BADAN PUSAT STATISTIK

POC Netezza

Data Warehouse Project

10/8/2012

TABLE OF CONTENT

Contents

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

REVISION HISTORY

Version No	Description of changes	Date of Release	Prepared By	Approved By
1.0	Documentasi POC Netezza	24 September 2012	Renny I.W, Ratih N, Sekar R.R	
1.1	Documentasi POC Netezza	25 September 2012	Renny I.W, Ratih N, Sekar R.R	
1.2	Documentasi POC Netezza	8 Oktober 2012	Renny I.W	

USER APPROVAL

No	Approved By	Role/Division	Approval Date	Signature

POC NETEZZA

1. LATAR BELAKANG

BPS sebagai lembaga negara penyedia data mulai melakukan pengembangan data warehouse sebagai salah satu solusi untuk menyajikan data yang berkualitas. Era data warehouse di BPS dimulai dengan pembangunan prototype data warehouse menggunakan data ekspor impor. Pembangunan data warehouse BPS menggunakan perangkat data warehouse keluaran IBM yaitu IBM DB2 sebagai database data warehouse, IBM Infosphere Information Server sebagai ETL tools, dan IBM Cognos untuk business intelligence. Seiring berkembangnya data dalam data warehouse, performance data warehouse semakin berkurang. Untuk itu, tim data warehouse BPS perlu mencari solusi agar hasil yang diharapkan dari adanya data warehouse dapat tercapai (performance data yang acceptable).

2. TUJUAN POC

POC IBM Netezza merupakan salah satu kegiatan yang ditujukan untuk mencari solusi peningkatan performance data warehouse BPS. Adapun tujuan kegiatan POC sebagai berikut :

a. Melakukan komparasi performance antara IBM Netezza dengan IBM DB2 yang sudah diterapkan di BPS.
b. Mendapatkan masukan tentang database yang memiliki performance paling baik untuk karakteristik data yang dimiliki BPS baik dalam hal daya tampung data, kecepatan query, dan kemudahan dalam me-manage database.

Dalam kegiatan POC ini digunakan dua sumber data yaitu data Ekspor Impor dan data Sensus Penduduk 2010.

3. EKSPOR IMPOR

a. Spesifikasi Data

POC Netezza menggunakan data Ekspor Impor (clean) yang terdapat pada database DB2. Spesifikasi data ekspor impor yang digunakan sebagai berikut :

Export Import – IBM DB2	
Period	5 years data
File Size	± 409 GB
Number of Records	± 1.1 Billion records

Tabel 1. Spesifikasi Data Ekspor Impor

Adapun struktur data dan table ekspor impor di database IBM DB2 terdapat pada Lampiran 1.

b. Spesifikasi Hardware

Dalam POC Netezza ini, hardware yang digunakan yaitu server IBM DB2 dan IBM Netezza Skimmer (IBM Netezza 100). Spesifikasi hardware sebagai berikut :

Variabel	IBM DB2
Server OS	Red Hat Enterprise Linux 5 (64 bit)
Number of CPU	4 vCPU
Amount of RAM	32 576 MB
Storage Total	4 TB

Table 2. Spesifikasi Hardware IBM DB2

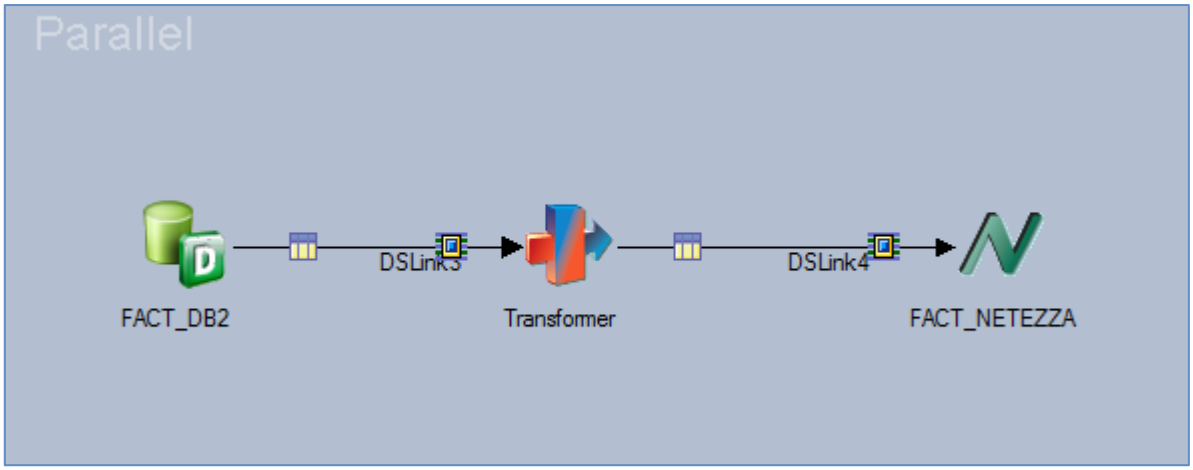
Netezza Skimmer 1 Specifications			
S-Blade	1	**Host Blade**	1
Disk Drives	9 (8 active, 1 hot spare)		
User Data in TB (Uncompressed)	2.8	**User Data in TB** (Compressed)	10
Height (cm)	30.63	**Width** (cm)	44.4
Weight (kg)	150.6	**Depth** (cm)	73.34
Heat (Max BTUs)	7,000	**Power** (Max Watts)	1,900
Power Supply	4 hot-swap/redundant – (110V = N+1, 220V = N+N) – 950W/1450W AC (110V/220V)		

Table 3. Spesifikasi Hardware IBM Netezza

c. Skenario Kegiatan

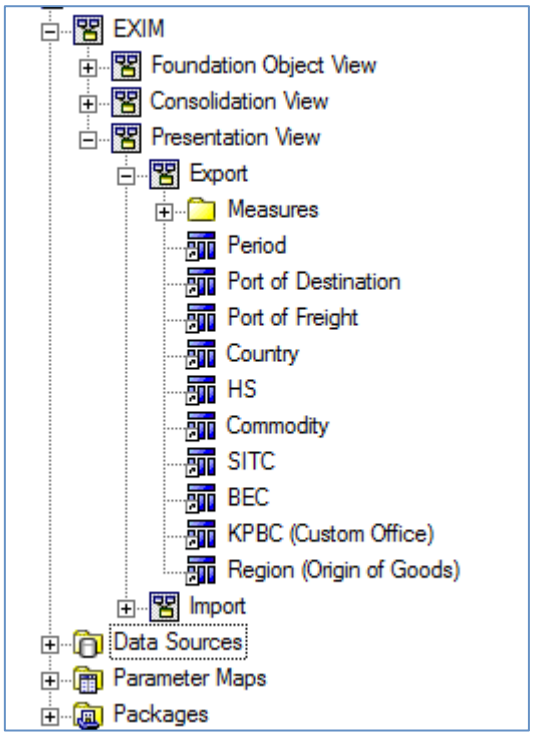
Pada IBM Netezza dibuat skema database yang sama dengan yang terdapat di DB2 (tidak tersedia relasi tabel di Netezza). Migrasi data dari IBM DB2 ke IBM Netezza menggunakan ETL tools yaitu Infosphere Information Server (IBM Data Stage). Sebelum migrasi data, diperlukan instalasi ODBC (Data Direct) Netezza ke Server Data Stage.

Proses Loading data tiap tabel menggunakan job seperti di bawah ini:



Gambar 1. Job Migrasi Data Ekspor Impor

Setelah semua data selesai di-load, proses dilanjutkan dengan membuat package Cognos dengan IBM Cognos Framework Manager. Package Netezza dibuat sama dengan package yang sudah dibuat sebelumnya pada DB2 (dengan menambahkan relasi tabel pada Netezza)



Gambar 2. Package Netezza untuk Ekspor Impor

Setelah package di-*publish*, dilakukan komparasi *performance* dengan kondisi database DB2 telah di-tuning (indexing, logging, Materialized Query Table), sedangkan pada Netezza tidak ada perlakuan apapun (tidak diperlukan tuning).

d. Komparasi Performance

Pada IBM Netezza terdapat proses *compression data,* sehingga ukuran database untuk data Ekspor Impor bisa lebih kecil. Berikut spesifikasi data Ekspor Impor pada IBM Netezza :

Export Import – IBM Netezza	
Period	5 years data
File Size	± 27.51 GB
Number of Records	± 1.1 Billion records

Tabel 4. Spesifikasi Data Ekspor Impor di IBM Netezza

Untuk menguji kemampuan IBM Netezza, dalam kegiatan POC ini menggunakan beberapa query tabel yang biasa digunakan dalam pembuatan laporan Ekspor impor oleh Subjectmatter. Query tabel dikelompokkan menjadi tiga kelompok berdasarkan kompleksitasnya, yaitu :

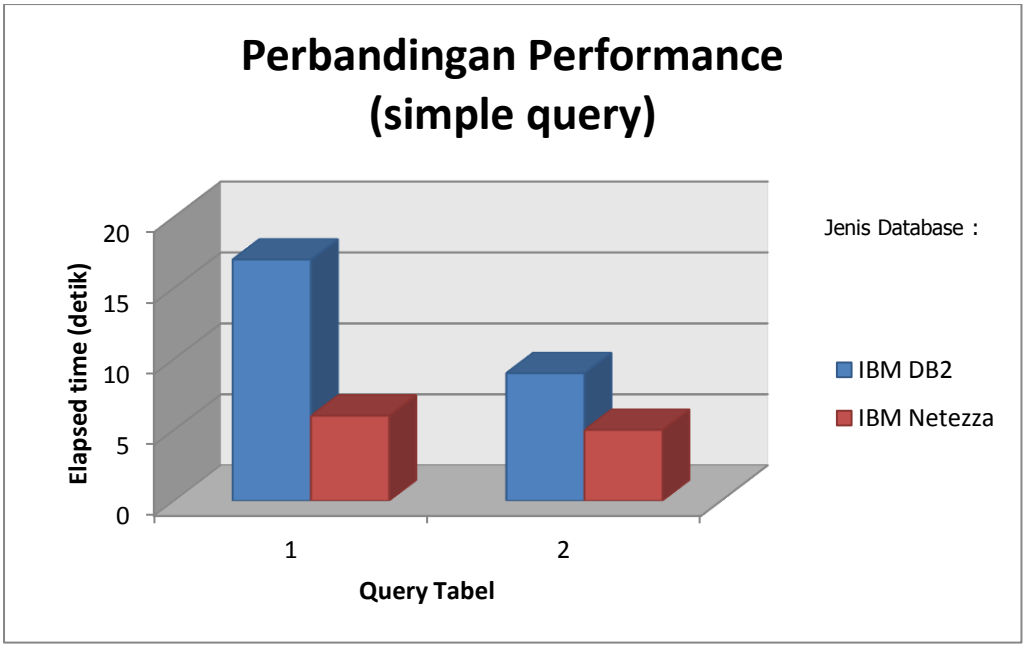
a. Simple Query : query tabel dengan join 1 variabel
    a. Export of Indonesia By Country of Destination
    b. Export of Indonesia By SITC 3 Digit
b. Complex Query : query tabel dengan join lebih dari 1 variabel dan perhitungan yang kompleks
    a. The Summary of Indonesian Exports
    b. Export of Indonesia By Province And Port of Loading
    c. Import By Province And Port Of Importation
    d. Exports by Destination Countries And Groups of Goods (Non Oil & Gas, Oil and Gas)
    e. Development of the Agricultural Products are Exported
c. Heavy Query : query tabel dengan join dimensi yang besar datanya
    a. Export by Commodity (HS) And Country of Destination
    b. Import By Commodity (HS) And Country Of Origin

Query untuk setiap tabel POC dapat dilihat pada Lampiran 2.

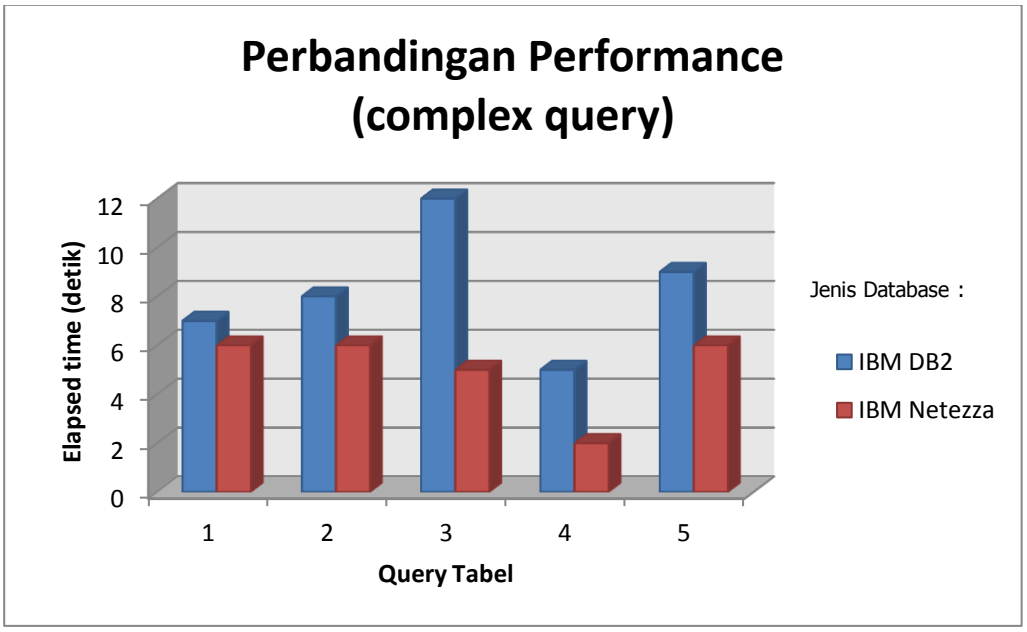
Komparasi performance antara IBM DB2 dan IBM Netezza dilakukan berdasarkan *elapsed time* dalam eksekusi tiap-tiap query tabel. Hasilnya ditunjukkan pada tabel 5.

