIKUM

1. Select, Simple Queries, Sorting

2. Syntax Select

```
SELECT "field_name" FROM table_name
```

Where

```
SELECT "field_name"
```

```
FROM table_name
```

```
WHERE {condition}
```

Between

```
SELECT "field_name"
```

```
FROM table_name
```

```
WHERE "field_name" BETWEEN 'value1' AND 'value2'
```

Like

```
SELECT "field_name"
```

```
FROM table_name
```

```
WHERE "field_name" LIKE {PATTERN}
```

Order by

```
SELECT "field_name"
```

```
FROM table_name WHERE {condition} ORDER BY "field_name" [ASC | DESC]
```

3. Relational Table

4. Assessment

- a. Tampilkan DoctorID, DoctorName, dan DoctorPhone dari tabel MsDoctor.
- b. Tampilkan MedicineID, MedicineName, dan MedicinePrice dimana MedicineStock lebih besar dari 100.
- c. Tampilkan PatientName dan PatientAddress dimana tahun kelahiran pasien lebih besar dari 2000.
(year)
- d. Tampilkan PatientID, PatientName, dan PatientPhone dimana 3 karakter pertama PatientPhone adalah '021'.
(left)
- e. Tampilkan TransactionID dan TransactionDate dimana bulan dari TransactionDate di antara 5 dan 7.
(month, between)
- f. Tampilkan DoctorID, DoctorName, dan DoctorAddress dimana huruf pertama DoctorName merupakan huruf vokal (a,i,u,e,o).
(like)
- g. Tampilkan PatientNumber (didapat dari PatientID dengan mengubah 'PA' menjadi 'Patient #'), PatientName, dan PatientAddress dimana PatientAddress terdiri dari 3 kata atau lebih.
(replace, like)
- h. Tampilkan TransactionID dan TransactionDate dimana 3 digit terakhir dari DoctorID yang melayani merupakan bilangan genap. Urutkan data-data tersebut berdasarkan TransactionDate secara ascending.
(right, order by).
- i. Tampilkan PatientID, PatientName, PatientBirthDate, dan TransactionDate dimana pasien tersebut memiliki umur di bawah 17 tahun saat melakukan transaksi. Urutkan data-data tersebut berdasarkan PatientBirthDate dari pasien yang termuda. (datediff, order by)
- j. Tampilkan MedicineID, MedicineName, TypeName (dengan format uppercase), dan MedicineStock (dengan menambahkan kata ' pcs' di akhir) dimana MedicineStock merupakan bilangan kelipatan 25 dan panjang karakter MedicineName kurang dari 10 karakter. Urutkan data-data tersebut berdasarkan MedicineName dari huruf terkecil.
(upper, cast, len, order by)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-29206118-4, Chapter 6

MODULE 9

DATA MANIPULATION III



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Manipulation: Simple SQL command, Sorting the query results, Date Functions, Character Functions, Operator <, >, =, %, /

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

LANGKAH-LANGKAH PRAKTIKUM

1. SQL Aggregate Functions

Syntax

Berikut ini adalah beberapa fungsi aggregate:

1. Sum (nama_field) = menjumlahkan isi baris-baris pada field tersebut.
2. Count (nama_field) = untuk menghitung banyaknya baris pencacahan data.
3. Avg (nama_field) = untuk menghitung rata-rata dari isi baris-baris data.
4. Max (nama_field) = untuk menghitung nilai max dari baris-baris data.
5. Min (nama_field) = untuk menghitung nilai min dari baris-baris data.

Tambahan *syntax*:

1. Order by

Pengurutan/sorting dilakukan dengan sintak order by. Order by digunakan untuk mengurutkan hasil-set dengan sebuah kolom yang ditetapkan. ORDER BY mengurutkan hasil dalam urutan menaik (ascending) secara default. Jika Anda ingin mengurutkan hasil yang menurun (besar ke kecil) dalam urutan, Anda dapat menggunakan kata kunci DESC.

Syntax:

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name(s) ASC|DESC
```


2. **Group by**

GROUP BY digunakan untuk pengelompokan data, biasa digunakan bersama dengan fungsi SQL agregat seperti SUM untuk menyediakan sarana pengelompokan dataset hasilnya dengan tabel database kolom tertentu. Jika suatu query mengandung proses perhitungan/kalkulasi dengan menggunakan aggregate, maka field-field lain yang tidak termasuk dalam aggregate harus dimasukkan dalam group by untuk mengelompokkan hasil kalkulasi.

Syntax:

