

LAPORAN KONVERSI DATA WAREHOUSE & OLAP
“PENGEMBANGAN DATA WAREHOUSE DALAM ANALISIS DATA
DVD RENTAL”
SEMESTER GANJIL 2024/2025



Dosen Pengampu:

Mohamad Irwan Afandi, ST., MSC.

Disusun oleh:

Fidyah Salsabila Putri Sillehu
22082010047

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA
TIMUR
2024

DAFTAR ISI

DAFTAR ISI.....	1
BAB I PENDAHULUAN.....	2
1.1 Latar Belakang.....	2
1.2 Rumusan Masalah.....	3
1.3 Tujuan.....	3
BAB II METODE.....	4
2.1 Metode Pengembangan.....	4
2.1.1 Perencanaan.....	4
2.1.2 Perancangan Skema.....	4
2.1.3 Ekstraksi Data.....	5
2.1.4 Pembuatan Cube Mondrian pada Tomcat.....	5
BAB III HASIL DAN IMPLEMENTASI.....	6
3.1 Implementasi Model Multidimensi.....	6
3.1.1 Skema dwh.....	6
3.2 Implementasi Cube dan OLAP.....	16
3.2.1 Cube dan OLAP (dwh).....	16
BAB IV KESIMPULAN DAN SARAN.....	22
4.1 Kesimpulan.....	22
4.2 Saran.....	22
LAMPIRAN.....	23

BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam era digital saat ini, pengelolaan data menjadi hal aspek yang sangat penting bagi organisasi dalam mengambil keputusan yang strategis. Data yang tersebar di berbagai sistem seringkali sulit dianalisis secara menyeluruh. Kondisi ini membuat proses analisis data menjadi tidak efisien dan menghambat pemahaman terhadap informasi. Untuk mengatasi kendala tersebut, *data warehouse* merupakan solusi yang dirancang untuk mengintegrasikan, menyimpan, dan mengelola data dari berbagai sumber. Penggunaan *data warehouse* dapat membuat data yang sebelumnya terpisah-pisah dapat dikumpulkan dalam satu tempat sehingga lebih mudah diakses dan dianalisis.

Data warehouse adalah tempat penyimpanan data perusahaan atau institusi yang terpusat dan terintegrasi, serta disusun sedemikian rupa, sehingga sangat mendukung proses pengambilan keputusan, dengan memberikan *input* pada *software* aplikasi (Wahono & Ali, 2021). Dalam *project* “Pengembangan Data Warehouse dalam Analisis Data DVD Rental” *data warehouse* dirancang untuk mengintegrasikan data dari berbagai sumber terkait penyewaan DVD, seperti informasi pelanggan, film, kategori film, staf, toko, serta transaksi penyewaan. Dengan pendekatan ini, data yang tersimpan di dalam *data warehouse* dapat diakses dan dianalisis secara efisien untuk mendukung pengambilan keputusan yang lebih baik dalam manajemen bisnis rental DVD.

DVD rental merupakan perusahaan yang bergerak dalam bidang penyewaan khususnya DVD. Bisnis *DVD rental* menyediakan jasa penyewaan cakram digital video (DVD) bagi pelanggan yang memiliki minat pada film namun tanpa harus membelinya. Dalam proses operasional bisnisnya, perusahaan ini memiliki data terkait dengan transaksi penyewaan, pelanggan, dan deskripsi film. Berdasarkan data-data yang dimilikinya diperlukan adanya analisis data untuk mengidentifikasi perkembangan proses bisnis yang dialami.

1.2 Rumusan Masalah

Bagaimana proses penerapan OLAP *Cube* menggunakan *tools* Mondrian pada database DVD *Rental* untuk mendukung analisis data secara multidimensional dan membantu dalam pengambilan keputusan strategis.

1.3 Tujuan

Menerapkan OLAP *Cube* menggunakan *tools* Mondrian pada database DVD Rental dengan tujuan untuk mendukung analisis data secara multidimensional yang lebih terstruktur dan efisien.

BAB II METODE

2.1 Metode Pengembangan

Metode pengembangan data warehouse dalam project ini meliputi beberapa tahap yang terstruktur untuk mendukung proses analisis data. Setiap tahap dijelaskan sebagai berikut.

2.1.1 Perencanaan

Proses perencanaan dalam pengembangan *data warehouse* dimulai dengan analisis kebutuhan data dan identifikasi struktur database. Database yang digunakan adalah database *DVD Rental*, yang terdiri dari berbagai tabel utama seperti rental, payment, customer, store, staff, date, inventory, dan film. Setiap tabel memiliki peran spesifik dalam mencatat transaksi penyewaan, data pelanggan, hingga informasi detail film. Pada tahap ini, hubungan antar-tabel dipetakan untuk memastikan data yang diperlukan dapat diintegrasikan dengan baik. Pemahaman menyeluruh terhadap sumber data ini menjadi dasar dalam perancangan skema data warehouse yang efektif dan sesuai dengan tujuan analisis. Selain itu, juga dirancang strategi untuk proses *Extract, Transform, Load* (ETL), sehingga data yang ada dapat diproses dengan optimal dan menghasilkan informasi yang terstruktur.