No	Nama Tabel	Tanpa Normalisasi HW				Dengan Normalisasi HW			
		DB2		Netezza		DB2		Netezza	
		1	**2**	**1**	**2**	**1**	**2**	**1**	**2**
Simple Query									
1	Export Of Indonesia By Country Of Destination	17"	7"	9"	6"	17"	7"	18″	6″
2	Export Of Indonesia By Sitc 3 Digit	6"	1"	5"	2"	6"	1"	10″	2″
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**1.54**				**0.72**			
Complex Query									
1	The Summary Of Indonesian Exports	7"	6"	6"	6"	7"	6"	12″	6"
2	Export Of Indonesia By Province And Port Of Loading	8"	1"	6"	2"	8"	1"	12″	2"
3	Import By Province And Port Of Importation	12"	1"	5"	2"	12"	1"	10″	2"
4	Exports By Destination Countries And Groups Of Goods (Non Oil And Gas, Oil And Gas)	5"	1"	2"	6"	5"	1"	4″	6"
5	Development Of The Agricultural Products Are Exported	9"	1"	6"	2"	9"	1"	12″	2"
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**1.78**				**0.89**			
Heavy Query									
1	Export By Commodity (HS) And Country Of Destination	9' 33"	9' 30"	21"	17"	9' 33"	9' 30"	42"	17"
2	Import By Commodity (HS) And Country Of Origin	11'28"	13'20"	19"	5"	11'28"	13'20"	38"	5"
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**31.75**				**15.87**			

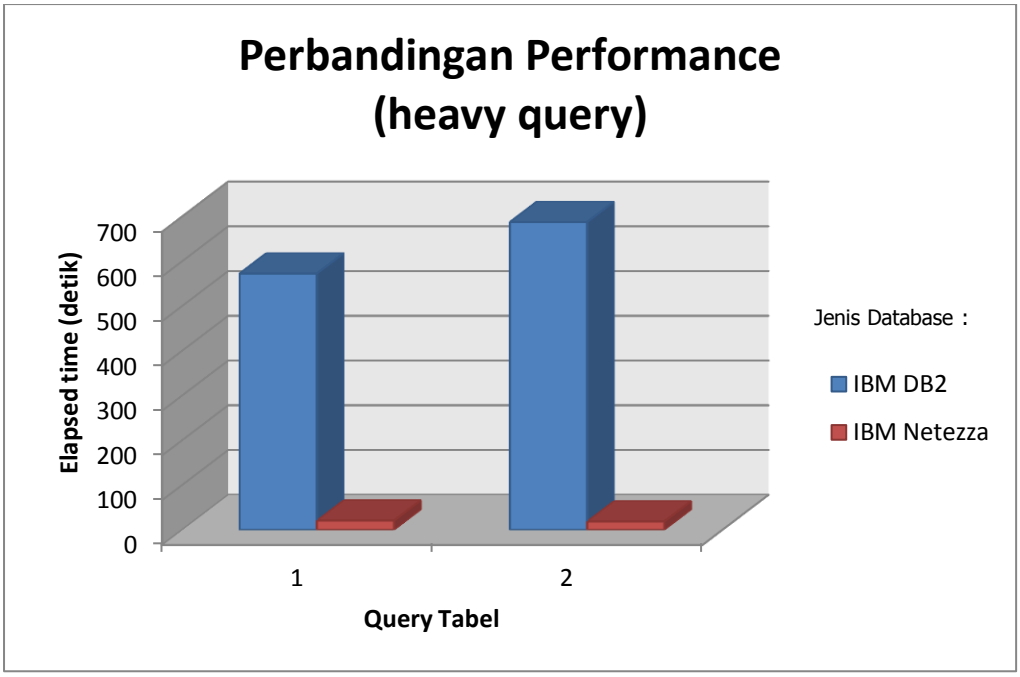
Tabel 5. Hasil Komparasi Performance Ekspor Impor



Gambar 3. Perbandingan Performance Simple Query – Ekspor Impor



Gambar 4. Perbandingan Performance Complex Query – Ekspor Impor



Gambar 5. Perbandingan Performance Heavy Query – Ekspor Impor

4. SENSUS PENDUDUK 2010

a. Spesifikasi Data

POC Netezza menggunakan data Sensus Penduduk 2010 (clean) yang terdapat pada database DB2. Spesifikasi data Sensus Penduduk 2010 yang digunakan sebagai berikut :

Sensus Penduduk 2010 – IBM DB2	
Period	1 year data (2010)
File Size	± 161 GB
Number of Records	± 237 million records

Tabel 6. Spesifikasi Data Sensus Penduduk 2010

Adapun struktur data dan table ekspor impor di database IBM DB2 terdapat pada Lampiran 3.

b. Spesifikasi Hardware

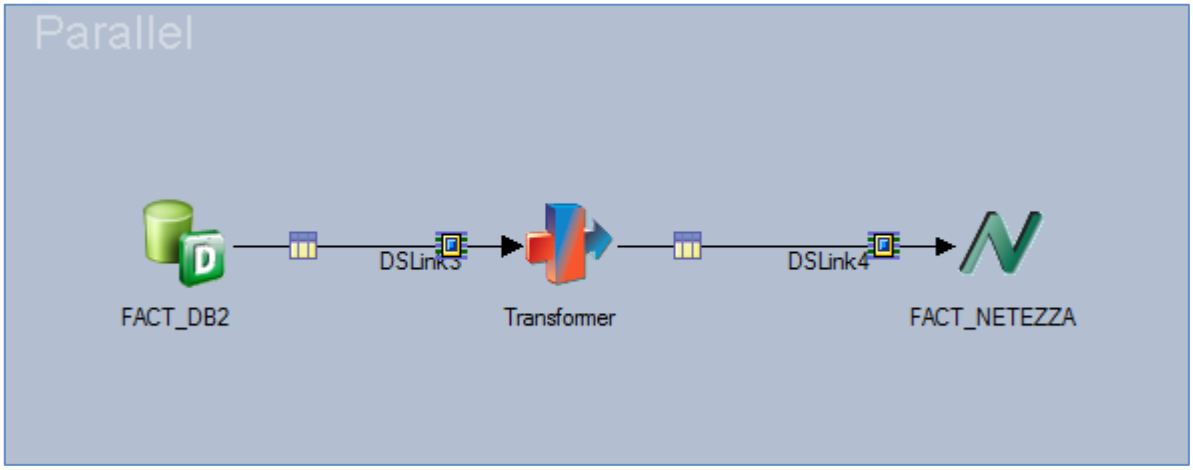
POC Netezza untuk data Sensus Penduduk 2010 menggunakan hardware yang sama dengan yang digunakan untuk data Export Import (tabel 2 dan 3). Hal ini karena

data Sensus Penduduk 2010 disimpan pada mesin DB2 yang sama dengan data Export Import.

c. Skenario Kegiatan

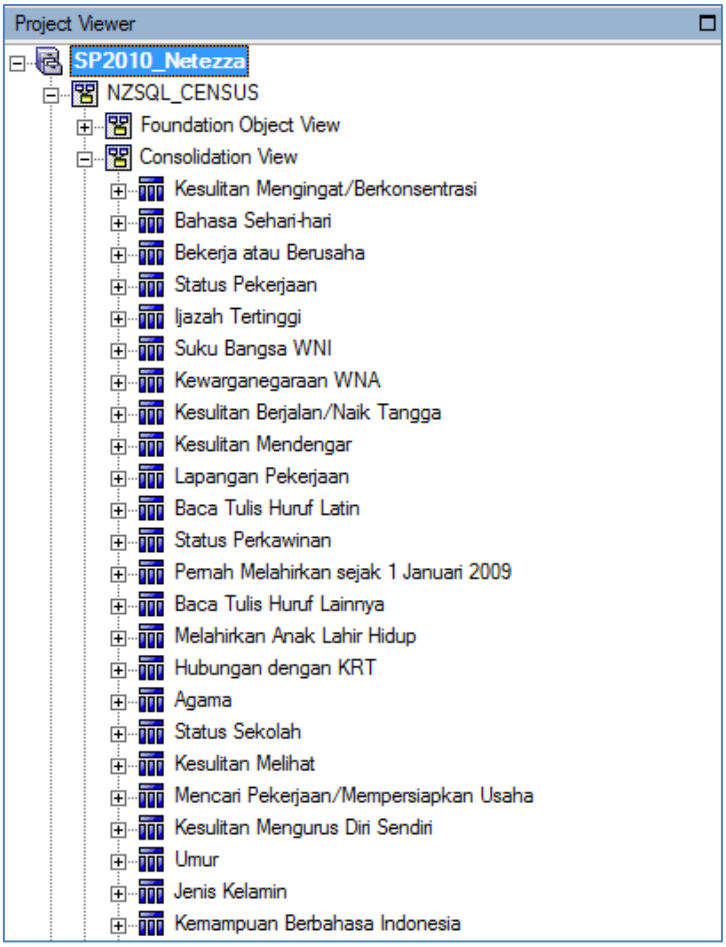
Pada IBM Netezza dibuat skema database yang sama dengan yang terdapat di DB2 (tidak tersedia relasi tabel di Netezza). Migrasi data dari IBM DB2 ke IBM Netezza menggunakan ETL tools yaitu Infosphere Information Server (IBM Data Stage).

Proses Loading data tiap tabel menggunakan job seperti di bawah ini:



Gambar 6. Job Migrasi Data Sensus Penduduk 2010

Setelah semua data selesai di-load, proses dilanjutkan dengan membuat package Cognos dengan IBM Cognos Framework Manager. Package Sensus Penduduk di Netezza dibuat sama dengan package yang sudah dibuat sebelumnya pada DB2 (dengan menambahkan relasi tabel pada Netezza).



Gambar 7. Package Netezza Sensus Penduduk

Setelah package di-*publish*, dilakukan komparasi *performance* dengan kondisi database DB2 telah di-tuning (indexing, logging, Materialized Query Table), sedangkan pada Netezza tidak ada perlakuan apapun (tidak diperlukan tuning).

d. Komparasi Performance

Pada IBM Netezza terdapat proses *compression data,* sehingga ukuran database untuk data Sensus Penduduk 2010 bisa lebih kecil. Berikut spesifikasi data Sensus Penduduk 2010 pada IBM Netezza :

Sensus Penduduk 2010 – IBM Netezza	
Period	1 year data (2010)
File Size	± 10.13 GB
Number of Records	± 237 million records

Tabel 7. Spesifikasi Data Sensus Penduduk 2010 di IBM Netezza

POC dengan data Sensus Penduduk 2010 menggunakan beberapa query tabel yang biasa digunakan dalam pembuatan laporan Sensus Penduduk 2010 oleh

Subjectmatter. Query tabel dikelompokkan menjadi tiga kelompok berdasarkan kompleksitasnya, yaitu :

a. Simple Query : query tabel dengan join 2 variabel
    a. Penduduk menurut Kelompok Umur dan Hubungan dengan KRT
    b. Penduduk menurut Kelompok Umur dan Agama
    c. Penduduk menurut Provinsi dan Agama
    d. Penduduk menurut Kelompok Umur dan Kewarganegaraan
    e. Penduduk menurut Provinsi dan Kewarganegaraan
    f. Penduduk menurut Provinsi dan Suku Bangsa
b. Complex Query : query tabel dengan join 2 variabel dan perhitungan yang kompleks
    a. Penduduk Berumur 10 Tahun ke Atas menurut Kelompok Umur dan Status Perkawinan
    b. Penduduk berumur 10 Tahun ke Atas menurut Provinsi dan Status Perkawinan
    c. Penduduk WNI menurut kelompok Umur dan Suku Bangsa
    d. Penduduk WNA menurut Kelompok Umur dan Kewarganegaraan
    e. Penduduk WNA menurut Provinsi dan Kewarganegaraan
c. Heavy Query : query tabel dengan join lebih dari 2 variabel
    a. Penduduk menurut Umur Tunggal, Daerah Perkotaan/Pedesaan, dan Jenis Kelamin
    b. Penduduk menurut Kelompok Umur, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin
    c. Penduduk menurut Provinsi, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin

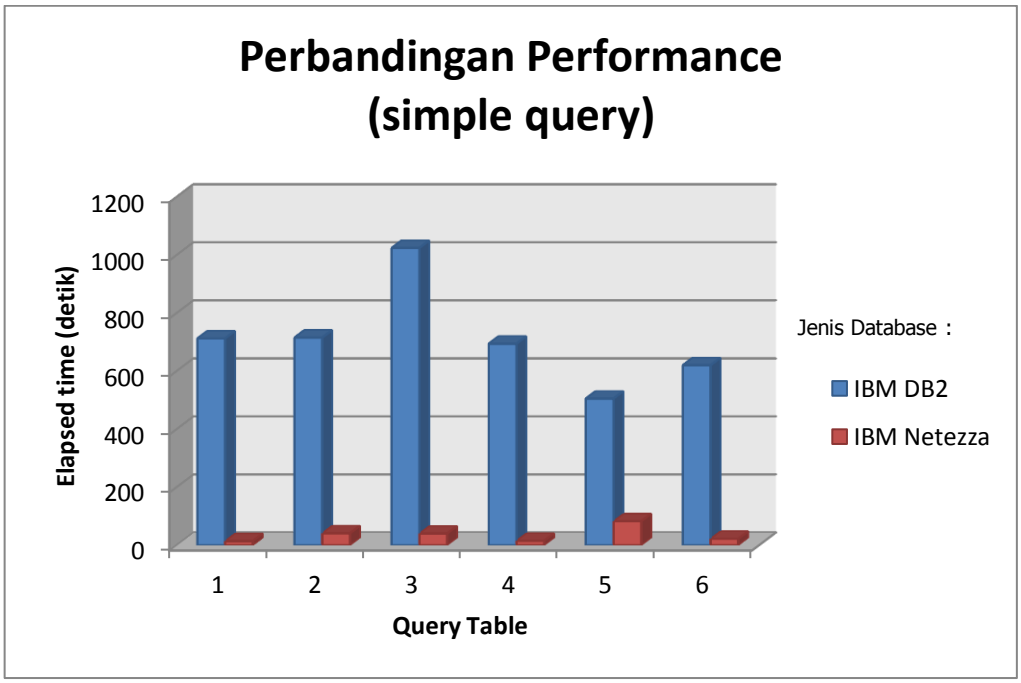
Query untuk setiap tabel POC dapat dilihat pada Lampiran 4.