```
SELECT column_name, aggregate_function(column_name) FROM table_name  
WHERE column_name operator value  
GROUP BY column_name
```

3. **Having**

Untuk membuat syarat bagi fungsi *aggregate* yang kita pakai.

4. **Datetime**

Untuk mendapatkan nama dari bagian tanggal tertentu Bentuk umum:

Datetime(SatuanTanggal,Tanggal)

Contoh

datetime(month, '10/5/10') akan mengembalikan October

5. **Like**

Mengembalikan nilai TRUE, jika operand yang diuji sesuai dengan pattern yang ditentukan.

6. **Not Like**

Membalikkan nilai Boolean dari operator-operator logika.

7. **Year**

Untuk mengembalikan data tahun

Bentuk umum: year (Tanggal)

Contoh

Year (Tanggal) dan tanggal hari ini adalah 4 January 2011, maka akan mengembalikan nilai 2011.

8. **Len**

Digunakan untuk mencari panjang kata atau kalimat.

Bentuk umum: len(namaField)

Contoh

Len(nama) dan jika nama adalah "Edbertius", maka akan menghasilkan nilai 9.

9. Convert

Digunakan untuk mengubah tipe data suatu nilai ke tipe data lainnya dan melakukan formatting untuk data berupa tanggal. Bentuk umum: `Convert(tipeDataTujuan, namaField [, jenisStyle])`

Contoh

`Convert(char, tgl, 106)` dan jika tgl adalah "4 January 2011", maka tgl yang dihasilkan adalah dengan format "dd MMM YYYY", 4 Jan 2011

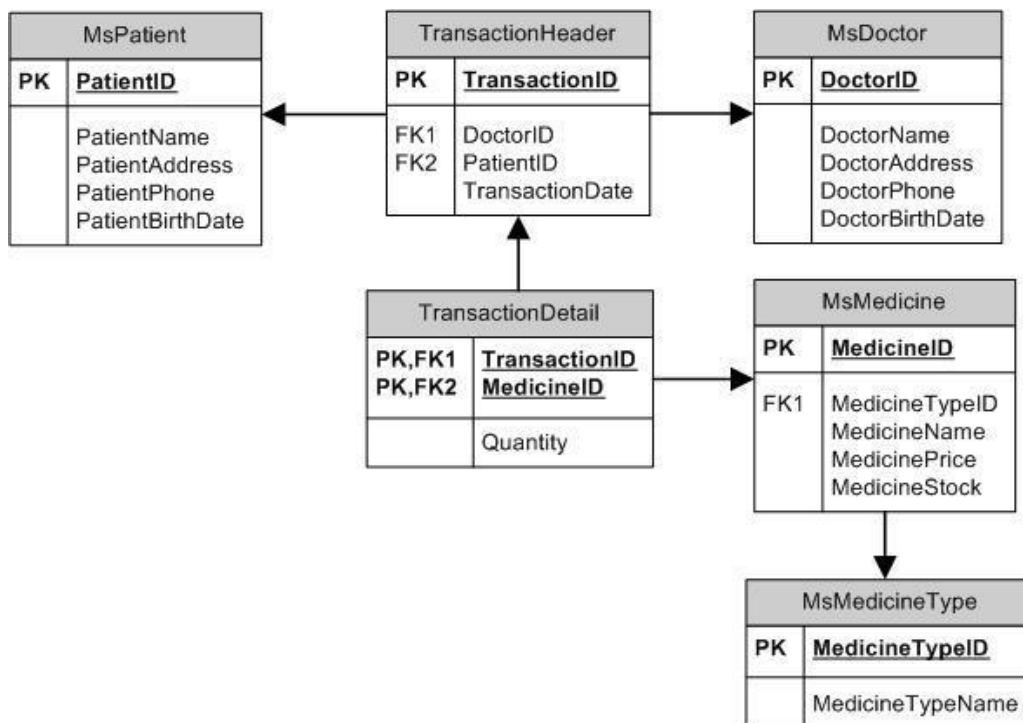
10. Between

Mengembalikan nilai TRUE, jika operand yang diuji berada dalam jangkauan.

Urutan penggunaan :

- group by
- having (untuk menggunakan having, harus menggunakan group by)
- order by

11. Relational Table



12. Assessment

- a. Tampilkan MaxQuantity (didapat dari Quantity terbanyak obat yang terjual), MinQuantity (didapat dari Quantity tersedikit obat yang terjual), dan AvgQuantity (didapat dari rata-rata Quantity obat yang terjual).

(max, min, avg)

	MaxQuantity	MinQuantity	AvgQuantity
1	55	1	7

- b. Tampilkan MedicineTypeID, MedicineTypeName, dan MedicineCount (didapat dari banyaknya MedicineID pada setiap MedicineType).

(count, group by)

	MedicineTypeID	MedicineTypeName	MedicineCount
1	MT001	Tablet	3
2	MT002	Syrup	5
3	MT003	Balm	2

- c. Tampilkan DoctorID, DoctorName, dan TransactionCount (didapat dari banyaknya pemeriksaan yang dilayani oleh dokter tersebut) **dimana** pemeriksaan tersebut dilakukan sebelum bulan ke-7.

(count, month, group by)

	DoctorID	DoctorName	TransactionCount
1	DC001	Yogie Soesanto	2
2	DC005	Indra Saswita	2
3	DC008	Yudy Pumama	1
4	DC010	Andy Putra	1

- d. Tampilkan PatientID, PatientName, dan TotalMedicineBought (didapat dari jumlah Quantity obat yang dibeli) **dimana** MedicineID dari obat yang dibeli adalah 'ME008' dan PatientName terdiri dari 2 kata atau lebih.

(sum, like, group by)

	PatientID	PatientName	TotalMedicineBought
1	PA006	Hianto Hendry	5
2	PA009	Jeklin Harefa	7

- e. Tampilkan Year (didapat dari tahun TransactionDate) dan MaxQuantity (didapat dari Quantity terbanyak obat yang terjual) **dimana** transaksi ditangani oleh dokter dengan DoctorID = 'DC001'.

(year, max, group by)

	Year	MaxQuantity
1	2008	55
2	2010	12

- f. Tampilkan PatientID, PatientName, dan TransactionCount (didapat dari banyaknya pemeriksaan yang dilakukan oleh pasien tersebut) **dimana** 3 karakter pertama PatientPhone bukan merupakan '021' dan PatientAddress tidak terdiri dari 3 kata atau lebih.

(count, left, not like, group by)

	PatientID	PatientName	TransactionCount
1	PA002	Calvin	1
2	PA003	Ferlim Putra	2
3	PA006	Hianto Hendry	1
4	PA007	Junius	1
5	PA009	Jeklin Harefa	1

- g. Tampilkan DoctorID, DoctorName, DoctorBirthday (didapat dari DoctorBirthDate dengan format 'dd mon yyyy'), dan TotalMedicineSold (didapat dari jumlah Quantity obat yang terjual) **dimana** panjang DoctorName lebih besar dari 14 karakter.

(convert, sum, len, group by)

	DoctorID	DoctorName	DoctorBirthday	TotalMedicineSold
1	DC004	Mellisa Pratiwi	08 Aug 1994	9

- h. Tampilkan MedicineID, MedicineName, dan TotalProfit (didapat dari jumlah perkalian antara MedicinePrice dan Quantity) **dimana** tanggal saat obat tersebut dibeli di antara 10 dan 20.

(sum, day, between, group by)

	MedicineID	MedicineName	TotalProfit
1	ME002	Aspirin	5000
2	ME008	Decadryl	105000
3	ME010	Woods	90000

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

MODULE 10

DATA MANIPULATION IV



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Manipulation: Simple SQL command, Sorting the query results, Date Functions, Character Functions, Operator <, >, =, %, /

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6.

LANGKAH-LANGKAH PRAKTIKUM

1. Subqueries & Multi Table Queries

2. Syntax

Alias subquery

Alias subquery adalah subquery yang diberi nama, alias subquery dianggap sebagai tabel virtual sehingga dapat digunakan untuk keperluan-keperluan yang lain.

Contoh :

Tampilkan semua NamaKaryawan dan NamaPelanggan dengan cara cross product.

Jawab:

```
SELECT t1.NamaKaryawan, t2.NamaPelanggan
FROM (SELECT * FROM MsKaryawan) as t1, (SELECT * FROM MsPelanggan) as t2
```

IN

Sintaks in digunakan jika kita sudah mengetahui nilai yang mau dikembalikan dari satu kolom dari tabel yang berbeda. IN akan mengembalikan 1 kolom, jadi pertama - tama kita harus sebutkan dulu foreignKey yang menghubungkan tabel1 dan tabel2, dan didalam subquery-nya, yang kita select HANYA 1 kolom saja, sehingga foreignKey dari table1 akan disamakan dengan foreignKey dari table2. penggunaan in tidak hanya untuk menghubungkan tabel - tabel saja, namun juga bisa digunakan untuk mengambil nilai tertentu.

Sintaks:

```
SELECT column_name FROM table_name
```

```
WHERE column_name <primary key> in ( SELECT column_name FROM table_name)
```

Contoh:

Tampilkan KodeKaryawan dan NamaKaryawan dimana KodeKaryawan berakhiran 2, 3 atau 5 (Gunakan sintak in).

Jawab:

```
SELECT KodeKaryawan, NamaKaryawan
FROM MsKaryawan
WHERE RIGHT(KodeKaryawan,1) IN (2,3,5)
```

Exists

Sintak Exists mirip dengan in bedanya pada pengembalian nilai, sintak exists tidak mengembalikan data-data yang di SELECT dalam subquery, namun mengembalikan nilai boolean, yaitu true atau false. Nilai true dikembalikan jika subquery dalam exists mengandung minimal satu baris data, sebaliknya nilai false dikembalikan jika subquery dalam exists tidak menghasilkan baris data sama sekali.

Sintaks:

```
SELECT column_name FROM table_name
WHERE exists (SELECT * FROM table_name WHERE column_name = column_name)
```

Contoh:

Tampilkan NamaGame dan Harga dimana panjang kata dari JenisGame lebih dari 3 karakter (gunakan sintak exists).

Jawab:

```
SELECT NamaGame, Harga
FROM MsGame mg
WHERE EXISTS (
    SELECT 1
    FROM MsJenisGame mjpg
    WHERE mjpg.KodeJenisGame=mg.KodeJenisGame
    AND LEN(JenisGame)>3
)
```

Alias

Untuk field name