Komparasi performance untuk data Sensus Penduduk 2010 juga dilakukan berdasarkan *elapsed time* eksekusi tiap-tiap query tabel. Hasilnya ditunjukkan pada tabel 8.

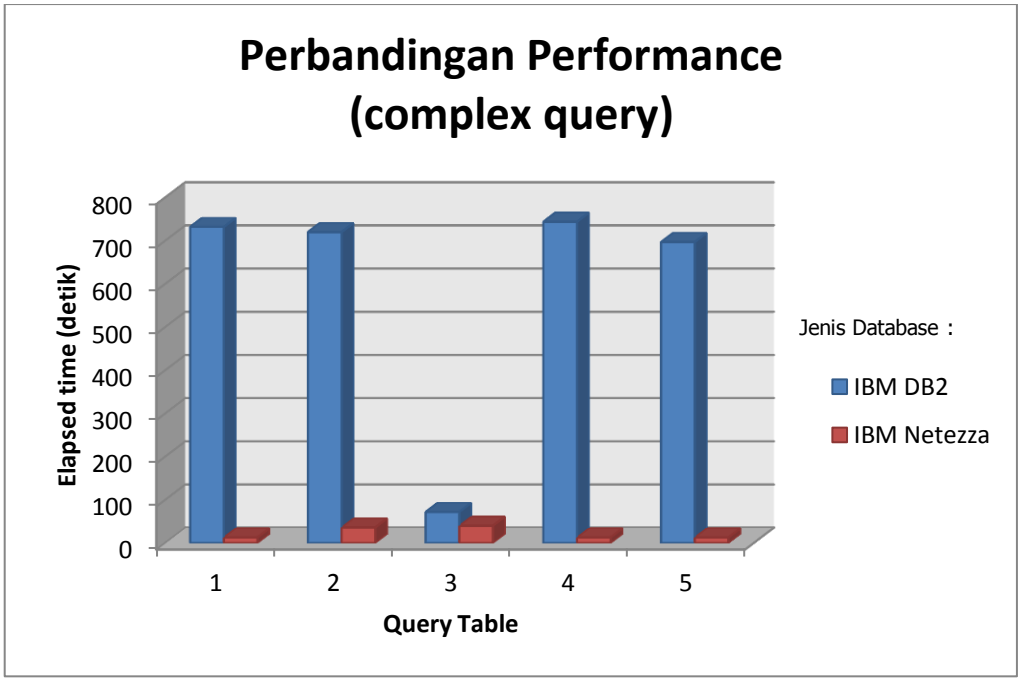
No	Nama Tabel	Tanpa Normalisasi HW				Dengan Normalisasi HW			
		DB2		Netezza		DB2		Netezza	
		1	**2**	**1**	**2**	**1**	**2**	**1**	**2**
Simple Query									
1	Penduduk menurut Kelompok Umur dan Hubungan dengan KRT	12′ 52″	6″	14"	2"	12′ 52″	6″	28"	2"
2	Penduduk menurut Kelompok Umur dan Agama	11′ 55″	5″	41"	2"	11′ 55″	5″	1′ 22"	2"
3	Penduduk menurut Provinsi dan Agama	17' 4"	6"	40"	2"	17' 4"	6"	1′ 20"	2"
4	Penduduk menurut Kelompok Umur dan Kewarganegaraan	11′ 34″	4″	15"	3"	11′ 34″	4″	30"	3"
5	Penduduk menurut Provinsi dan Kewarganegaraan	8′ 25″	5″	1' 22"	20"	8′ 25″	5″	2' 44"	20"
6	Penduduk menurut Provinsi dan Suku Bangsa	10′ 20″	14″	22"	15"	10′ 20″	14″	44"	15"
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**29.08**				**14.9**			
Complex Query									
1	Penduduk Berumur 10 Tahun ke Atas menurut Kelompok Umur dan Status Perkawinan	12′ 12″	7″	13"	2"	12′ 12″	7″	26"	2"
2	Penduduk berumur 10 Tahun ke Atas menurut Provinsi dan Status Perkawinan	12′	7″	36"	2"	12′	7″	1′ 2"	2"
3	Penduduk WNI menurut kelompok Umur dan Suku Bangsa	1′ 12″	15″	40"	26"	1′ 12″	15″	1′ 20"	26"
4	Penduduk WNA menurut Kelompok Umur dan Kewarganegaraan	12′ 24″	7″	13"	2"	12′ 24″	7″	26"	2"
5	Penduduk WNA menurut Provinsi dan Kewarganegaraan	11′ 37″	3″	13"	2"	11′ 37″	3″	26"	2"
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**37.79**				**19.22**			
Heavy Query									
1	Penduduk menurut Umur Tunggal, Daerah Perkotaan/Pedesaan, dan Jenis Kelamin	12'	2″	38"	9"	12'	2″	1′ 16"	9"

2	Penduduk menurut Kelompok Umur, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin	11' 54"	5″	1' 16"	12"	11' 54"	5″	2' 32"	12"
3	Penduduk menurut Provinsi, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin	11' 50"	2″	34"	7"	11' 50"	2″	1′ 8"	7"
Rata-rata Rasio Kecepatan Netezza terhadap DB2		**16.41**				**8.20**			

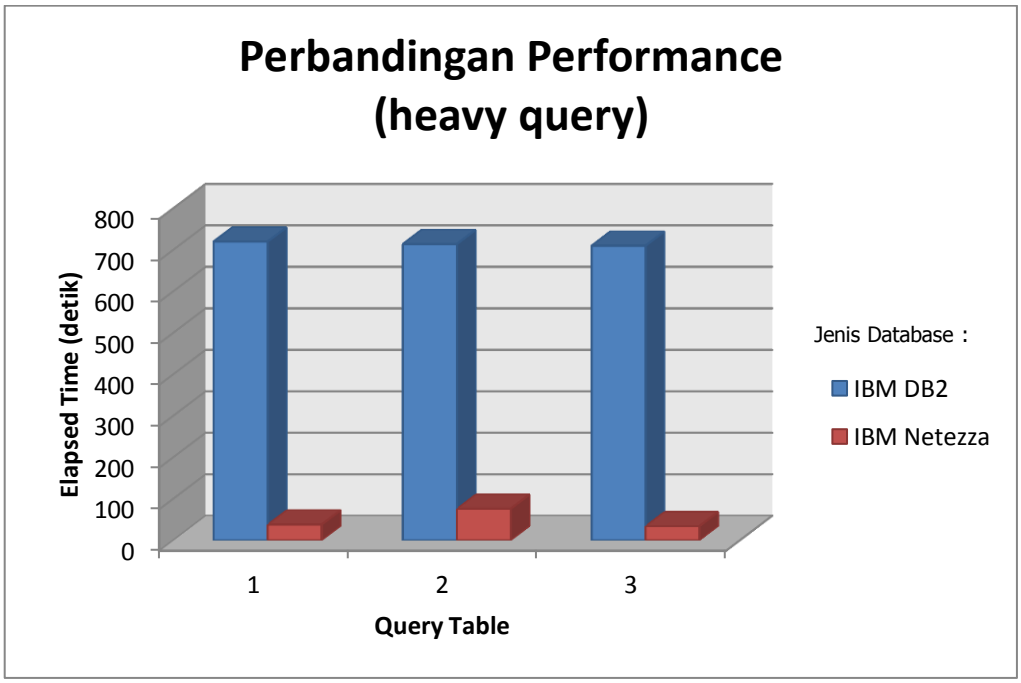
Tabel 8. Hasil Komparasi Performance Sensus Penduduk 2010



Gambar 8. Perbandingan Performance Simple Query – Sensus Penduduk 2010



Gambar 9. Perbandingan Performance Simple Query – Sensus Penduduk 2010



Gambar 10. Perbandingan Performance Heavy Query – Sensus Penduduk 2010

5. KESIMPULAN

Kesimpulan yang diperoleh dari kegiatan POC IBM Netezza sebagai berikut :

a. Kapasitas data yang ditampung oleh IBM Netezza lebih besar karena terdapat proses compression data.
b. Performance IBM Netezza lebih baik dari IBM DB2. Hal ini terlihat dari elapsed time yang dibutuhkan untuk eksekusi query tabel lebih singkat daripada IBM DB2 (komparasi menggunakan query tabel yang sama) dan rata-rata rasio kecepatan eksekusi yang tinggi.
c. Dalam hal managing database, IBM Netezza lebih mudah dilakukan karena hampir tidak memerlukan pemeliharaan (tuning).
d. Proses loading data (migrasi data) di IBM Netezza yang mudah dan lebih cepat daripada IBM DB2.

LAMPIRAN

1. Struktur Data Ekspor Impor

1) Fact Table Structure

No	Field Name	Data Type	Description
1	FACT_ID	BIGINT	-
2	PS_ID	BIGINT	Processing Stage ID
3	CR_ID	BIGINT	Change Reason ID
4	CM_ID	BIGINT	Change Modul ID
5	SAMPLING_ID	BIGINT	Sampling ID
6	ISIC_ID	BIGINT	International Standard Industrial Classification ID
7	SITC_ID	BIGINT	Standard International Trade Classification ID
8	COM_ID	BIGINT	Commodity ID
9	BEC_ID	BIGINT	Broad Economic Categories ID
10	DS_ID	BIGINT	Data Source ID
11	DI_ID	BIGINT	Data Item ID
12	PER_ID	BIGINT	Period ID
13	HS_ID	BIGINT	Harmonized System ID
14	COUNTRY_ID	BIGINT	Country ID
15	POD_ID	BIGINT	Port of Destination ID
16	POF_ID	BIGINT	Port of Freight ID
17	REG_ID	BIGINT	Region ID
18	KPBC_ID	BIGINT	Custom Office ID
19	DOCUMENT_ID	VARCHAR	Document ID
20	LATEST	VARCHAR	-
21	DELETED	VARCHAR	-
22	OBSERVATION	DOUBLE	Observation Value of Data Item (Numeric)
23	OBSERVATION_TEXT	VARCHAR	Observation Value of Data Item (Non Numeric)
24	COUNT_TEXT	INTEGER	'1' for Non Numeric Observation
25	START_DATE	DATE	-
26	END_DATE	DATE	-

2) Dimension Table Structure

a) SITC

No	Field Name	Data Type	Description
1	SITC_ID	BIGINT	Standard International Trade Classification ID
2	SITC_CODE	VARCHAR	SITC Code
3	SITC_DESC	VARCHAR	-

4	SITC_DESC_ENG	VARCHAR	-
5	SITC4	VARCHAR	SITC 4 Digits Code
6	SITC4_DESC	VARCHAR	-
7	SITC4_DESC_ENG	VARCHAR	-
8	SITC3	VARCHAR	-
9	SITC3_DESC	VARCHAR	-
10	SITC3_DESC_ENG	VARCHAR	-
11	SITC2	VARCHAR	-
12	SITC2_DESC	VARCHAR	-
13	SITC2_DESC_ENG	VARCHAR	-
14	SITC1	VARCHAR	-
15	SITC1_DESC	VARCHAR	-
16	SITC1_DESC_ENG	VARCHAR	-
17	SITC_START_DATE	DATE	-
18	SITC_END_DATE	DATE	-

b) COMMODITY

No	Field Name	Data Type	Description
1	COM_ID	BIGINT	Commodity ID
2	COM_CODE	VARCHAR	Commodity Code
3	COM_DESC	VARCHAR	-
4	COM_DESC_ENG	VARCHAR	-
5	COMGRP	VARCHAR	Commodity Group Code
6	COMGRP_DESC	VARCHAR	-
7	COMGRP_DESC_ENG	VARCHAR	-
8	SECTOR	VARCHAR	Sector Code
9	SECTOR_DESC	VARCHAR	-
10	SECTOR_DESC_ENG	VARCHAR	-
11	OILGRP	VARCHAR	Oil Group Code
12	OILGRP_DESC	VARCHAR	-
13	OILGRP_DESC_ENG	VARCHAR	-
14	COM_START_DATE	DATE	-
15	COM_END_DATE	DATE	-

c) BEC

No	Field Name	Data Type	Description
1	BEC_ID	BIGINT	Broad Economic Categories ID
2	BEC_CODE	VARCHAR	BEC Code
3	BEC_DESC	VARCHAR	-
4	BEC_DESC_ENG	VARCHAR	-
5	BEC2	VARCHAR	BEC 2 Digits Code
6	BEC2_DESC	VARCHAR	-