```
SELECT "field_name" AS "field_alias" FROM "table_name"
```

Untuk table name

```
SELECT "field_name" FROM "table_name" AS "table_alias"
```

In SELECT "field_name" FROM "table_name" WHERE "field_name" IN (value1, value2, ..)

Exists SELECT "field_name" FROM "table_name" WHERE EXISTS (value1, value2, ..)

3. Syntax (+) Cast

Fungsi

Digunakan untuk mengubah tipe data suatu nilai ke tipe data lainnya.

Bentuk umum: Cast(namaField AS tipeDataTujuan)

Contoh

„Rp. "+Cast(harga AS char)+",00" dan jika harga adalah 5000, maka hasilnya adalah "Rp. 5000,00"

Upper

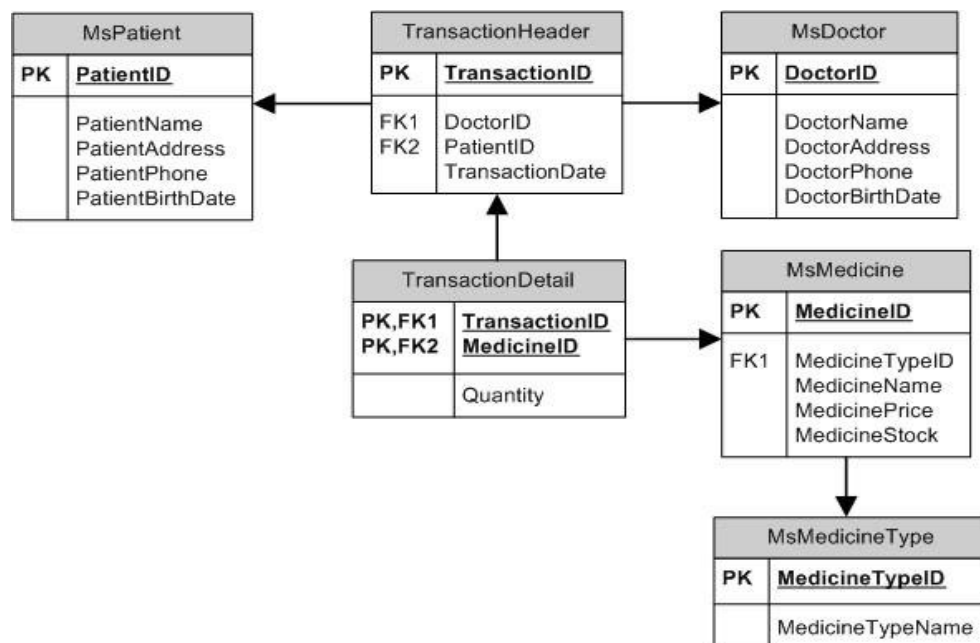
Fungsi UPPER() digunakan untuk dapat mengubah string menjadi huruf besar.

Syntax

UPPER(text)

Functions yang ada di soal penjelasannya silahkan lihat di modul 9

4. Relational Table



5. Assessment

- a. Tampilkan MedicineID dan MedicineName dimana MedicineTypeName merupakan 'Balm'.
(in)

	MedicineID	MedicineName
1	ME003	Balpirik
2	ME004	Vicks

- b. Tampilkan DoctorID, DoctorName, DoctorPhone, dan DoctorBirthYear (didapat dari tahun DoctorBirthDate) dimana dokter tersebut tidak pernah melayani pemeriksaan.
(year, not in)

	DoctorID	DoctorName	DoctorPhone	DoctorBirthYear
1	DC009	Budiman	+6289991221	1954

- c. Tampilkan MedicineName, MedicineTypeName, dan MedicinePrice dimana obat tersebut terjual pada bulan ke-12.

(exists, month)

	MedicineName	MedicineTypeName	MedicinePrice
1	OBH Combi Plus	Syrup	18000
2	Balpink	Balm	15500
3	Decadryl	Syrup	15000

- d. Tampilkan MedicineID, MedicineName, dan MedicinePrice (didapat dengan menambahkan kata 'Rp.' di awal) dimana obat tersebut tidak dijual oleh dokter yang memiliki DoctorID = 'DC001'.

(cast, not exists)

	MedicineID	MedicineName	MedicinePrice
1	ME003	Balpink	Rp.15500
2	ME004	Vicks	Rp.16000
3	ME007	Claritin	Rp.14500
4	ME009	Bisolvon	Rp.25000

- e. Tampilkan PatientID, PatientName, dan PatientBirthYear (didapat dari tahun PatientBirthDate) dimana pasien tersebut dilayani oleh dokter yang berumur lebih muda dari pada pasiennya.

(year, in)

	PatientID	PatientName	PatientBirthYear
1	PA001	Fifilia	1991
2	PA009	Jeklin Harefa	1976
3	PA010	Davin Timothy	1955

- f. Tampilkan TransactionID dan TransactionDate (dengan format 'yyyy.mm.dd') dimana MedicineTypeName bukan merupakan 'Syrup' dan MedicinePrice lebih besar dari 15000.

(convert, exists, in)

	TransactionID	TransactionDate
1	TR001	2011.10.22
2	TR004	2011.05.05
3	TR006	2011.12.01
4	TR015	2012.07.31

- g. Tampilkan PatientID, PatientName, PatientRoadAddress (didapat dari PatientAddress dengan menghilangkan kata 'Street') dimana panjang karakter PatientName lebih besar dari rata-rata panjang karakter PatientName keseluruhan. (replace, alias subquery, avg, len)

	PatientID	PatientName	PatientRoadAddress
1	PA003	Ferlim Putra	Kapuk
2	PA004	Falerian Jonathan	Sunter
3	PA005	Gradiyanto Sanjaya	Greenville
4	PA006	Hianto Hendry	Syahdan
5	PA009	Jeklin Harefa	Web
6	PA010	Davin Timothy	Strong Horse

- h. Tampilkan PatientName, PatientBirthDate (dengan format 'dd/mm/yy'), dan PatientAddress dimana transaksi dilakukan di antara tanggal 15 dan 25, dan DoctorName mengandung huruf 'j' atau 'k'.

(convert, in, day, between, like)

	PatientName	PatientBirthDate	PatientAddress
1	Junius	03/03/98	Kijang Street
2	Davin Timothy	15/12/55	Strong Horse Street

- i. Tampilkan PatientID, PatientName, Initial (didapat dari 2 karakter pertama PatientName dengan format uppercase), PatientBirthDate, dan TransactionCount (didapat dari banyaknya pemeriksaan yang dilakukan pasien dan ditambahkan kata ' Transaction(s)' di akhir).

(upper, left, cast, subquery, count)

	PatientID	PatientName	Initial	PatientBirthDate	TransactionCount
1	PA001	Fifilia	FI	1991-12-24	3 Transaction(s)
2	PA002	Calvin	CA	2003-12-23	1 Transaction(s)
3	PA003	Ferlim Putra	FE	2001-08-02	2 Transaction(s)
4	PA004	Falerian Jonathan	FA	1974-05-30	1 Transaction(s)
5	PA005	Gradiyanto Sanjaya	GR	1982-07-31	1 Transaction(s)
6	PA006	Hianto Hendry	HI	1987-04-01	1 Transaction(s)
7	PA007	Junius	JU	1998-03-03	1 Transaction(s)
8	PA008	Andi Juardi	AN	2005-11-30	3 Transaction(s)
9	PA009	Jeklin Harefa	JE	1976-02-25	1 Transaction(s)
10	PA010	Davin Timothy	DA	1955-12-15	1 Transaction(s)

- j. Tampilkan DoctorID, DoctorName, dan DoctorBirthDate (dengan format 'mon dd yyyy') dimana dokter tersebut merupakan dokter tertua atau dokter termuda dari semua dokter yang ada.

(convert, subquery, min, max)

	DoctorID	DoctorName	DoctorBirthDate
1	DC004	Mellisa Pratiwi	Aug 8 1994
2	DC009	Budiman	Feb 2 1954

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, Management. 6th edition. Pearson Education. USA. ISBN: 978-1-29206118-4, Chapter 6



MODULE 11

DATA MANIPULATION V



DESKRIPSI TEMA

Construct query of SQL that suitable with the problem

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to SQL Data Manipulation: Simple SQL command, Sorting the query results, Date Functions, Character Functions, Operator <, >, =, %, /

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These Module have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

LANGKAH-LANGKAH PRAKTIKUM

1. Subqueries, Multi Table Queries & Combining Result Tables

2. Syntax

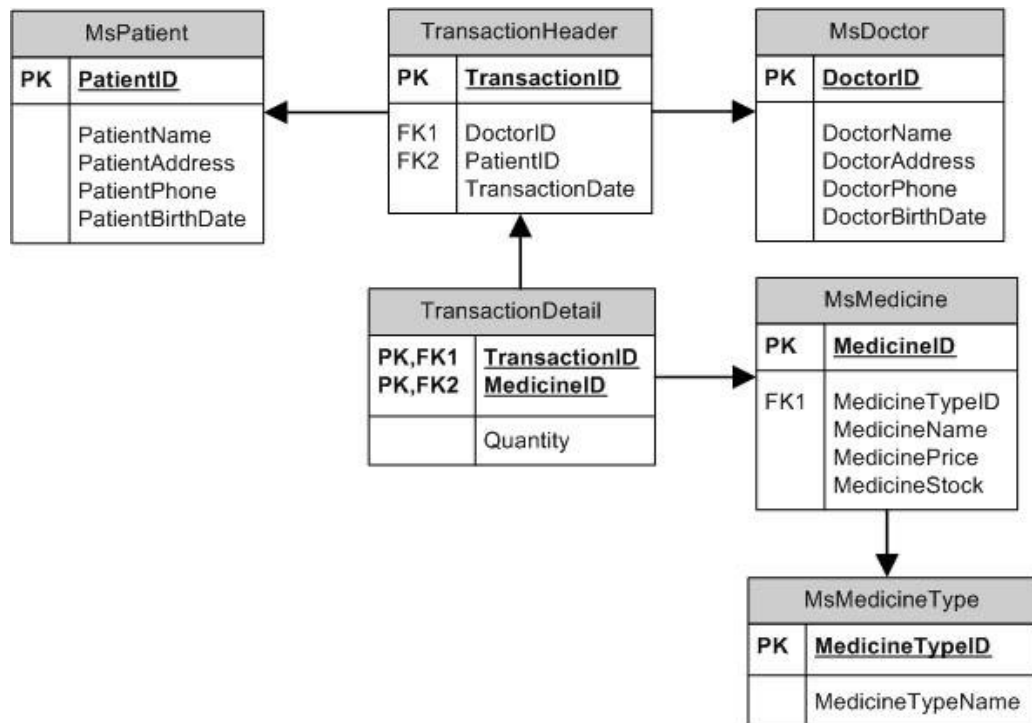
JOIN

```
SELECT field1, field2, field3
FROM first_table
[INNER JOIN / LEFT JOIN / RIGHT JOIN] second_table
ON first_table.keyfield = second_table.foreign_keyfield
```

UNION

```
SQL Statement 1
[UNION / UNION ALL]
SQL Statement 2
```

3. Relational Table



4. Assessment

- a. Tampilkan MedicineID, MedicineName, MedicineTypeName dan MedicineStock dimana panjang MedicineName lebih besar dari 10 karakter.

(join, len)

	MedicineID	MedicineName	MedicineTypeName	MedicineStock
1	ME001	OBH Combi Plus	Syrup	50

- b. Tampilkan DoctorID, DoctorName, TransactionID, dan TransactionDate dimana dokter tersebut belum pernah melayani transaksi pemeriksaan.

(left join, is null)

	DoctorID	DoctorName	TransactionID	TransactionDate
1	DC009	Budiman	NULL	NULL

- c. Tampilkan PatientID, PatientName, dan TransactionID dimana pasien pernah melakukan pemeriksaan di antara bulan 3 dan 7 dan obat yang dibeli memiliki kode 'ME001'.

(join, month, between)

	PatientID	PatientName	TransactionID
1	PA004	Falerian Jonathan	TR004
2	PA001	Fifilia	TR008
3	PA008	Andi Juardi	TR011

- d. Tampilkan TransactionID, DoctorName, TransactionDate, MedicineName, dan Quantity dimana MedicineName yang dikeluarkan diakhiri dengan huruf 'L' dan Quantity nya lebih dari 5.

(join, like)

	TransactionID	DoctorName	TransactionDate	MedicineName	Quantity
1	TR001	Nico Natalie	2011-10-22	Decadryl	8
2	TR002	Yogie Soesanto	2008-02-29	Panadol	55
3	TR013	Yogie Soesanto	2010-10-20	Decadryl	7

- e. Tampilkan TransactionID, PatientID, PatientName, DoctorID, dan DoctorName dimana kata kedua pada DoctorName tidak diawali dengan huruf 's' atau 'p' dan Quantity obat yang dibeli merupakan bilangan kelipatan 5.

(join, not like, in)

	TransactionID	PatientID	PatientName	DoctorID	DoctorName
1	TR006	PA006	Hianto Hendry	DC003	Julian Wesley
2	TR007	PA007	Junius	DC006	Franky Quo

- f. Tampilkan Code (didapat dari PatientID) dan Name (didapat dari PatientName) dimana PatientName hanya terdiri dari satu kata digabungkan dengan DoctorID dan DoctorName dimana DoctorName hanya terdiri dari satu kata.

(union, not like)

	Code	Name
1	DC009	Budiman
2	PA001	Ffilia
3	PA002	Calvin
4	PA007	Junius

- g. Tampilkan MedicineID, MedicineName, MedicineTypeName, dan MedicinePrice dimana MedicineName diawali dengan huruf 'a' sampai huruf 'e' digabungkan dengan data yang sama dimana panjang MedicineName sama dengan panjang MedicineTypeName.

(like, union, len)

	MedicineID	MedicineName	MedicineTypeName	MedicinePrice
1	ME002	Aspirin	Tablet	5000
2	ME003	Balpink	Balm	15500
3	ME006	Komix	Syrup	2000
4	ME007	Clartin	Tablet	14500
5	ME008	Decadryl	Syrup	15000
6	ME009	Bisolvon	Syrup	25000
7	ME010	Woods	Syrup	18000

- h. Tampilkan Name (didapat dari DoctorName), Address (didapat dari DoctorAddress), dan BirthDate (didapat dari DoctorBirthDate dengan format 'yyyy-mm-dd') dimana dokter tersebut melayani transaksi dengan 3 digit terakhir TransactionID-nya merupakan bilangan kelipatan 3 digabungkan dengan PatientName, PatientAddress, dan PatientBirthDate dimana pasien tersebut melakukan transaksi saat berumur kurang dari 10 tahun. Urutkan data-data tersebut berdasarkan Name secara descending.

(convert, join, right, union, datediff, order by)

	Name	Address	BirthDate
1	Yudy Pumama	Brother Street	1965-05-22
2	Nico Natalie	Anggrek Street	1992-12-15
3	Mellisa Pratiwi	Bowl Street	1994-08-08
4	Julian Wesley	Syahdan Street	1964-03-07
5	Calvin	Syahdan Street	2003-12-23
6	Andi Juardi	Google Street	2005-11-30

- i. Tampilkan Name (didapat dari kata kedua DoctorName) dan Phone (didapat dari DoctorPhone dengan mengubah '+62' menjadi '0') dimana MedicineTypeID merupakan 'MT003' digabungkan dengan kata kedua PatientName dan PatientPhone dimana bulan pada TransactionDate sama dengan bulan pada PatientBirthDate.

(substring, charindex, replace, union, month)

	Name	Phone
1	Jonathan	021921125252
2	Natalie	08181231345
3	Pratiwi	021112331
4	Saswita	0216600127
5	Wesley	08181212

- j. Tampilkan DoctorID, DoctorName, dan TransactionHandled (didapat dari banyaknya pemeriksaan yang dilayani) dimana dokter tersebut melayani pemeriksaan di tahun 2012, jika data yang ditampilkan berulang cukup tampilkan sekali saja.

(distinct, join, alias subquery, count, group by, year)

	DoctorID	DoctorName	TransactionHandled
1	DC004	Mellisa Pratiwi	1
2	DC005	Indra Saswita	2
3	DC006	Franky Quo	1
4	DC008	Yudy Pumama	2

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 6

MODULE 12

NORMALIZATION



DESKRIPSI TEMA

Design database using structured data model

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to Normalization: 1 Normal Form (1NF), 2 Normal Form (2NF), 3 Normal Form (3NF)

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These slides have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 12&14

LANGKAH-LANGKAH PRAKTIKUM

Untuk merancang database yang baik, biasa dilakukan normalisasi. Normalisasi merupakan sebuah teknik untuk menghasilkan set relasi dengan property yang desirable dan memberikan data sesuai dengan kebutuhan enterprise.

Tujuan normalisasi yaitu:

- mengidentifikasi hubungan antar atribut
- mengkombinasikan atribut untuk membentuk relasi
- mengkombinasikan relasi untuk membentuk database
- menghindari anomaly

Proses Normalisasi

1. UNF

Dalam proses normalisasi UNF kita menampilkan semua field atau atribut yang ada dalam suatu form yang ingin kita normalisasi.

2. 1NF

Sebuah relasi berada dalam 1NF jika relasi tersebut tidak berisi atribut yang berulang (repeating group), field hasil perhitungan dihilangkan dan sudah mempunyai primary key.

3. 2NF