7	BEC2_DESC_ENG	VARCHAR	-
8	BEC1	VARCHAR	-
9	BEC1_DESC	VARCHAR	-
10	BEC1_DESC_ENG	VARCHAR	-
11	BEC_START_DATE	DATE	-
12	BEC_END_DATE	DATE	-

d) HS

No	Field Name	Data Type	Description
1	HS_ID	BIGINT	Harmonized System ID
2	HS_CODE	VARCHAR	HS Code
3	HS_DESC	VARCHAR	-
4	HS_DESC_ENG	VARCHAR	-
5	HS_DIGIT	VARCHAR	HS Number of Digit
6	HS_YEAR	VARCHAR	HS Version
7	HS6	VARCHAR	HS 6 Digits Code
8	HS6_DESC	VARCHAR	-
9	HS6_DESC_ENG	VARCHAR	-
10	HS4	VARCHAR	-
11	HS4_DESC	VARCHAR	-
12	HS4_DESC_ENG	VARCHAR	-
13	HS2	VARCHAR	-
14	HS2_DESC	VARCHAR	-
15	HS2_DESC_ENG	VARCHAR	-
16	HS1	VARCHAR	-
17	HS1_DESC	VARCHAR	-
18	HS1_DESC_ENG	VARCHAR	-
19	SECTION_CODE	VARCHAR	Section Code
20	SECTION_CODE_ROM	VARCHAR	-
21	SECTION_DESC	VARCHAR	-
22	SECTION_DESC_ENG	VARCHAR	-
23	HS_START_DATE	DATE	-
24	HS_END_DATE	DATE	-

e) ISIC

No	Field Name	Data Type	Description
1	ISIC_ID	BIGINT	International Standard Industrial Classification ID
2	ISIC_CODE	VARCHAR	ISIC Code
3	ISIC_DESC	VARCHAR	-
4	ISIC_DESC_ENG	VARCHAR	-
5	ISIC3	VARCHAR	-
6	ISIC3_DESC	VARCHAR	-

7	ISIC3_DESC_ENG	VARCHAR	-
8	ISIC2	VARCHAR	-
9	ISIC2_DESC	VARCHAR	-
10	ISIC2_DESC_ENG	VARCHAR	-
11	ISIC1	VARCHAR	-
12	ISIC1_DESC	VARCHAR	-
13	ISIC1_DESC_ENG	VARCHAR	-
14	ISIC_START_DATE	DATE	-
15	ISIC_END_DATE	DATE	-

f) KPBC

No	Field Name	Data Type	Description
1	KPBC_ID	BIGINT	Kantor Pelayanan Bea Cukai (Custom Office) ID
2	KPBC_CODE	VARCHAR	-
3	KPBC_DESC	VARCHAR	-
4	PROV_CODE	VARCHAR	-
5	PROV_DESC	VARCHAR	-
6	KWSN_BERIKAT_CODE	VARCHAR	-
7	KWSN_BERIKAT_DESC	VARCHAR	-
8	KPBC_START_DATE	DATE	-
9	KPBC_END_DATE	DATE	-

g) PORT

No	Field Name	Data Type	Description
1	PORT_ID	BIGINT	Port ID
2	PORT_CODE	VARCHAR	-
3	PORT_DESC	VARCHAR	-
4	PROV_CODE	VARCHAR	Province Code
5	PROV_DESC	VARCHAR	-
6	ISLAND_CODE	VARCHAR	-
7	ISLAND_DESC	VARCHAR	-
8	PORT_START_DATE	DATE	-
9	PORT_END_DATE	DATE	-

h) REGION

No	Field Name	Data Type	Description
1	REG_ID	BIGINT	Region ID
2	MUN_CODE	VARCHAR	Municipality Code
3	MUN_DESC	VARCHAR	-
4	PROV_CODE	VARCHAR	Province Code
5	PROV_DESC	VARCHAR	-

6	ISLAND_CODE	VARCHAR	-
7	ISLAND_DESC	VARCHAR	-
8	REG_START_DATE	DATE	-
9	REG_END_DATE	DATE	-

i) COUNTRY

No	Field Name	Data Type	Description
1	COUNTRY_ID	BIGINT	Country ID
2	COUNTRY_CODE	VARCHAR	-
3	COUNTRY_DESC	VARCHAR	-
4	COUNTRY_REGION_CODE	VARCHAR	Country Region Code
5	COUNTRY_REGION_DESC	VARCHAR	-
6	COUNTRY_START_DATE	DATE	-
7	COUNTRY_END_DATE	DATE	-

j) DATA_ITEM

No	Field Name	Data Type	Description
1	DI_ID	BIGINT	Data Item ID
2	DI_NAME	VARCHAR	eg: FOB, CIF, Nett Weight, etc
3	DI_UNIT	VARCHAR	eg: US Dollar, Kg, etc
4	DI_DESC	VARCHAR	-
5	DI_START_DATE	DATE	-
6	DI_END_DATE	DATE	-

k) DATA_SOURCE

No	Field Name	Data Type	Description
1	DS_ID	BIGINT	Data Source ID
2	DS_COLLECTION_CODE	VARCHAR	eg: E for Export, I for Import
3	DS_COLLECTION_NAME	VARCHAR	-
4	DS_MONTH	VARCHAR	-
5	DS_YEAR	VARCHAR	-
6	DS_CYCLE_ID	VARCHAR	-

l) PERIOD

No	Field Name	Data Type	Description
1	PER_ID	BIGINT	Period ID
2	PER_MONTH	VARCHAR	-
3	PER_MONTH_NAME	VARCHAR	-
4	PER_QUARTER	VARCHAR	-
5	PER_SEMESTER	VARCHAR	-
6	PER_YEAR	VARCHAR	-
7	PER_START_DATE	DATE	-

8	PER_END_DATE	DATE	-

m) CHANGE_MODUL

No	Field Name	Data Type	Description
1	CM_ID	BIGINT	Change Modul ID
2	CM_MODULE_NAME	VARCHAR	Modul/Apps Name for DW change
3	CM_MODULE_VERSION	VARCHAR	Modul Version
4	CM_MODULE_DESC	VARCHAR	-
5	CM_START_DATE	DATE	-
6	CM_END_DATE	DATE	-

n) CHANGE_REASON

No	Field Name	Data Type	Description
1	CR_ID	BIGINT	Change Reason ID
2	CR_DESC	VARCHAR	Change Reason Description
3	CR_MODE	VARCHAR	Tools for Record Change

o) SAMPLING

No	Field Name	Data Type	Description
1	SAMPLING_ID	BIGINT	Sampling ID
2	SAMPLING_TYPE	VARCHAR	Sampling Type
3	CYCLE_ID	VARCHAR	-
4	FRAME_COUNT	BIGINT	-
5	SAMPLE_COUNT	BIGINT	-
6	SIZE_CODE	BIGINT	-
7	NOTE	VARCHAR	-

p) PROCESSING_STAGE

No	Field Name	Data Type	Description
1	PS_ID	BIGINT	Processing Stage ID
2	PS_CODE	VARCHAR	Processing Stage Code
3	PS_LABEL	VARCHAR	Processing Stage Description

2. Query Tabel POC Data Ekspor Impor

No	Ket	Query
1	Export of Indonesia By Country of Destination --> Annual Table (jan-march 2007 and jan-march 2008)	select Query120.L_Country_Desc as levelkey, Query120.L_Nett_Weight as L_Nett_Weight, Query120.L_FOB as L_FOB, Query221.C_Nett_Weight as C_Nett_Weight, Query221.C_FOB as C_FOB from (select coalesce(D10.L_Country_Desc,D11.L_Country_Desc) as L_Country_Desc, D11.L_Nett_Weight as L_Nett_Weight, D10.L_FOB as L_FOB from (select COUNTRY.COUNTRY_DESC as L_Country_Desc, SUM(Ex_FOB.OBSERVATION) as L_FOB from DB2INST1.COUNTRY COUNTRY, (select FACT.PER_ID as PER_ID, FACT.COUNTRY_ID as COUNTRY_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '03') and (COUNTRY.COUNTRY_ID = Ex_FOB.COUNTRY_ID) and (PERIOD.PER_ID = Ex_FOB.PER_ID) group by COUNTRY.COUNTRY_DESC) as D10 full outer join (select COUNTRY.COUNTRY_DESC as L_Country_Desc, SUM(Ex_NettWeight.OBSERVATION) as L_Nett_Weight from COUNTRY COUNTRY, (select FACT.PER_ID as PER_ID, FACT.COUNTRY_ID as COUNTRY_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE

```
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
 ) as Ex_NettWeight,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COUNTRY.COUNTRY_ID = Ex_NettWeight.COUNTRY_ID) and
        (PERIOD.PER_ID = Ex_NettWeight.PER_ID)
group by
        COUNTRY.COUNTRY_DESC
 ) as D11
         on (D10.L_Country_Desc = D11.L_Country_Desc)
) as Query120,
     (select
        D13.C_FOB  as  C_FOB,
        D14.C_Nett_Weight  as  C_Nett_Weight,
        coalesce(D13.C_Country_Desc,D14.C_Country_Desc)  as
C_Country_Desc
    from
        (select
         COUNTRY.COUNTRY_DESC  as  C_Country_Desc,
         SUM(Ex_FOB18.OBSERVATION )  as  C_FOB
from
         COUNTRY COUNTRY,
 (select
  FACT.PER_ID  as  PER_ID,
  FACT.COUNTRY_ID  as  COUNTRY_ID,
  FACT.OBSERVATION  as  OBSERVATION
 from
         FACT FACT,
         DATA_SOURCE DATA_SOURCE
where
         (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
         (FACT.LATEST = 'Y') and
         (FACT.DI_ID = '1') and
         (DATA_SOURCE.DS_ID = FACT.DS_ID)
 ) as Ex_FOB18,
         PERIOD PERIOD
where
         (PERIOD.PER_YEAR = '2008') and
         (PERIOD.PER_MONTH between '01' and '03') and
         (COUNTRY.COUNTRY_ID = Ex_FOB18.COUNTRY_ID) and
         (PERIOD.PER_ID = Ex_FOB18.PER_ID)
group by
         COUNTRY.COUNTRY_DESC
) as D13
         full outer join
        (select
         COUNTRY.COUNTRY_DESC  as  C_Country_Desc,
         SUM(Ex_NettWeight19.OBSERVATION)  as  C_Nett_Weight
       from
         COUNTRY COUNTRY,
         (select
```

```
            FACT.PER_ID  as  PER_ID,
           FACT.COUNTRY_ID  as  COUNTRY_ID,
            FACT.OBSERVATION  as  OBSERVATION
      from
           FACT FACT,
           DATA_SOURCE DATA_SOURCE
 where
           (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
           (FACT.LATEST = 'Y') and
           (FACT.DI_ID = '3') and
           (DATA_SOURCE.DS_ID = FACT.DS_ID)
 ) as Ex_NettWeight19,
           PERIOD PERIOD
         where
           (PERIOD.PER_YEAR = '2008') and
           (PERIOD.PER_MONTH between '01' and '03') and
           (COUNTRY.COUNTRY_ID = Ex_NettWeight19.COUNTRY_ID)
and
           (PERIOD.PER_ID = Ex_NettWeight19.PER_ID)
          group by
           COUNTRY.COUNTRY_DESC
           ) as D14
           on (D13.C_Country_Desc = D14.C_Country_Desc)
    ) as Query221
 where
      (Query120.L_Country_Desc = Query221.C_Country_Desc)
```

2	The Summary of Indonesian Exports --> Buletin Tabel of Export (jan-march 2007 and jan-march 2008)	select Query220.C_Oil_Group_Desc as levelkey, Query220.C_Sector_Desc as level0key, Query121.L_Nett_Weight as L_Nett_Weight, Query220.C_Nett_Weight as C_Nett_Weight, Query121.L_FOB as L_FOB, Query220.C_FOB as C_FOB, (Query220.C_FOB / SUM(Query220.C_FOB)) as peranan_thd_total_fob from (select D13.C_FOB as C_FOB, D14.C_Nett_Weight as C_Nett_Weight, coalesce(D13.C_Sector_Desc,D14.C_Sector_Desc) as C_Sector_Desc, coalesce(D13.C_Oil_Group_Desc,D14.C_Oil_Group_Desc) as C_Oil_Group_Desc from (select COMMODITY.SECTOR_DESC as C_Sector_Desc, COMMODITY.OILGRP_DESC as C_Oil_Group_Desc, SUM(Ex_FOB18.OBSERVATION) as C_FOB from COMMODITY COMMODITY, (select FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB18, PERIOD PERIOD where (PERIOD.PER_YEAR = '2008') and (PERIOD.PER_MONTH between '01' and '03') and (COMMODITY.COM_ID = Ex_FOB18.COM_ID) and (PERIOD.PER_ID = Ex_FOB18.PER_ID) group by COMMODITY.SECTOR_DESC, COMMODITY.OILGRP_DESC) as D13 full outer join (select COMMODITY.SECTOR_DESC as C_Sector_Desc, COMMODITY.OILGRP_DESC as C_Oil_Group_Desc, SUM(Ex_NettWeight19.OBSERVATION) as C_Nett_Weight from COMMODITY COMMODITY, (select FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID,

```
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight19,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2008') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COMMODITY.COM_ID = Ex_NettWeight19.COM_ID) and
        (PERIOD.PER_ID = Ex_NettWeight19.PER_ID)
group by
        COMMODITY.SECTOR_DESC,
        COMMODITY.OILGRP_DESC
 ) as D14
on ((D13.C_Sector_Desc = D14.C_Sector_Desc)
 and (D13.C_Oil_Group_Desc = D14.C_Oil_Group_Desc))
) as Query220,
     (select
        coalesce(D10.L_Oil_Group_Desc,D11.L_Oil_Group_Desc)  as
L_Oil_Group_Desc,
       coalesce(D10.L_Sector_Desc,D11.L_Sector_Desc)  as
L_Sector_Desc,
D11.L_Nett_Weight  as  L_Nett_Weight,
        D10.L_FOB  as  L_FOB
  from
   (select
        COMMODITY.OILGRP_DESC  as  L_Oil_Group_Desc,
        COMMODITY.SECTOR_DESC  as  L_Sector_Desc,
        SUM(Ex_FOB.OBSERVATION  )  as  L_FOB
from
        COMMODITY COMMODITY,
 (select
        FACT.COM_ID  as  COM_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '1') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as  Ex_FOB,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COMMODITY.COM_ID = Ex_FOB.COM_ID) and
        (PERIOD.PER_ID = Ex_FOB.PER_ID)
group by
```

```
            COMMODITY.OILGRP_DESC,
            COMMODITY.SECTOR_DESC
 ) as D10
full outer join
       (select
            COMMODITY.OILGRP_DESC  as  L_Oil_Group_Desc,
            COMMODITY.SECTOR_DESC  as  L_Sector_Desc,
            SUM(Ex_NettWeight.OBSERVATION )  as  L_Nett_Weight
 from
            COMMODITY COMMODITY,
  (select
            FACT.COM_ID  as  COM_ID,
            FACT.PER_ID  as  PER_ID,
            FACT.OBSERVATION  as  OBSERVATION
 from
            FACT FACT,
            DATA_SOURCE DATA_SOURCE
 where
            (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
            (FACT.LATEST = 'Y') and
            (FACT.DI_ID = '3') and
            (DATA_SOURCE.DS_ID = FACT.DS_ID)
  ) as Ex_NettWeight,
            PERIOD PERIOD
 where
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH between '01' and '03') and
            (COMMODITY.COM_ID = Ex_NettWeight.COM_ID) and
            (PERIOD.PER_ID = Ex_NettWeight.PER_ID)
 group by
            COMMODITY.OILGRP_DESC,
            COMMODITY.SECTOR_DESC
 ) as D11
    on ((D10.L_Oil_Group_Desc = D11.L_Oil_Group_Desc)
and (D10.L_Sector_Desc = D11.L_Sector_Desc))
    ) as Query121
 where
       ((Query121.L_Oil_Group_Desc = Query220.C_Oil_Group_Desc) and
(Query121.L_Sector_Desc = Query220.C_Sector_Desc))
group by
  Query220.C_Oil_Group_Desc ,
 Query220.C_Sector_Desc  ,
 Query121.L_Nett_Weight ,
 Query220.C_Nett_Weight ,
 Query121.L_FOB ,
 Query220.C_FOB
```

3	Export By Commodity (HS) and Country of Destination --> Annual Table (jan-may 2007)	select coalesce(D8.HS_Codekey,D9.HS_Codekey,D10.HS_Codekey) as HS_Codekey, coalesce(D8.Commodity_Desckey,D9.Commodity_Desckey,D10.Commodity_Desckey) as Commodity_Desckey, coalesce(D8.SITC_Codekey,D9.SITC_Codekey,D10.SITC_Codekey) as SITC_Codekey, coalesce(D8.Country_Codekey,D9.Country_Codekey,D10.Country_Codekey) as Country_Codekey, coalesce(D8.Country_Desckey,D9.Country_Desckey,D10.Country_Desckey) as Country_Desckey, D10.Quantity as Quantity, D9.Nett_Weight as Nett_Weight, D8.FOB as FOB from (select HS.HS_CODE as HS_Codekey, COMMODITY.COM_DESC as Commodity_Desckey, SITC.SITC_CODE as SITC_Codekey, COUNTRY.COUNTRY_CODE as Country_Codekey, COUNTRY.COUNTRY_DESC as Country_Desckey, SUM(Ex_FOB.OBSERVATION) as FOB from HS HS, COMMODITY COMMODITY, SITC SITC, COUNTRY COUNTRY, (select FACT.SITC_ID as SITC_ID, FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID, FACT.HS_ID as HS_ID, FACT.COUNTRY_ID as COUNTRY_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '05') and (HS.HS_ID = Ex_FOB.HS_ID) and (COMMODITY.COM_ID = Ex_FOB.COM_ID) and (COUNTRY.COUNTRY_ID = Ex_FOB.COUNTRY_ID) and (PERIOD.PER_ID = Ex_FOB.PER_ID) and (SITC.SITC_ID = Ex_FOB.SITC_ID) group by HS.HS_CODE,

```
        COMMODITY.COM_DESC,
        SITC.SITC_CODE,
        COUNTRY.COUNTRY_CODE,
        COUNTRY.COUNTRY_DESC
) as D8
    full outer join
  (select
        HS.HS_CODE  as  HS_Codekey,
        COMMODITY.COM_DESC  as  Commodity_Desckey,
        SITC.SITC_CODE  as  SITC_Codekey,
        COUNTRY.COUNTRY_CODE  as  Country_Codekey,
        COUNTRY.COUNTRY_DESC  as  Country_Desckey,
        SUM(Ex_NettWeight.OBSERVATION  )  as  Nett_Weight
from
        HS HS,
        COMMODITY COMMODITY,
        SITC SITC,
        COUNTRY COUNTRY,
(select
        FACT.SITC_ID  as  SITC_ID,
        FACT.COM_ID  as  COM_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.HS_ID  as  HS_ID,
        FACT.COUNTRY_ID  as  COUNTRY_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '05') and
        (HS.HS_ID = Ex_NettWeight.HS_ID) and
        (COMMODITY.COM_ID = Ex_NettWeight.COM_ID) and
        (COUNTRY.COUNTRY_ID = Ex_NettWeight.COUNTRY_ID) and
        (PERIOD.PER_ID = Ex_NettWeight.PER_ID) and
        (SITC.SITC_ID = Ex_NettWeight.SITC_ID)
group by
        HS.HS_CODE,
        COMMODITY.COM_DESC,
        SITC.SITC_CODE,
        COUNTRY.COUNTRY_CODE,
        COUNTRY.COUNTRY_DESC
) as D9
    on (((((D8.HS_Codekey = D9.HS_Codekey) and
(D8.Commodity_Desckey = D9.Commodity_Desckey)) and
(D8.SITC_Codekey = D9.SITC_Codekey)) and (D8.Country_Codekey =
D9.Country_Codekey)) and (D8.Country_Desckey =
D9.Country_Desckey))
    full outer join
    (select
```

```
      HS.HS_CODE as HS_Codekey,
          COMMODITY.COM_DESC as Commodity_Desckey,
          SITC.SITC_CODE as SITC_Codekey,
          COUNTRY.COUNTRY_CODE as Country_Codekey,
          COUNTRY.COUNTRY_DESC as Country_Desckey,
          SUM(Ex_Quantity.OBSERVATION ) as Quantity
from
          HS HS,
          COMMODITY COMMODITY,
          SITC SITC,
          COUNTRY COUNTRY,
(select
          FACT.SITC_ID as SITC_ID,
          FACT.COM_ID as COM_ID,
          FACT.PER_ID as PER_ID,
          FACT.HS_ID as HS_ID,
          FACT.COUNTRY_ID as COUNTRY_ID,
          FACT.OBSERVATION as OBSERVATION
from
          FACT FACT,
          DATA_SOURCE DATA_SOURCE
where
          (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
          (FACT.LATEST = 'Y') and
          (FACT.DI_ID = '5') and
          (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_Quantity,
          PERIOD PERIOD
where
          (PERIOD.PER_YEAR = '2007') and
          (PERIOD.PER_MONTH between '01' and '05') and
          (HS.HS_ID = Ex_Quantity.HS_ID) and
          (COMMODITY.COM_ID = Ex_Quantity.COM_ID) and
          (COUNTRY.COUNTRY_ID = Ex_Quantity.COUNTRY_ID) and
          (PERIOD.PER_ID = Ex_Quantity.PER_ID) and
          (SITC.SITC_ID = Ex_Quantity.SITC_ID)
group by
          HS.HS_CODE,
          COMMODITY.COM_DESC,
          SITC.SITC_CODE,
          COUNTRY.COUNTRY_CODE,
          COUNTRY.COUNTRY_DESC
) as D10
      on (((((coalesce(D8.HS_Codekey,D9.HS_Codekey) =
D10.HS_Codekey) and
(coalesce(D8.Commodity_Desckey,D9.Commodity_Desckey) =
D10.Commodity_Desckey)) and
(coalesce(D8.SITC_Codekey,D9.SITC_Codekey) = D10.SITC_Codekey))
and (coalesce(D8.Country_Codekey,D9.Country_Codekey) =
D10.Country_Codekey)) and
(coalesce(D8.Country_Desckey,D9.Country_Desckey) =
D10.Country_Desckey))
```

4	Export of Indonesia By Province and Port of Loading --> Monthly Table (march 2007 and jan-march 2007)	select Query220.Province_Desc as levelkey, Query220.POF_Code as level0key, Query220.POF_Desc as level1key, Query121.C_Nett_Weight as C_Nett_Weight, Query220.Cum_Nett_Weight as Cum_Nett_Weight, Query121.C_FOB as C_FOB, Query220.Cum_FOB as Cum_FOB from (select coalesce(D13.POF_Code,D14.POF_Code) as POF_Code, coalesce(D13.POF_Desc,D14.POF_Desc) as POF_Desc, coalesce(D13.Province_Desc,D14.Province_Desc) as Province_Desc, D13.Cum_FOB as Cum_FOB, D14.Cum_Nett_Weight as Cum_Nett_Weight from (select POF.PORT_CODE as POF_Code, POF.PORT_DESC as POF_Desc, POF.PROV_DESC as Province_Desc, SUM(Ex_FOB18.OBSERVATION) as Cum_FOB from PORT POF, (select FACT.PER_ID as PER_ID, FACT.POF_ID as POF_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB18, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '03') and (POF.PORT_ID = Ex_FOB18.POF_ID) and (PERIOD.PER_ID = Ex_FOB18.PER_ID) group by POF.PORT_CODE, POF.PORT_DESC, POF.PROV_DESC) as D13 full outer join (select POF.PORT_CODE as POF_Code, POF.PORT_DESC as POF_Desc, POF.PROV_DESC as Province_Desc, SUM(Ex_NettWeight19.OBSERVATION) as Cum_Nett_Weight from PORT POF, (select

```
            FACT.PER_ID  as  PER_ID,
            FACT.POF_ID  as  POF_ID,
            FACT.OBSERVATION  as  OBSERVATION
from
            FACT FACT,
            DATA_SOURCE DATA_SOURCE
where
            (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
            (FACT.LATEST = 'Y') and
            (FACT.DI_ID = '3') and
            (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight19,
            PERIOD PERIOD
      where
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH between '01' and '03') and
            (POF.PORT_ID = Ex_NettWeight19.POF_ID) and
            (PERIOD.PER_ID = Ex_NettWeight19.PER_ID)
group by
            POF.PORT_CODE,
            POF.PORT_DESC,
            POF.PROV_DESC
) as D14
on (((D13.POF_Code = D14.POF_Code) and (D13.POF_Desc =
D14.POF_Desc))
 and (D13.Province_Desc = D14.Province_Desc))
) as Query220,
  (select
            coalesce(D10.POF_Code,D11.POF_Code)  as  POF_Code,
            coalesce(D10.POF_Desc,D11.POF_Desc)  as  POF_Desc,
            coalesce(D10.Province_Desc,D11.Province_Desc)  as
Province_Desc,
            D11.C_Nett_Weight  as  C_Nett_Weight,
            D10.C_FOB  as  C_FOB
from
  (select
            POF.PORT_CODE  as  POF_Code,
            POF.PORT_DESC  as  POF_Desc,
            POF.PROV_DESC  as  Province_Desc,
            SUM(Ex_FOB.OBSERVATION  )  as  C_FOB
from
            PORT POF,
(select
            FACT.PER_ID  as  PER_ID,
            FACT.POF_ID  as  POF_ID,
            FACT.OBSERVATION  as  OBSERVATION
from
            FACT FACT,
            DATA_SOURCE DATA_SOURCE
where
            (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
            (FACT.LATEST = 'Y') and
            (FACT.DI_ID = '1') and
            (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB,
            PERIOD PERIOD
where
```

```
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH = '03') and
            (POF.PORT_ID = Ex_FOB.POF_ID) and
            (PERIOD.PER_ID = Ex_FOB.PER_ID)
 group by
            POF.PORT_CODE,
            POF.PORT_DESC,
            POF.PROV_DESC
 ) as D10
             full outer join
 (select
            POF.PORT_CODE  as  POF_Code,
            POF.PORT_DESC  as  POF_Desc,
            POF.PROV_DESC  as  Province_Desc,
            SUM(Ex_NettWeight.OBSERVATION  )  as  C_Nett_Weight
 from
            PORT POF,
 (select
            FACT.PER_ID  as  PER_ID,
            FACT.POF_ID  as  POF_ID,
            FACT.OBSERVATION  as  OBSERVATION
 from
            FACT FACT,
            DATA_SOURCE DATA_SOURCE
 where
            (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
            (FACT.LATEST = 'Y') and
            (FACT.DI_ID = '3') and
            (DATA_SOURCE.DS_ID = FACT.DS_ID)
 ) as Ex_NettWeight,
            PERIOD PERIOD
 where
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH = '03') and
            (POF.PORT_ID = Ex_NettWeight.POF_ID) and
            (PERIOD.PER_ID = Ex_NettWeight.PER_ID)
 group by
            POF.PORT_CODE,
            POF.PORT_DESC,
            POF.PROV_DESC
 ) as D11
 on (((D10.POF_Code = D11.POF_Code) and (D10.POF_Desc =
D11.POF_Desc))
  and (D10.Province_Desc = D11.Province_Desc))
 )as Query121
 where
       (((Query121.POF_Code = Query220.POF_Code) and
(Query121.POF_Desc = Query220.POF_Desc)) and
(Query121.Province_Desc = Query220.Province_Desc))
```