Sebuah relasi berada dalam 2NF jika relasi tersebut dalam 1NF dan untuk setiap atribut non key bergantung fungsional penuh kepada primary key. Jadi pada 2NF kita akan menghilangkan ketergantungan sebagian / partial : ketergantungan field-field tertentu hanya kepada salah satu key yang composit.

Contoh :

Tabel Mahasiswa (Nim, Nama, Alamat)

Nama & alamat tergantung pada Nim dalam arti dengan Nim kita dapat menentukan nama maupun alamat sebaliknya nama / alamat tidak menentukan nim, maka diartikan bahwa nama & alamat tergantung secara partial kepada nim.

4. 3NF

Sebuah relasi berada dalam 3NF bila relasi tersebut dalam 1NF dan 2NF dan tidak ada atribut non key yang tergantung fungsional kepada atribut non key yang lainnya (transitive dependency).

Contoh :

Tabel Pegawai (NoPegawai, honor, KdProyek, Tanggal)

KdProyek & Tanggal adalah atribut non key. Tapi tanggal bergantung pada KdProyek. Pemecahannya dengan membagi menjadi 2 relasi :

Proyek (KdProyek, Tanggal)

PegProyek (Nopegawai, honor, KdProyek)

5. BCNF (Bentuk Normal Boyce-Codd, 1974)

Relational R dikatakan dalam BCNF jika dan hanya jika setiap determinan adalah suatu candidate key sehingga saling overlap. Tujuan BCNF adalah memisahkan determinan yang bukan merupakan candidate key pada satu relasi tertentu. Kodisinya :

A, B \rightarrow C, D

A, C \rightarrow B, D

B, C \rightarrow A, D



Contoh:

Tabel interview (NoClient, tanggalInterview, waktuInterview, NoStaff, NoRuang) Dipisahkan menjadi :

Tabel Interview (NoClient, tanggalInterview, waktuInterview, NoStaff)

Tabel RuangStaff (NoStaff, tanggalInterview, NoRuang)

Berikut adalah latihan membuat Normalisasi:



Sales Order Form
2010 / 2011

Sales Order Number : SO001
Payment Type : Cash
Customer ID : CS001
Customer Name : Lucas
Customer Contact : (656) 617 2727
Customer Address : P.O. Box 303, 1418 Mi Street, Jakarta

No	Lego ID	Lego Name	Category	Sale Price	Quantity	Sub Total
1	L001	Bleach - 01	Komik / Manga	Rp12,000.00	1	Rp12,000.00
2	L002	Naruto - 01	Komik / Manga	Rp12,000.00	1	Rp12,000.00
3	L003	Child	Child	Rp36,000.00	1	Rp36,000.00
4	L004	OODAD - Lars Mathiasen	Teknologi	Rp49,000.00	1	Rp49,000.00
5	L006	Abstract	Nature	Rp56,000.00	2	Rp112,000.00
Grand Total						Rp221,000.00

Jakarta, August 14, 2010

Janies
{ Staff Sales }

Sales Order

UNF

Sales Order (No. SO, Payment Type, Customer ID, Customer Name, Customer Contact, Customer Address, {No, Lego ID, Lego Name, Category, Sale Price, Quantity, Sub Total}, Grand Total, SO Date, Employee Name, Position)

1NF

Sales Order (No. SO, Customer ID, Payment Type, Customer Name, Customer Contact, Customer Address, SO Date, Employee Name, Position)

Detail SO (No. SO, Lego ID, Lego Name, Category, Sale Price, Quantity)

2NF

Sales Order (No. SO, Customer ID, Payment Type, Customer Name, Customer Contact, Customer Address, SO Date, Employee Name, Position)

Detail SO (No. SO, Lego ID, Quantity)

Lego (Lego ID, Lego Name, Category, Sale Price)

3NF

Customer (Customer ID, Customer Name, Customer Contact, Customer Address) Customer Category (Customer ID, Category)

Position (Position ID, Position)

Employee (Employee ID, Position ID, Employee Name, Employee Contact, Employee Address,

Employee Salary)

Payment Type (Payment Type ID, Payment Type)

Sales Order (No. SO, Payment Type ID, Customer ID, Employee ID, SO Date) Detail SO (No. SO, Lego ID, Quantity)

Category (Category ID, Category)

Lego (Lego ID, Category ID, Lego Name, Sale Price, Stock)



Purchase Order
2010 / 2011

Purchase Number : PO001
Supplier ID : S002
Supplier Name : Aretha Addison
Supplier Contact : (891) 424 5486
Supplier Address : Ap #7536034 Ut Road, Jakarta

No	Lego ID	Lego Name	Category	Purchase Price	Quantity	Sub Total
1	L005	OOAD - Lars Mathiasen	Teknologi	Rp10,000.00	1	Rp10,000.00
2	L006	Abstract	Nature	Rp10,000.00	1	Rp10,000.00
3	L007	Abstract	Nature	Rp10,000.00	1	Rp10,000.00
Grand Total						Rp30,000.00

Jakarta, August 14, 2010

Enepard

(Staff Warehouse)

Purchase Order

UNF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, {No,Lego ID,Lego Name, Category, Purchase Price, Quantity, Sub Total}, Grand Total, PO Date, Employee Name, Position)

1NF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, PO Date, Employee Name, Position)

Detail PO (No. PO, Lego ID, Lego Name, Category, Purchase Price, Quantity)

2NF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, PO Date, Employee Name, Quantity)

Detail PO (No. PO, Lego ID, Quantity)

Lego (Lego ID, Lego Name, Category, Purchase Price)


3NF

Supplier (Supplier ID, Supplier Name, Supplier Contact, Supplier Address) Position (Position ID, Position)

Employee (Employee ID, Position ID, Employee Name, Employee Contact, Employee Address, Employee Salary)

Purchase Order (No. PO, Supplier ID, Employee ID, PO Date) Detail PO (No. PO, Lego ID, Quantity) Category (Category ID, Category)

Lego (Lego ID, Category, Lego Name, Purchase Price, Stock)



**Register Form
2010 / 2011**

No. Registration	: RL003
Member ID	: CS003
Member Name	: Kate Anderson
Member Contact	: (021) 454 3721
Member Address	: P.O. Box 390, 4512 Parturient Avenue, Jakarta
Member Email	: kanderson@yahoo.com
Member Type	: Pelajar
Registration Price	: Rp75,000.00

Jakarta, August 14, 2010

Jeanne Bolt,
.....
(Staff Sales)

Registration

UNF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Member Type, Registration Price, Registration Date, Employee Name, Position)

1NF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Payment Type, Registration Price, Registration Date, Employee Name, Position)

2NF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Payment Type, Registration Price, Registration Date, Employee Name, Position)