5	Import By Commodity (HS) And Country Of Origin --> Annual Table (jan-may 2007)	select coalesce(D8.HS_Codekey,D9.HS_Codekey,D10.HS_Codekey) as HS_Codekey, coalesce(D8.HS_Desckey,D9.HS_Desckey,D10.HS_Desckey) as HS_Desckey, coalesce(D8.SITC_3_Digits_Codekey,D9.SITC_3_Digits_Codekey,D10.SITC_3_Digits_Codekey) as SITC_3_Digits_Codekey, coalesce(D8.Country_Codekey,D9.Country_Codekey,D10.Country_Codekey) as Country_Codekey, coalesce(D8.Country_Desckey,D9.Country_Desckey,D10.Country_Desckey) as Country_Desckey, D10.Quantity as Quantity, D9.Nett_Weight as Nett_Weight, D8.CIF as CIF from (select HS.HS_CODE as HS_Codekey, HS.HS_DESC_ENG as HS_Desckey, SITC.SITC3 as SITC_3_Digits_Codekey, COUNTRY.COUNTRY_CODE as Country_Codekey, COUNTRY.COUNTRY_DESC as Country_Desckey, SUM(Im_CIF.OBSERVATION) as CIF from HS HS, SITC SITC, COUNTRY COUNTRY, (select FACT.SITC_ID as SITC_ID, FACT.PER_ID as PER_ID, FACT.HS_ID as HS_ID, FACT.COUNTRY_ID as COUNTRY_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'I') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '2') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Im_CIF, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '05') and (HS.HS_ID = Im_CIF.HS_ID) and (COUNTRY.COUNTRY_ID = Im_CIF.COUNTRY_ID) and (PERIOD.PER_ID = Im_CIF.PER_ID) and (SITC.SITC_ID = Im_CIF.SITC_ID) group by HS.HS_CODE, HS.HS_DESC_ENG, SITC.SITC3, COUNTRY.COUNTRY_CODE,

```
        COUNTRY.COUNTRY_DESC
) as D8
      full outer join
     (select
        HS.HS_CODE  as  HS_Codekey,
        HS.HS_DESC_ENG  as  HS_Desckey,
        SITC.SITC3  as  SITC_3_Digits_Codekey,
 COUNTRY.COUNTRY_CODE  as  Country_Codekey,
        COUNTRY.COUNTRY_DESC  as  Country_Desckey,
        SUM(Im_NettWeight.OBSERVATION  )  as  Nett_Weight
from
        HS HS,
        SITC SITC,
        COUNTRY COUNTRY,
 (select
        FACT.SITC_ID  as  SITC_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.HS_ID  as  HS_ID,
        FACT.COUNTRY_ID  as  COUNTRY_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'I') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Im_NettWeight,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '05') and
        (HS.HS_ID = Im_NettWeight.HS_ID) and
        (COUNTRY.COUNTRY_ID = Im_NettWeight.COUNTRY_ID) and
        (PERIOD.PER_ID = Im_NettWeight.PER_ID) and
        (SITC.SITC_ID = Im_NettWeight.SITC_ID)
group by
        HS.HS_CODE,
        HS.HS_DESC_ENG,
        SITC.SITC3,
        COUNTRY.COUNTRY_CODE,
        COUNTRY.COUNTRY_DESC
) as D9
      on (((((D8.HS_Codekey = D9.HS_Codekey) and (D8.HS_Desckey
= D9.HS_Desckey)) and (D8.SITC_3_Digits_Codekey =
D9.SITC_3_Digits_Codekey)) and (D8.Country_Codekey =
D9.Country_Codekey)) and (D8.Country_Desckey =
D9.Country_Desckey))
     full outer join
    (select
        HS.HS_CODE  as  HS_Codekey,
        HS.HS_DESC_ENG  as  HS_Desckey,
        SITC.SITC3  as  SITC_3_Digits_Codekey,
        COUNTRY.COUNTRY_CODE  as  Country_Codekey,
        COUNTRY.COUNTRY_DESC  as  Country_Desckey,
        SUM(Im_Quantity.OBSERVATION  )  as  Quantity
```

```
from
        HS HS,
        SITC SITC,
        COUNTRY COUNTRY,
 (select
        FACT.SITC_ID  as  SITC_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.HS_ID  as  HS_ID,
        FACT.COUNTRY_ID  as  COUNTRY_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
    where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'I') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '5') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Im_Quantity,
   PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '05') and
        (HS.HS_ID = Im_Quantity.HS_ID) and
        (COUNTRY.COUNTRY_ID = Im_Quantity.COUNTRY_ID) and
        (PERIOD.PER_ID = Im_Quantity.PER_ID) and
        (SITC.SITC_ID = Im_Quantity.SITC_ID)
group by
        HS.HS_CODE,
        HS.HS_DESC_ENG,
        SITC.SITC3,
 COUNTRY.COUNTRY_CODE,
        COUNTRY.COUNTRY_DESC
) as D10
      on (((((coalesce(D8.HS_Codekey,D9.HS_Codekey) =
D10.HS_Codekey) and (coalesce(D8.HS_Desckey,D9.HS_Desckey) =
D10.HS_Desckey)) and
(coalesce(D8.SITC_3_Digits_Codekey,D9.SITC_3_Digits_Codekey) =
D10.SITC_3_Digits_Codekey)) and
(coalesce(D8.Country_Codekey,D9.Country_Codekey) =
D10.Country_Codekey)) and
(coalesce(D8.Country_Desckey,D9.Country_Desckey) =
D10.Country_Desckey))
```

6	Import By Province And Port Of Importation --> Monthly Table (march 2007 and jan-march 2007)	select Query220.Province_Desc as levelkey, Query220.POD_Code as level0key, Query220.POD_Desc as level1key, Query121.C_Nett_Weight as C_Nett_Weight, Query220.Cum_Nett_Weight as Cum_Nett_Weight, Query121.C_CIF as C_CIF, Query220.Cum_CIF as Cum_CIF from (select D13.Cum_CIF as Cum_CIF, D14.Cum_Nett_Weight as Cum_Nett_Weight, coalesce(D13.POD_Code,D14.POD_Code) as POD_Code, coalesce(D13.POD_Desc,D14.POD_Desc) as POD_Desc, coalesce(D13.Province_Desc,D14.Province_Desc) as Province_Desc from (select POD.PORT_CODE as POD_Code, POD.PORT_DESC as POD_Desc, POD.PROV_DESC as Province_Desc, SUM(Im_CIF18.OBSERVATION) as Cum_CIF from PORT POD, (select FACT.PER_ID as PER_ID, FACT.POD_ID as POD_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'I') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '2') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Im_CIF18, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '03') and (POD.PORT_ID = Im_CIF18.POD_ID) and (PERIOD.PER_ID = Im_CIF18.PER_ID) group by POD.PORT_CODE, POD.PORT_DESC, POD.PROV_DESC) as D13 full outer join (select POD.PORT_CODE as POD_Code, POD.PORT_DESC as POD_Desc, POD.PROV_DESC as Province_Desc, SUM(Im_NettWeight19.OBSERVATION) as Cum_Nett_Weight from PORT POD, (select

```
		FACT.PER_ID as PER_ID,
		FACT.POD_ID as POD_ID,
		FACT.OBSERVATION as OBSERVATION
from
		FACT FACT,
		DATA_SOURCE DATA_SOURCE
where
		(DATA_SOURCE.DS_COLLECTION_CODE = 'I') and
		(FACT.LATEST = 'Y') and
		(FACT.DI_ID = '3') and
		(DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Im_NettWeight19,
		PERIOD PERIOD
where
		(PERIOD.PER_YEAR = '2007') and
		(PERIOD.PER_MONTH between '01' and '03') and
		(POD.PORT_ID = Im_NettWeight19.POD_ID) and
		(PERIOD.PER_ID = Im_NettWeight19.PER_ID)
group by
		POD.PORT_CODE,
		POD.PORT_DESC,
		POD.PROV_DESC
) as D14
on (((D13.POD_Code = D14.POD_Code) and (D13.POD_Desc =
D14.POD_Desc))
 and (D13.Province_Desc = D14.Province_Desc))
) as Query220,
(select
		coalesce(D10.Province_Desc,D11.Province_Desc) as
Province_Desc,
		coalesce(D10.POD_Code,D11.POD_Code) as POD_Code,
		coalesce(D10.POD_Desc,D11.POD_Desc) as POD_Desc,
		D11.C_Nett_Weight as C_Nett_Weight,
		D10.C_CIF as C_CIF
from
(select
		POD.PROV_DESC as Province_Desc,
		POD.PORT_CODE as POD_Code,
		POD.PORT_DESC as POD_Desc,
		SUM(Im_CIF.OBSERVATION ) as C_CIF
from
		PORT POD,
(select
		FACT.PER_ID as PER_ID,
		FACT.POD_ID as POD_ID,
		FACT.OBSERVATION as OBSERVATION
from
		FACT FACT,
		DATA_SOURCE DATA_SOURCE
where
		(DATA_SOURCE.DS_COLLECTION_CODE = 'I') and
		(FACT.LATEST = 'Y') and
		(FACT.DI_ID = '2') and
		(DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Im_CIF,
		PERIOD PERIOD
where
```

```
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH = '03') and
            (POD.PORT_ID = Im_CIF.POD_ID) and
            (PERIOD.PER_ID = Im_CIF.PER_ID)
 group by
            POD.PROV_DESC,
            POD.PORT_CODE,
            POD.PORT_DESC
 ) as D10
             full outer join
 (select
            POD.PROV_DESC  as  Province_Desc,
            POD.PORT_CODE  as  POD_Code,
            POD.PORT_DESC  as  POD_Desc,
            SUM(Im_NettWeight.OBSERVATION  )  as  C_Nett_Weight
 from
            PORT POD,
  (select
            FACT.PER_ID  as  PER_ID,
            FACT.POD_ID  as  POD_ID,
            FACT.OBSERVATION  as  OBSERVATION
 from
            FACT FACT,
            DATA_SOURCE DATA_SOURCE
 where
            (DATA_SOURCE.DS_COLLECTION_CODE = 'I') and
            (FACT.LATEST = 'Y') and
            (FACT.DI_ID = '3') and
            (DATA_SOURCE.DS_ID = FACT.DS_ID)
 ) as Im_NettWeight,
            PERIOD PERIOD
 where
            (PERIOD.PER_YEAR = '2007') and
            (PERIOD.PER_MONTH = '03') and
            (POD.PORT_ID = Im_NettWeight.POD_ID) and
            (PERIOD.PER_ID = Im_NettWeight.PER_ID)
 group by
            POD.PROV_DESC,
            POD.PORT_CODE,
            POD.PORT_DESC
 ) as D11
 on (((D10.Province_Desc = D11.Province_Desc) and (D10.POD_Code =
D11.POD_Code))
  and (D10.POD_Desc = D11.POD_Desc))
 ) as Query121
 where
       (((Query121.Province_Desc = Query220.Province_Desc) and
(Query121.POD_Code = Query220.POD_Code)) and
(Query121.POD_Desc = Query220.POD_Desc))
```

7	Exports by Destination Countries and Groups of Goods (Non Oil & Gas, Oil and Gas) --> Buletin Table (feb 2007, feb 2008, jan-feb 2007, and jan-feb 2008)	select Query620.Oil_Group_Desc as levelkey, Query620.Country_Desc as level0key, Query321.L_FOB as L_FOB, Query321.C_FOB as C_FOB, Query321.Kenaikan_FOB as Kenaikan_FOB, Query620.Jan_L_FOB as Jan_L_FOB, Query620.Jan_C_FOB as Jan_C_FOB, Query620.Kenaikan_Jan_x as Kenaikan_Jan_x from (select Query418.Jan_L_FOB as Jan_L_FOB, Query519.Jan_C_FOB as Jan_C_FOB, Query519.Oil_Group_Desc as Oil_Group_Desc, Query519.Country_Desc as Country_Desc, (((Query519.Jan_C_FOB - Query418.Jan_L_FOB) / Query418.Jan_L_FOB) * 100) as Kenaikan_Jan_x from (select COMMODITY.OILGRP_DESC as Oil_Group_Desc, COUNTRY.COUNTRY_DESC as Country_Desc, SUM(Ex_FOB16.OBSERVATION) as Jan_L_FOB from COMMODITY COMMODITY, COUNTRY COUNTRY, (select FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID, FACT.COUNTRY_ID as COUNTRY_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB16, PERIOD PERIOD where ((PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '02')) and (COMMODITY.COM_ID = Ex_FOB16.COM_ID) and (COUNTRY.COUNTRY_ID = Ex_FOB16.COUNTRY_ID) and (PERIOD.PER_ID = Ex_FOB16.PER_ID) group by COMMODITY.OILGRP_DESC, COUNTRY.COUNTRY_DESC) as Query418, (select SUM(Ex_FOB17.OBSERVATION) as Jan_C_FOB, COMMODITY.OILGRP_DESC as Oil_Group_Desc, COUNTRY.COUNTRY_DESC as Country_Desc from (select FACT.COM_ID as COM_ID,

```
		FACT.PER_ID  as  PER_ID,
		FACT.COUNTRY_ID  as  COUNTRY_ID,
		FACT.OBSERVATION  as  OBSERVATION
from
		FACT FACT,
		DATA_SOURCE DATA_SOURCE
where
		(DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
		(FACT.LATEST = 'Y') and
		(FACT.DI_ID = '1') and
		(DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB17,
		COMMODITY COMMODITY,
		COUNTRY COUNTRY,
		PERIOD PERIOD
where
		((PERIOD.PER_YEAR = '2008') and (PERIOD.PER_MONTH
between '01' and '02')) and
		(COMMODITY.COM_ID = Ex_FOB17.COM_ID) and
		(COUNTRY.COUNTRY_ID = Ex_FOB17.COUNTRY_ID) and
		(PERIOD.PER_ID = Ex_FOB17.PER_ID)
group by
		COMMODITY.OILGRP_DESC,
		COUNTRY.COUNTRY_DESC
) as Query519
where
		((Query418.Oil_Group_Desc = Query519.Oil_Group_Desc)
  and (Query418.Country_Desc = Query519.Country_Desc))
) as Query620,
(select
		Query214.Country_Desc  as  Country_Desc,
		Query214.Oil_Group_Desc  as  Oil_Group_Desc,
		Query115.L_FOB  as  L_FOB,
		Query214.C_FOB  as  C_FOB,
		(((Query214.C_FOB - Query115.L_FOB) / Query115.L_FOB) *
100)  as  Kenaikan_FOB
from
(select
		SUM(Ex_FOB13.OBSERVATION  )  as  C_FOB,
		COUNTRY.COUNTRY_DESC  as  Country_Desc,
		COMMODITY.OILGRP_DESC  as  Oil_Group_Desc
from
(select
		FACT.COM_ID  as  COM_ID,
		FACT.PER_ID  as  PER_ID,
		FACT.COUNTRY_ID  as  COUNTRY_ID,
		FACT.OBSERVATION  as  OBSERVATION
from
		FACT FACT,
		DATA_SOURCE DATA_SOURCE
where
		(DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
		(FACT.LATEST = 'Y') and
		(FACT.DI_ID = '1') and
		(DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB13,
		COUNTRY COUNTRY,
```

```
        COMMODITY COMMODITY,
        PERIOD PERIOD
where
        ((PERIOD.PER_YEAR = '2008') and (PERIOD.PER_MONTH =
'02')) and
        (COMMODITY.COM_ID = Ex_FOB13.COM_ID) and
        (COUNTRY.COUNTRY_ID = Ex_FOB13.COUNTRY_ID) and
        (PERIOD.PER_ID = Ex_FOB13.PER_ID)
group by
        COUNTRY.COUNTRY_DESC,
        COMMODITY.OILGRP_DESC
) as Query214,
(select
        COMMODITY.OILGRP_DESC  as  Oil_Group_Desc,
        COUNTRY.COUNTRY_DESC  as  Country_Desc,
        SUM(Ex_FOB.OBSERVATION  )  as  L_FOB
from
        COMMODITY COMMODITY,
        COUNTRY COUNTRY,
 (select
        FACT.COM_ID  as  COM_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.COUNTRY_ID  as  COUNTRY_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '1') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB,
        PERIOD PERIOD
where
        ((PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH =
'02')) and
        (COMMODITY.COM_ID = Ex_FOB.COM_ID) and
        (COUNTRY.COUNTRY_ID = Ex_FOB.COUNTRY_ID) and
        (PERIOD.PER_ID = Ex_FOB.PER_ID)
group by
        COMMODITY.OILGRP_DESC,
        COUNTRY.COUNTRY_DESC
) as Query115
where
        ((Query115.Oil_Group_Desc = Query214.Oil_Group_Desc)
  and (Query115.Country_Desc = Query214.Country_Desc))
) as Query321
where
     ((Query321.Country_Desc = Query620.Country_Desc) and
(Query321.Oil_Group_Desc = Query620.Oil_Group_Desc))
```