3NF

Member Category(Member Type ID, Member Type, Diskon)

Member (Member ID, Member Type ID, Member Name, Member Contact, Member Address, Member Email)

Position(Position ID, Position)

Employee(Employee ID, Position ID, Employee Name, Employee Contact, Employee Address, Employee Salary)

Registration(No.Registration, Member ID, Employee ID, Registration Date)

Berikut adalah contoh desain Form:



Customer | Supplier | Sales Order

Form SO

No. SO

: SO001

SO Date

: 2010-08-14 00:00:00.0

Customer ID

: CS001

Employee ID

: KR001

Payment Type ID

: PB001

First

Prev

Next

Last

Detail SO

No. SO	Book ID	Quantity
SO001	BK001	1
SO001	BK002	1
SO001	BK003	1
SO001	BK004	1
SO001	BK006	2



Customer | Supplier | Sales Order

Form Customer

Customer ID

: CS001

Customer Name

: Lucas

Customer Contact

: (656) 617 2727

Customer Address

: P.O. Box 303, 1418 M Street, Jakarta

Customer Email

: at@malesuada.com

Customer Type ID

: JP001

First

Prev

Next

Last

Laundry Store

Laundry Store is one of the most famous laundries around Multimedia Nusantara University. Because there are so many transactions that occur every day. Laundry Store has trouble sorting out their sales. To assist them in sorting their data, the company decided to hire you work and make an application to help them.

As a system analyst, you're assigned to design a database by using normalization UNF, 1 NF, 2NF and 3NF based on the form and existing business process.

Here is the business process of Laundry Store:

To take the order Laundry Store, firstly, list the name of the employee that serves the customer. After listing those, the employee will start listing the item being sent to the laundry. After all items listed, a total amount of the payment will be calculated. The customer and employee will held onto Laundry Receipt. Customer can pay via transfer, credit, or cash.

Laundry Store also lists the employee absence status, the first thing the employee should do is signing in to take their attendance. The laundry store attendance time is at 07.00 AM, above 07:00 AM the attendance status will be absent. This item will be record in the Employee absence. The Employee Absence will be printed once every day.

These are the provided forms:

1. Laundry Receipt

Laundry Store		Laundry Receipt			
		Kemanggisan Street Number 16 (021) - 5014986			
SalesID	: SA001	Customer Name	: Justinus Hermawan		
Date	: 1 Dec 2015	Phone	: 08999465851		
Payment Type	: Credit	Address	: Anggrek Street 6		
No	ItemID	ItemName	Price	Weight	SubTotal
1	IT001	Pants	Rp 20.000.00	1.5 Kg	Rp 30.000.00
2	IT002	Shirt	Rp 15.000.00	1.0 Kg	Rp 15.000.00
Total					Rp 45.000.00
Cashier (Komala Surya)					

2. Employee Absence

Laundry Store

Laundry Receipt
Kemanggisan Street Number 16
(021) - 5014986

SalesID : SA001

Customer Name : Justinus Hermawan

Date : 1 Dec 2015

Phone : 08999465851

Payment Type : Credit

Address : Anggrek Street 6

No	ItemID	ItemName	Price	Weight	SubTotal
1	IT001	Pants	Rp 20.000.00	1.5 Kg	Rp 30.000.00
2	IT002	Shirt	Rp 15.000.00	1.0 Kg	Rp 15.000.00
				Total	Rp 45.000.00

Cashier (Komala Surya)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-29206118-4, Chapter 12&14

MODULE 13

ER MODELING



DESKRIPSI TEMA

Design database using structured data model

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

Student are able to ER Modelling Design

PENUNJANG PRAKTIKUM

1. Microsoft SQL Server management studio, SQL Server 2012
2. Module Practicum
3. These slides have been adapted from Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 12&14

LANGKAH-LANGKAH PRAKTIKUM

Objectives

- Database Concept
- Multiplicity
- Data Relational
- Normalization

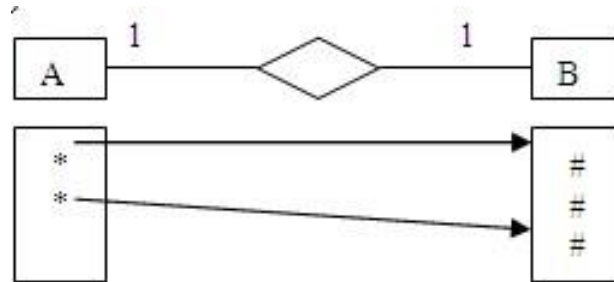
1. Database Concept

- Conceptual model: Merupakan pengumpulan / integrasi seluruh kebutuhan atribut dari para user / aplikasi menjadi satu pandangan organisasi
- Logical model: Dari model conceptual yang terbentuk dapat dipilih salah satu model data dasar logikal : hirarki, network atau relational. Kemudian dibagikan kepada para user yang berwenang
- Physical model: Bagaimana secara fisik data tersimpan pada penyimpanan sekunder, yang perlu dipertimbangkan mengenai metode akses (menyimpan / stored dan mengambil / retrieval) dan teknik pengindeksan untuk retrieval (pencarian lebih cepat).

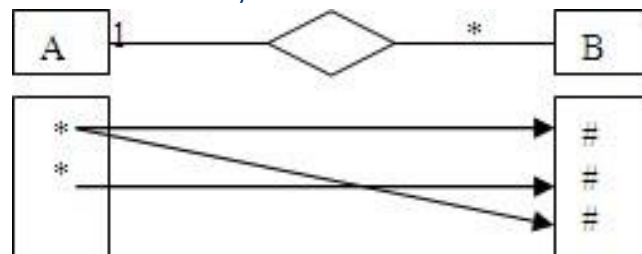
2. Multiplicity

Bertujuan untuk mengidentifikasi entity dan hubungan (relationship) pada model data Ada 3 macam hubungan :

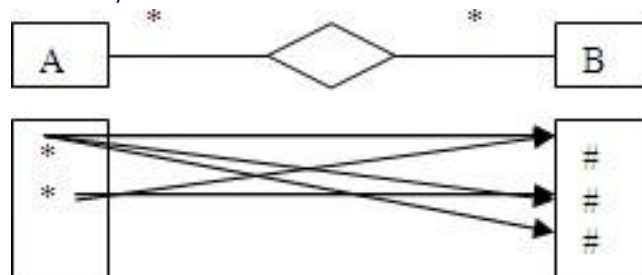
- a. One To One : Terjadi bila tiap anggota entity A hanya boleh berpasangan dengan satu anggota



- b. One To Many: Terjadi bila tiap anggota entity A boleh berpasangan dengan lebih dari satu anggota B tetapi tidak berlaku sebaliknya



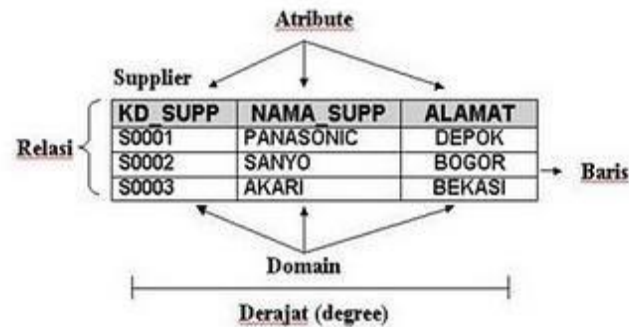
- c. Many To Many: Terjadi bila tiap anggota A boleh berpasangan dengan lebih dari satu anggota B dan berlaku untuk sebaliknya



3. Data Relational

Relasi adalah hubungan sebuah tabel yang terdiri dari beberapa kolom dan beberapa baris.

- a. Berikut adalah isu-isu yang terdapat dalam pembahasan Data relationship:
 - i. Atribut adalah Kolom pada sebuah relasi
 - ii. Tupel adalah Baris pada sebuah relasi



- iii. Domain adalah Kumpulan nilai yang valid untuk satu atau lebih atribut
- iv. Derajat (degree) adalah Jumlah atribut dalam sebuah relasi
- v. Cardinality adalah Jumlah tupel dalam sebuah relasi

b. Dalam Data Relasional terdapat pula Relational Key, yang terdiri dari :

- i. Super key yakni, Satu atribut/kumpulan atribut yang secara unik mengidentifikasi sebuah tupel di dalam relasi

NIM	NAMA	NO_KWIT	JUMLAH
0222500250	TUTI	789	50000
0222300023	WATI	254	60000
0144500024	ALE	365	80000

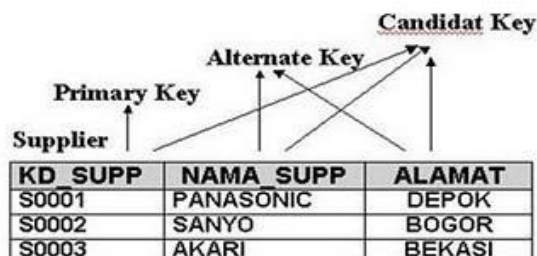
Candidate key

- ii. Candidate key yakni, Atribut di dalam relasi yang biasanya mempunyai nilai unik dan merupakan Field-field yang bisa dipilih (dipakai) menjadi primary key.

NIM	NAMA	UMUR
0222500250	TUTI	21
0222300023	WATI	20
0144500024	ALE	24

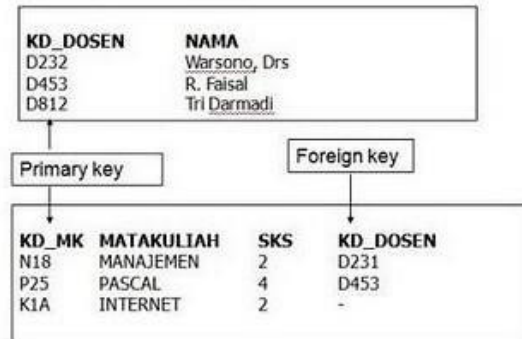
Primari Key

- iii. Primary key yakni, Candidate key yang dipilih untuk mengidentifikasi tupel secara unik dalam relasi dan merupakan Field yang mengidentifikasi sebuah record dalam file, bersifat unik.



- iv. Alternate key yakni, Candidate key yang tidak dipilih sebagai primary key.

- v. Foreign key yakni, Atribut dengan domain yang sama yang menjadi kunci utama pada sebuah relasi tetapi pada relasi lain atribut tersebut hanya sebagai atribut biasa. Foreign Key ini adalah Field yang bukan key, tetapi adalah key pada file yang lain.



4. Normalization

Untuk merancang database yang baik, biasa dilakukan normalisasi. Normalisasi merupakan sebuah teknik untuk menghasilkan set relasi dengan property yang desirable dan memberikan data sesuai dengan kebutuhan enterprise.

Tujuan normalisasi yaitu:

- mengidentifikasi hubungan antar atribut
- mengkombinasikan atribut untuk membentuk relasi
- mengkombinasikan relasi untuk membentuk database
- menghindari anomaly

Proses Normalisasi

1. UNF

Dalam proses normalisasi UNF kita menampilkan semua field atau atribut yang ada dalam suatu form yang ingin kita normalisasi.

2. 1NF

Sebuah relasi berada dalam 1NF jika relasi tersebut tidak berisi atribut yang berulang (repeating group), field hasil perhitungan dihilangkan dan sudah mempunyai primary key.

3. 2NF

Sebuah relasi berada dalam 2NF jika relasi tersebut dalam 1NF dan untuk setiap atribut non key bergantung fungsional penuh kepada primary key. Jadi pada 2NF kita akan menghilangkan ketergantungan sebagian / partial : ketergantungan field-field tertentu hanya kepada salah satu key yang composit.

Contoh :

Tabel Mahasiswa (Nim, Nama, Alamat)

Nama & alamat tergantung pada Nim dalam arti dengan Nim kita dapat menentukan nama maupun alamat sebaliknya nama / alamat tidak menentukan nim, maka diartikan bahwa nama & alamat tergantung secara partial kepada nim.

4. 3NF

Sebuah relasi berada dalam 3NF bila relasi tersebut dalam 1NF dan 2NF dan tidak ada atribut non key yang tergantung fungsional kepada atribut non key yang lainnya (transitive dependency).

Contoh :

Tabel Pegawai (NoPegawai, honor, KdProyek, Tanggal)

KdProyek & Tanggal adalah atribut non key. Tapi tanggal bergantung pada KdProyek. Pemecahannya dengan membagi menjadi 2 relasi :

Proyek (KdProyek, Tanggal)

PegProyek (Nopegawai, honor, KdProyek)

5. BCNF (Bentuk Normal Boyce-Codd, 1974)

Relational R dikatakan dalam BCNF jika dan hanya jika setiap determinan adalah suatu candidate key sehingga saling overlap. Tujuan BCNF adalah memisahkan determinan yang bukan merupakan candidate key pada satu relasi tertentu. Kodisinya :

A, B -> C, D

A, C -> B, D

B, C -> A, D


Contoh:

Tabel interview (NoClient, tanggalInterview, waktuInterview, NoStaff, NoRuang) Dipisahkan menjadi :

Tabel Interview (NoClient, tanggalInterview, waktuInterview, NoStaff)

Tabel RuangStaff (NoStaff, tanggalInterview, NoRuang)

Berikut adalah latihan membuat Normalisasi:



**Sales Order Form
2010 / 2011**

Sales Order Number : SD001
Payment Type : Cash
Customer ID : CS001
Customer Name : Lucas
Customer Contact : (856) 617 2727
Customer Address : P.O. Box 303, 1418 Mi Street, Jakarta

No	Lego ID	Lego Name	Category	Sale Price	Quantity	Sub Total
1	L001	Bleach - 01	Komik / Manga	Rp12,000.00	1	Rp12,000.00
2	L002	Naruto - 01	Komik / Manga	Rp12,000.00	1	Rp12,000.00
3	L003	Child	Child	Rp36,000.00	1	Rp36,000.00
4	L004	OOAD - Lars Mathiasen	Teknologi	Rp49,000.00	1	Rp49,000.00
5	L006	Abstract	Nature	Rp56,000.00	2	Rp112,000.00
Grand Total						Rp221,000.00

Jakarta, August 14, 2010

Janies
 (Staff Sales)

Sales Order

UNF

Sales Order (No. SO, Payment Type, Customer ID, Customer Name, Customer Contact, Customer Address, {No, Lego ID, Lego Name, Category, Sale Price, Quantity, Sub Total}, Grand Total, SO Date, Employee Name, Position)

1NF

Sales Order (No. SO, Customer ID, Payment Type, Customer Name, Customer Contact, Customer Address, SO Date, Employee Name, Position)

Detail SO (No. SO, Lego ID, Lego Name, Category, Sale Price, Quantity)

2NF

Sales Order (No. SO, Customer ID, Payment Type, Customer Name, Customer Contact, Customer Address, SO Date, Employee Name, Position)

Detail SO (No. SO, Lego ID, Quantity)

Lego (Lego ID, Lego Name, Category, Sale Price)

3NF

Customer (Customer ID, Customer Name, Customer Contact, Customer Address) Customer Category (Customer ID, Category)

Position (Position ID, Position)

Employee (Employee ID, Position ID, Employee Name, Employee Contact, Employee Address, Employee Salary)

Payment Type (Payment Type ID, Payment Type)

Sales Order (No. SO, Payment Type ID, Customer ID, Employee ID, SO Date) Detail SO (No. SO, Lego ID, Quantity)

Category (Category ID, Category)

Lego (Lego ID, Category ID, Lego Name, Sale Price, Stock)



Purchase Order
2010 / 2011

Purchase Number : PO001
Supplier ID : S002
Supplier Name : Aretha Addison
Supplier Contact : (891) 424 5486
Supplier Address : Ap #7536034 Ut Road, Jakarta

No	Lego ID	Lego Name	Category	Purchase Price	Quantity	Sub Total
1	L005	OOAD - Lars Mathiasen	Teknologi	Rp10,000.00	1	Rp10,000.00
2	L006	Abstract	Nature	Rp10,000.00	1	Rp10,000.00
3	L007	Abstract	Nature	Rp10,000.00	1	Rp10,000.00