8	Export of Indonesia By SITC 3 Digit --> Monthly Table (march 2007 and jan-march 2007)	select Query220.SITC_3_Digits_Code as levelkey, Query220.SITC_3_Digits_Desc__Eng_ as level0key, Query121.C_Nett_Weight as C_Nett_Weight, Query220.Cum_Nett_Weight as Cum_Nett_Weight, Query121.C_FOB as C_FOB, Query220.Cum_FOB as Cum_FOB from (select D13.Cum_FOB as Cum_FOB, D14.Cum_Nett_Weight as Cum_Nett_Weight, coalesce(D13.SITC_3_Digits_Code,D14.SITC_3_Digits_Code) as SITC_3_Digits_Code, coalesce(D13.SITC_3_Digits_Desc__Eng_,D14.SITC_3_Digits_Desc__Eng_) as SITC_3_Digits_Desc__Eng_ from (select SITC.SITC3 as SITC_3_Digits_Code, SITC.SITC3_DESC_ENG as SITC_3_Digits_Desc__Eng_, SUM(Ex_FOB18.OBSERVATION) as Cum_FOB from SITC SITC, (select FACT.SITC_ID as SITC_ID, FACT.PER_ID as PER_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB18, PERIOD PERIOD where (PERIOD.PER_YEAR = '2007') and (PERIOD.PER_MONTH between '01' and '03') and (SITC.SITC_ID = Ex_FOB18.SITC_ID) and (PERIOD.PER_ID = Ex_FOB18.PER_ID) group by SITC.SITC3, SITC.SITC3_DESC_ENG) as D13 full outer join (select SITC.SITC3 as SITC_3_Digits_Code, SITC.SITC3_DESC_ENG as SITC_3_Digits_Desc__Eng_, SUM(Ex_NettWeight19.OBSERVATION) as Cum_Nett_Weight from SITC SITC, (select FACT.SITC_ID as SITC_ID, FACT.PER_ID as PER_ID, FACT.OBSERVATION as OBSERVATION

```
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight19,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (SITC.SITC_ID = Ex_NettWeight19.SITC_ID) and
        (PERIOD.PER_ID = Ex_NettWeight19.PER_ID)
group by
        SITC.SITC3,
        SITC.SITC3_DESC_ENG
) as D14
on ((D13.SITC_3_Digits_Code = D14.SITC_3_Digits_Code)
 and (D13.SITC_3_Digits_Desc__Eng_ =
D14.SITC_3_Digits_Desc__Eng_))
) as Query220,
(select
        coalesce(D10.SITC_3_Digits_Code,D11.SITC_3_Digits_Code)
as  SITC_3_Digits_Code,

coalesce(D10.SITC_3_Digits_Desc__Eng_,D11.SITC_3_Digits_Desc__En
g_)  as  SITC_3_Digits_Desc__Eng_,
        D10.C_FOB  as  C_FOB,
        D11.C_Nett_Weight  as  C_Nett_Weight
from
(select
        SITC.SITC3  as  SITC_3_Digits_Code,
        SITC.SITC3_DESC_ENG  as  SITC_3_Digits_Desc__Eng_,
        SUM(Ex_FOB.OBSERVATION  )  as  C_FOB
from
        SITC SITC,
(select
        FACT.SITC_ID  as  SITC_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '1') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH = '03') and
        (SITC.SITC_ID = Ex_FOB.SITC_ID) and
        (PERIOD.PER_ID = Ex_FOB.PER_ID)
```

```
group by
        SITC.SITC3,
        SITC.SITC3_DESC_ENG
) as D10
         full outer join
(select
        SITC.SITC3  as  SITC_3_Digits_Code,
        SITC.SITC3_DESC_ENG  as  SITC_3_Digits_Desc__Eng_,
        SUM(Ex_NettWeight.OBSERVATION  )  as  C_Nett_Weight
from
        SITC SITC,
(select
        FACT.SITC_ID  as  SITC_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH = '03') and
        (SITC.SITC_ID = Ex_NettWeight.SITC_ID) and
        (PERIOD.PER_ID = Ex_NettWeight.PER_ID)
group by
        SITC.SITC3,
        SITC.SITC3_DESC_ENG
) as D11
on ((D10.SITC_3_Digits_Code = D11.SITC_3_Digits_Code)
 and (D10.SITC_3_Digits_Desc__Eng_ =
D11.SITC_3_Digits_Desc__Eng_))
) as Query121
where
     ((Query121.SITC_3_Digits_Code = Query220.SITC_3_Digits_Code)
and (Query121.SITC_3_Digits_Desc__Eng_ =
Query220.SITC_3_Digits_Desc__Eng_))
```

9	Development of the Agricultural Products are Exported --> Buletin Table(jan-march 2007 and jan-march 2008)	select Query220.Commodity_Group_Desc as levelkey, Query121.JanX_L_Nett_Weight as JanX_L_Nett_Weight, Query220.JanX_C_Nett_Weight as JanX_C_Nett_Weight, Query121.JanX_L_FOB as JanX_L_FOB, Query220.JanX_C_FOB as JanX_C_FOB, (SUM(Query220.JanX_C_FOB) / SUM(Query220.JanX_C_FOB)) as Peranan_FOB from (select coalesce(D13.Commodity_Group_Desc,D14.Commodity_Group_Desc) as Commodity_Group_Desc, D14.JanX_C_Nett_Weight as JanX_C_Nett_Weight, D13.JanX_C_FOB as JanX_C_FOB from (select COMMODITY.COMGRP_DESC as Commodity_Group_Desc, SUM(Ex_FOB18.OBSERVATION) as JanX_C_FOB from COMMODITY COMMODITY, (select FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT, DATA_SOURCE DATA_SOURCE where (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and (FACT.LATEST = 'Y') and (FACT.DI_ID = '1') and (DATA_SOURCE.DS_ID = FACT.DS_ID)) as Ex_FOB18, PERIOD PERIOD where (PERIOD.PER_YEAR = '2008') and (PERIOD.PER_MONTH between '01' and '03') and (COMMODITY.SECTOR = '201') and (COMMODITY.COM_ID = Ex_FOB18.COM_ID) and (PERIOD.PER_ID = Ex_FOB18.PER_ID) group by COMMODITY.COMGRP_DESC) as D13 full outer join (select COMMODITY.COMGRP_DESC as Commodity_Group_Desc, SUM(Ex_NettWeight19.OBSERVATION) as JanX_C_Nett_Weight from COMMODITY COMMODITY, (select FACT.COM_ID as COM_ID, FACT.PER_ID as PER_ID, FACT.OBSERVATION as OBSERVATION from FACT FACT,

```
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight19,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2008') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COMMODITY.SECTOR = '201') and
        (COMMODITY.COM_ID = Ex_NettWeight19.COM_ID) and
        (PERIOD.PER_ID = Ex_NettWeight19.PER_ID)
group by
        COMMODITY.COMGRP_DESC
) as D14
on (D13.Commodity_Group_Desc = D14.Commodity_Group_Desc)
) as Query220,
(select
        D10.JanX_L_FOB  as  JanX_L_FOB,
        D11.JanX_L_Nett_Weight  as  JanX_L_Nett_Weight,

coalesce(D10.Commodity_Group_Desc,D11.Commodity_Group_Desc)
as  Commodity_Group_Desc
 from
(select
        COMMODITY.COMGRP_DESC  as  Commodity_Group_Desc,
        SUM(Ex_FOB.OBSERVATION  )  as  JanX_L_FOB
from
        COMMODITY COMMODITY,
(select
        FACT.COM_ID  as  COM_ID,
        FACT.PER_ID  as  PER_ID,
        FACT.OBSERVATION  as  OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '1') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_FOB,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COMMODITY.SECTOR = '201') and
        (COMMODITY.COM_ID = Ex_FOB.COM_ID) and
        (PERIOD.PER_ID = Ex_FOB.PER_ID)
group by
        COMMODITY.COMGRP_DESC
) as D10
         full outer join
(select
        COMMODITY.COMGRP_DESC  as  Commodity_Group_Desc,
```

```
        SUM(Ex_NettWeight.OBSERVATION ) as JanX_L_Nett_Weight
from
        COMMODITY COMMODITY,
(select
        FACT.COM_ID as COM_ID,
        FACT.PER_ID as PER_ID,
        FACT.OBSERVATION as OBSERVATION
from
        FACT FACT,
        DATA_SOURCE DATA_SOURCE
where
        (DATA_SOURCE.DS_COLLECTION_CODE = 'E') and
        (FACT.LATEST = 'Y') and
        (FACT.DI_ID = '3') and
        (DATA_SOURCE.DS_ID = FACT.DS_ID)
) as Ex_NettWeight,
        PERIOD PERIOD
where
        (PERIOD.PER_YEAR = '2007') and
        (PERIOD.PER_MONTH between '01' and '03') and
        (COMMODITY.SECTOR = '201') and
        (COMMODITY.COM_ID = Ex_NettWeight.COM_ID) and
        (PERIOD.PER_ID = Ex_NettWeight.PER_ID)
group by
        COMMODITY.COMGRP_DESC
) as D11
on (D10.Commodity_Group_Desc = D11.Commodity_Group_Desc)
) as Query121
where
     (Query121.Commodity_Group_Desc =
Query220.Commodity_Group_Desc)
Group by
  Query220.Commodity_Group_Desc ,
Query121.JanX_L_Nett_Weight ,
Query220.JanX_C_Nett_Weight ,
Query121.JanX_L_FOB ,
Query220.JanX_C_FOB
```

3. Struktur Data Sensus Penduduk 2010

1) Fact Table Structure

No	Field Name	Data Type	Description
1	FACT_ID	BIGINT	-
2	PER_ID	BIGINT	Period ID
3	REG_ID	BIGINT	Region ID
4	CENSUS_BLOCK	CHARACTER	Nomor blok sensus
5	SLS_NUMBER	CHARACTER	Nomor SLS
6	PHYSICAL_BUILDING	CHARACTER	Nomor Bangunan fisik
7	CENSUS_BUILDING	CHARACTER	Nomor Bangunan Sensus
8	HH_NUMBER	CHARACTER	Nomor rumah tangga
9	PERSNUM	CHARACTER	Nomor anggota rumah tangga

10	DOCTYPE	CHARACTER	Tipe Dokumen
11	NAME	VARCHAR	Nama anggota rumah tangga
11	RELAT	BIGINT	Hubungan dengan KRT
12	SEX	BIGINT	-
13	DATEBORN	CHARACTER	-
14	MONTHBORN	CHARACTER	-
15	YEARBORN	CHARACTER	-
16	AGE	INTEGER	-
17	RELIGION	BIGINT	-
18	SEEING	BIGINT	Mempunyai kesulitan melihat
19	HEARING	BIGINT	Mempunyai kesulitan mendengar
20	HAND_AND_FOOT	BIGINT	Mempunyai kesulitan berjalan atau naik tangga
21	CONCENTRATING	BIGINT	Mempunyai kesulitan mengingat
22	SELFCARE	BIGINT	Mempunyai kesulitan mengurus diri sendiri
23	ETHNICITY	BIGINT	Suku bangsa
24	FOREIGNER	BIGINT	Kewarganegaraan
25	PROV5	CHARACTER	Kode propinsi 5 tahun lalu
26	DISTRICT5	CHARACTER	Kode kabupaten 5 tahun lalu
27	DAILY_LANGUAGE	BIGINT	Kode bahasa sehari-hari
28	SPEAK_INDONESIAN	BIGINT	Mampu berbahasa Indonesia
29	SCHOOL_ATTENDANCE	BIGINT	Status sekolah
30	EDUCATION	BIGINT	Ijasah tertinggi
31	LATIN_LITERACY	BIGINT	Dapat membaca dan menulis huruf latin
32	OTHERLITERACY	BIGINT	Dapat membaca dan menulis huruf lainnya
33	MARSTAT	BIGINT	Status perkawinan
34	ECONACTIVE	BIGINT	Bekerja seminggu yang lalu
35	TEMPORARYNOTWORKING	BIGINT	Mempunyai pekerjaan tetap, tapi sementara tidak bekerja
36	SEEKINGWORK	BIGINT	Mencari pekerjaan
37	WILLINGTOWORK	BIGINT	Bersedia bekerja bila disediakan
38	INDUSTRY	BIGINT	Kode lapangan usaha
39	ECONSTATUS	BIGINT	Kedudukan di pekerjaan utama
40	PERNAH_MELAHIRKAN	BIGINT	Melahirkan anak hidup