Grand Total						Rp30,000.00

Jakarta, August 14, 2010

Enepard
(Staff Warehouse)

Purchase Order

UNF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, {No,Lego ID,Lego Name, Category, Purchase Price, Quantity, Sub Total}, Grand Total, PO Date, Employee Name, Position)

1NF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, PO Date, Employee Name, Position)

Detail PO (No. PO, Lego ID, Lego Name, Category, Purchase Price, Quantity)

2NF

Purchase Order (No. PO, Supplier ID, Supplier Name, Supplier Contact, Supplier Address, PO Date, Employee Name, Quantity)

Detail PO (No. PO, Lego ID, Quantity)

Lego (Lego ID, Lego Name, Category, Purchase Price)


3NF

Supplier (Supplier ID, Supplier Name, Supplier Contact, Supplier Address) Position (Position ID, Position)

Employee(Employee ID, Position ID, Employee Name, Employee Contact, Employee Address, Employee Salary)

Purchase Order (No. PO, Supplier ID, Employee ID, PO Date) Detail PO (No. PO, Lego ID, Quantity)

Category (Category ID, Category)
Lego (Lego ID, Category, Lego Name, Purchase Price, Stock)



**Register Form
2010 / 2011**

No. Registration	: RL003
Member ID	: CS003
Member Name	: Kate Anderson
Member Contact	: (021) 454 3721
Member Address	: P.O. Box 390, 4512 Parturient Avenue, Jakarta
Member Email	: kanderson@yahoo.com
Member Type	: Pelajar
Registration Price	: Rp75,000.00

Jakarta, August 14, 2010

Jeanne Bolt,
.....
(Staff Sales)

Registration

UNF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Member Type, Registration Price, Registration Date, Employee Name, Position)

1NF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Payment Type, Registration Price, Registration Date, Employee Name, Position)

2NF

Registration (No.Registration, Member ID, Member Name, Member Contact, Member Address, Member Email, Payment Type, Registration Price, Registration Date, Employee Name, Position)

3NF

Member Category(Member Type ID, Member Type, Diskon)

Member (Member ID, Member Type ID, Member Name, Member Contact, Member Address, Member Email)

Position(Position ID, Position)

Employee(Employee ID, Position ID, Employee Name, Employee Contact, Employee Address, Employee Salary)

Berikut adalah contoh desain Form:



books are a good thing to be a good programmer

Bluejack Book Store

Customer | Supplier | Sales Order

Form SO

No. SO : SO001

SO Date : 2010-08-14 00:00:00.0

Customer ID : CS001

Employee ID : KR001

Payment Type ID : PB001

First Prev Next Last

Detail SO

No. SO	Book ID	Quantity
SO001	BK001	1
SO001	BK002	1
SO001	BK003	1
SO001	BK004	1
SO001	BK006	2



books are a good thing to be a good programmer

Bluejack Book Store

Customer | Supplier | Sales Order

Form Customer

Customer ID : CS001

Customer Name : Lucas

Customer Contact : (656) 617 2727

Customer Address : P.O. Box 303, 1418 MI Street, Jakarta

Customer Email : at@malesuada.com

Customer Type ID : JP001

First Prev Next Last

Assessment

Laundry Store

Laundry Store is one of the most famous laundries around Multimedia Nusantara University. Because there are so many transactions that occur every day. Laundry Store has trouble sorting out their sales. To assist them in sorting their data, the company decided to hire you work and make an application to help them.

As a system analyst, you're assigned to design a database by using normalization UNF, 1 NF, 2NF an 3NF along with the ERD based on the form and existing business process.

Here is the business process of Laundry Store:

To take the order Laundry Store, firstly, list the name of the employee that serves the customer.

After listing those, the employee will start listing the item being sent to the laundry. After all items listed, a total amount of the payment will be calculated. The customer and employee will held onto Laundry Receipt. Customer can pay via transfer, credit, or cash.

Laundry Store also lists the employee absence status, the first thing the employee should do is signing in to take their attendance. The laundry store attendance time is at 07.00 AM, above 07:00 AM the attendance status will be absent. This item will be record in the Employee absence. The Employee Absence will be printed once every day.

These are the provided forms:

1. Laundry Receipt

Laundry Store		Laundry Receipt			
		Kemanggisan Street Number 16 (021) - 5014986			
SalesID	: SA001	Customer Name	: Justinus Hermawan		
Date	: 1 Dec 2015	Phone	: 08999465851		
Payment Type	: Credit	Address	: Anggrek Street 6		
No	ItemID	ItemName	Price	Weight	SubTotal
1	IT001	Pants	Rp 20.000.00	1.5 Kg	Rp 30.000.00
2	IT002	Shirt	Rp 15.000.00	1.0 Kg	Rp 15.000.00
Total					Rp 45.000.00
<div style="text-align: right;"> _____ Cashier (Komala Surya) </div>					

2. Employee Absence



J
S
A
A



Employee Absence
Kemanggisan Street Number 16
[021] - 5014986

AttendanceID : SA001
Date : 1 Dec 2015

No	EmployeeID	Name	Status	Attendance Time
1	EM001	Timothy Agustian	Present	06:23:31
2	EM002	Jessica Casanova	Present	06:45:55
3	EM003	Hermanto	Absent	13:12:54
4	EM004	Stephanus	Present	06:39:56
5	EM005	Venson Wijaya	Present	06:15:34
Total Present				4
Total Absent				1

Manager (August Vin)

REFERENSI

1. Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education. USA. ISBN: 978-1-292-06118-4, Chapter 12&14

MODULE 14

CASE STUDY 2



DESKRIPSI TEMA

(Apa yang akan dipelajari mahasiswa pada minggu ini)

CAPAIAN PEMBELAJARAN MINGGUAN (SUB-CAPAIAN PEMBELAJARAN)

(Capaian pembelajaran pada minggu ini)

PENUNJANG PRAKTIKUM

1. ...
2. ...
3. dst

(+ Perlengkapan Apd/Alat Pelindung Diri Yang Harus Digunakan, Jika Ada)

LANGKAH-LANGKAH PRAKTIKUM

Studi Kasus Terlampir

REFERENSI