2) Dimension Table Structure

a. DIM_AGE

No	Field Name	Data Type	Description
1	ID_AGE	INTEGER	Age ID
2	AGE1	VARCHAR	-
3	AGEGROUP1	VARCHAR	-
4	AGEGROUP2	VARCHAR	-
5	AGEGROUP3	VARCHAR	-
6	AGEGROUP4	VARCHAR	-
7	AGEGROUP5	VARCHAR	-

b. DIM_AGEGROUP1

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Age group ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

c. DIM_CONCENTRATING

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Concentrating ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

d. DIM_DAILY_LANGUAGE

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Daily language ID
2	KDBAHASA	CHARACTER	Kode Bahasa
3	NMBAHASA	VARCHAR	Nama Bahasa
4	KDWIL	CHARACTER	Kode wilayah
5	START_DATE	DATE	-
6	END_DATE	DATE	-

e. DIM_ECONACTIVE

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Econactive ID
2	ID	CHARACTER	-

3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

f. DIM_ECONSTATUS

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Econstatus ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

g. DIM_EDUCATION

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Education ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

h. DIM_ETHNICITY

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Ethnicity ID
2	KDSUKU	CHARACTER	Kode suku
3	NMSUKU	VARCHAR	Nama suku
4	KDWIL	CHARACTER	Kode wilayah
5	START_DATE	DATE	-
6	END_DATE	DATE	-

i. DIM_FOREIGNER

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Foreigner ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

j. DIM_HAND_AND_FOOT

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Hand and foot ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

k. DIM_HEARING

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Hearing ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

l. DIM_INDUSTRY

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Industry ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

m. DIM_LATIN_LITERACY

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Latin literacy ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

n. DIM_MARSTAT

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Marital status ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

o. DIM_MELAHIRKAN_1JAN2009

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Melahirkan 1jan2009 ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

p. DIM_OTHERLITERACY

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Other literacy ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

q. DIM_PERNAH_MELAHIRKAN

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Pernah melahirkan ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

r. DIM_RELAT

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Relationship ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

s. DIM_RELIGION

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Religion ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

t. DIM_SCHOOL_ATTENDANCE

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	School attendance ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

u. DIM_SEEING

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Seeing ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

v. DIM_SEEKINGWORK

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Seeking work ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

w. DIM_SELFCARE

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Selfcare ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

x. DIM_SEX

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Sex ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

y. DIM_SPEAK_INDONESIAN

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Speak Indonesian ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

z. DIM_TEMPORARYNOTWORKING

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Temporary not working ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

aa. DIM_WILLINGTOWORK

No	Field Name	Data Type	Description
1	ID_SK	BIGINT	Willing to work ID
2	ID	CHARACTER	-
3	KETERANGAN	VARCHAR	-
4	START_DATE	DATE	-
5	END_DATE	DATE	-

4. Query Tabel POC Data Sensus Penduduk 2010

No	Table Name	Query
1	Tabel 01. Penduduk menurut Umur Tunggal, Daerah Perkotaan/Pedesaan, dan Jenis Kelamin	select AGE.AGE1 as Umur_Tunggalkey, AGE.ID_AGE as Umur, URBANRURAL.KET as Perkotaan_Perdesaankey, SEX.KETERANGAN as Jenis_Kelaminkey, SUM(FACT_3.OBSERVATION for AGE.AGE1,AGE.ID_AGE,URBANRURAL.KET,SEX.KETERANGAN) as Jumlah from AGE AGE, URBANRURAL URBANRURAL, SEX SEX, FACT_3 FACT_3 where (AGE.ID_AGE = FACT_3.AGE) and (SEX.ID = FACT_3.SEX) and (URBANRURAL.ID_URBANRURAL = FACT_3.URBAN_RURAL) group by

		AGE.AGE1, AGE.ID_AGE, URBANRURAL.KET, SEX.KETERANGAN
2	Tabel 02. Penduduk menurut Kelompok Umur, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin	select AGE.AGEGROUP1 as Kelompok_Umur_1key, URBANRURAL.KET as Perkotaan_Perdesaankey, SEX.KETERANGAN as Jenis_Kelaminkey, SUM(FACT_3.OBSERVATION for AGE.AGEGROUP1,URBANRURAL.KET,SEX.KETERANGAN) as Jumlah from AGE AGE, URBANRURAL URBANRURAL, SEX SEX, FACT_3 FACT_3 where (AGE.ID_AGE = FACT_3.AGE) and (SEX.ID = FACT_3.SEX) and (URBANRURAL.ID_URBANRURAL = FACT_3.URBAN_RURAL) group by AGE.AGEGROUP1, URBANRURAL.KET, SEX.KETERANGAN
3	Tabel 03. Penduduk menurut Provinsi, Daerah Perkotaan/Perdesaan, dan Jenis Kelamin	select REGION.PROV_DESC as Nama_Provinsikey, URBANRURAL.KET as Perkotaan_Perdesaankey, SEX.KETERANGAN as Jenis_Kelaminkey, SUM(FACT_3.OBSERVATION for REGION.PROV_DESC,URBANRURAL.KET,SEX.KETERANGAN) as Jumlah from REGION REGION, URBANRURAL URBANRURAL, SEX SEX, FACT_3 FACT_3 where (REGION.REG_ID = FACT_3.REG_ID) and (SEX.ID = FACT_3.SEX) and (URBANRURAL.ID_URBANRURAL = FACT_3.URBAN_RURAL) group by REGION.PROV_DESC, URBANRURAL.KET, SEX.KETERANGAN
4	Tabel 04. Penduduk menurut Kelompok Umur dan Hubungan dengan KRT	select RELAT.KETERANGAN as Hubungan_dengan_KRTkey, AGE.AGEGROUP1 as Kelompok_Umur_1key, SUM(FACT_3.OBSERVATION for RELAT.KETERANGAN,AGE.AGEGROUP1) as Jumlah from RELAT RELAT, AGE AGE, FACT_3 FACT_3 where (RELAT.ID = FACT_3.RELAT) and (AGE.ID_AGE = FACT_3.AGE) group by RELAT.KETERANGAN,

		AGE.AGEGROUP1
5	Tabel 05. Penduduk Berumur 10 Tahun ke Atas menurut Kelompok Umur dan Status Perkawinan	select MARSTAT.KETERANGAN as Status_Perkawinankey, AGE.AGEGROUP1 as Kelompok_Umur_1key, SUM(FACT_3.OBSERVATION for MARSTAT.KETERANGAN,AGE.AGEGROUP1) as Jumlah from MARSTAT MARSTAT, AGE AGE, FACT_3 FACT_3 where (MARSTAT.ID = FACT_3.MARSTAT) and (AGE.ID_AGE = FACT_3.AGE) group by MARSTAT.KETERANGAN, AGE.AGEGROUP1
6	Tabel 06. Penduduk berumur 10 Tahun ke Atas menurut Provinsi dan Status Perkawinan	select MARSTAT.KETERANGAN as Status_Perkawinankey, REGION.PROV_DESC as Nama_Provinsikey, SUM(FACT_3.OBSERVATION for MARSTAT.KETERANGAN,REGION.PROV_DESC) as Jumlah from MARSTAT MARSTAT, REGION REGION, FACT_3 FACT_3 where (MARSTAT.ID = FACT_3.MARSTAT) and (REGION.REG_ID = FACT_3.REG_ID) group by MARSTAT.KETERANGAN, REGION.PROV_DESC
7	Tabel 07. Penduduk menurut Kelompok Umur dan Agama	select RELIGION.KETERANGAN as Agamakey, REGION.PROV_DESC as Nama_Provinsikey, SUM(FACT_3.OBSERVATION for RELIGION.KETERANGAN,REGION.PROV_DESC) as Jumlah from RELIGION RELIGION, REGION REGION, FACT_3 FACT_3 where (RELIGION.ID = FACT_3.RELIGION) and (REGION.REG_ID = FACT_3.REG_ID) group by RELIGION.KETERANGAN, REGION.PROV_DESC
8	Tabel 08. Penduduk menurut Provinsi dan Agama	select REGION.PROV_ID as Kode_Provinsi, REGION.PROV_DESC as Nama_Provinsikey, RELIGION.KETERANGAN as Agamakey, SUM(FACT_3.OBSERVATION for REGION.PROV_ID,REGION.PROV_DESC,RELIGION.KETERANGAN) as Jumlah from REGION REGION, RELIGION RELIGION, FACT_3 FACT_3

		where (REGION.REG_ID = FACT_3.REG_ID) and (RELIGION.ID = FACT_3.RELIGION) group by REGION.PROV_ID, REGION.PROV_DESC, RELIGION.KETERANGAN order by Kode_Provinsi asc
9	Tabel 09. Penduduk menurut Kelompok Umur dan Kewarganegaraan	select FOREIGNER.KETERANGAN as Kewarganegaraan_WNAkey, SEX.KETERANGAN as Jenis_Kelaminkey, AGE.AGEGROUP1 as Kelompok_Umur_1key, SUM(FACT_3.OBSERVATION for FOREIGNER.KETERANGAN,SEX.KETERANGAN,AGE.AGEGROUP1) as Jumlah from FOREIGNER FOREIGNER, SEX SEX, AGE AGE, FACT_3 FACT_3 where (FOREIGNER.ID = FACT_3.FOREIGNER) and (AGE.ID_AGE = FACT_3.AGE) and (SEX.ID = FACT_3.SEX) group by FOREIGNER.KETERANGAN, SEX.KETERANGAN, AGE.AGEGROUP1
10	Tabel 10. Penduduk menurut Provinsi dan Kewarganegaraan	select FOREIGNER.KETERANGAN as Kewarganegaraan_WNAkey, SEX.KETERANGAN as Jenis_Kelaminkey, REGION.PROV_DESC as Nama_Provinsikey, XSUM(FACT_3.OBSERVATION for FOREIGNER.KETERANGAN,SEX.KETERANGAN,REGION.PROV_DESC) as Jumlah from FOREIGNER FOREIGNER, SEX SEX, REGION REGION, FACT_3 FACT_3 where (FOREIGNER.ID = FACT_3.FOREIGNER) and (REGION.REG_ID = FACT_3.REG_ID) and (SEX.ID = FACT_3.SEX) group by FOREIGNER.KETERANGAN, SEX.KETERANGAN, REGION.PROV_DESC
11	Tabel 11. Penduduk WNI menurut kelompok Umur dan Suku Bangsa	select ETHNICITY.NMSUKU as Nama_Sukukey, AGE.AGEGROUP1 as Kelompok_Umur_1key, SUM(FACT_3.OBSERVATION for ETHNICITY.NMSUKU,AGE.AGEGROUP1) as Jumlah from ETHNICITY ETHNICITY, AGE AGE,

		FACT_3 FACT_3 where (ETHNICITY.KDSUKU = FACT_3.ETHNICITY) and (AGE.ID_AGE = FACT_3.AGE) group by ETHNICITY.NMSUKU, AGE.AGEGROUP1
12	Tabel 12. Penduduk menurut Provinsi dan Suku Bangsa	select ETHNICITY.NMSUKU as Nama_Sukukey, REGION.PROV_DESC as Nama_Provinsikey, SUM(FACT_3.OBSERVATION for ETHNICITY.NMSUKU,REGION.PROV_DESC) as Jumlah from ETHNICITY ETHNICITY, REGION REGION, FACT_3 FACT_3 where (ETHNICITY.KDSUKU = FACT_3.ETHNICITY) and (REGION.REG_ID = FACT_3.REG_ID) group by ETHNICITY.NMSUKU, REGION.PROV_DESC
13	Tabel 13. Penduduk WNA menurut Kelompok Umur dan Kewarganegaraan	select FOREIGNER.KETERANGAN as Kewarganegaraan_WNAkey, AGE.AGEGROUP1 as Kelompok_Umur_1key, SUM(FACT_3.OBSERVATION for FOREIGNER.KETERANGAN,AGE.AGEGROUP1) as Jumlah from FOREIGNER FOREIGNER, AGE AGE, FACT_3 FACT_3 where (FOREIGNER.ID = FACT_3.FOREIGNER) and (AGE.ID_AGE = FACT_3.AGE) group by FOREIGNER.KETERANGAN, AGE.AGEGROUP1
14	Tabel 14. Penduduk WNA menurut Provinsi dan Kewarganegaraan	select FOREIGNER.KETERANGAN as Kewarganegaraan_WNAkey, REGION.PROV_DESC as Nama_Provinsikey, SUM(FACT_3.OBSERVATION for FOREIGNER.KETERANGAN,REGION.PROV_DESC) as Jumlah from FOREIGNER FOREIGNER, REGION REGION, FACT_3 FACT_3 where (FOREIGNER.ID = FACT_3.FOREIGNER) and (REGION.REG_ID = FACT_3.REG_ID) group by FOREIGNER.KETERANGAN, REGION.PROV_DESC