2.1.2 Perancangan Skema

Pada tahap ini, skema *data warehouse* dirancang menggunakan pendekatan *star schema* yang memungkinkan analisis data secara fleksibel dan efisien. *Star schema* ini terdiri dari satu tabel fakta (*fact_rentals*) dan beberapa tabel dimensi yang mendukung berbagai perspektif analisis. Sementara itu, tabel dimensi dirancang untuk memberikan detail tambahan yang relevan. Tabel dimensi yang digunakan dalam project ini terdapat *dim_customer*, *dim_date*, *dim_film*, *dim_inventory*, *dim_payment*, *dim_rental*, *dim_staff*, dan *dim_store*. Teknik *Slowly Changing Dimension* (SCD) diterapkan untuk menangani perubahan data pada dimensi tertentu. SCD tipe 1 digunakan untuk pembaruan langsung pada data yang berubah, sedangkan tipe 2 menyimpan riwayat perubahan dengan menambahkan rekaman baru.

2.1.3 Ekstraksi Data

Tahap ekstraksi data dilakukan untuk memindahkan data dari database awal ke *data warehouse*. Proses ini menggunakan alat Pentaho Data Integration (PDI) yang mendukung langkah-langkah ETL (*Extract, Transform, Load*). Pada tahap ini, data dari tabel-tabel awal seperti rental, payment, customer, store, staff, date, inventory, dan film diekstraksi, kemudian melalui proses transformasi untuk membersihkan dan menyelaraskan format data.

2.1.4 Pembuatan *Cube Mondrian* pada Tomcat

Pembuatan *cube Mondrian* dilakukan untuk mendukung analisis multidimensional pada *data warehouse*. Proses ini dimulai dengan mendefinisikan file skema XML yang menghubungkan tabel fakta dan dimensi dalam *star schema*. *Cube* ini memungkinkan pengguna untuk menganalisis data dari berbagai sudut pandang. Setelah skema *cube* selesai dibuat, implementasinya dilakukan pada server Tomcat. Server ini berfungsi sebagai penghubung antara *data warehouse* dan *tools OLAP* seperti. Dengan adanya *cube Mondrian*, analisis data menjadi lebih fleksibel dan mendalam, memungkinkan pengguna untuk menggali informasi secara efisien dan sesuai kebutuhan.

BAB III HASIL DAN IMPLEMENTASI

3.1 Implementasi Model Multidimensi

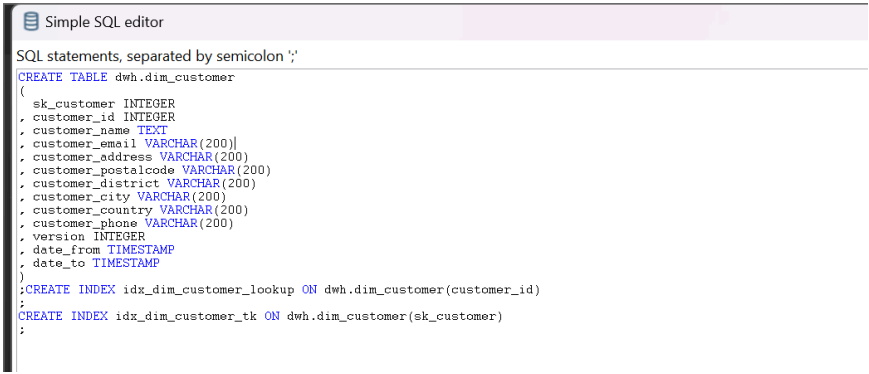
3.1.1 Skema dwh

Pada tahap awal perencanaan dalam membangun skema dwh, model yang dipilih untuk mendesain struktur data adalah *star schema*. Model ini digunakan karena mengorganisasi data secara sederhana dan mendukung analisis secara cepat dan efisien. Dalam implementasi ini, skema dwh dirancang dengan beberapa tabel dimensi yang saling terhubung dengan tabel fakta utama. Tabel-tabel dimensi tersebut memiliki peran penting dalam menyediakan konteks atau atribut pendukung yang dibutuhkan untuk analisis data. Dengan menggunakan pendekatan *star schema*, struktur data menjadi lebih terorganisir, sehingga mempermudah proses analisis mendalam dan pengambilan keputusan berbasis data. Berikut ini adalah daftar tabel-tabel dimensi yang telah dirancang sebagai bagian dari skema dwh.

1. Tabel Dimensi *Customer* (dim_customer)

Tabel “dim_customer” menyimpan informasi pelanggan untuk analisis dalam *data warehouse*. Tabel ini digunakan untuk analisis demografi, pola pembelian, dan segmentasi pelanggan guna mendukung strategi bisnis.

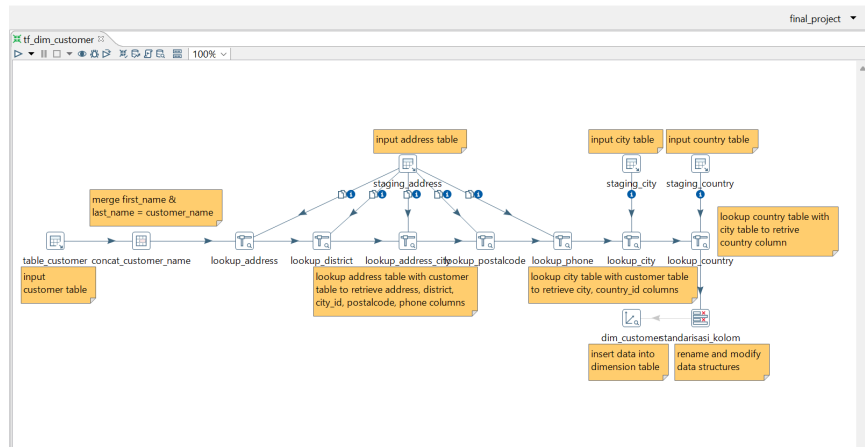
a) *Query* untuk kerangka data



```
Simple SQL editor
SQL statements, separated by semicolon ';'
CREATE TABLE dwh.dim_customer
(
  sk_customer INTEGER
, customer_id INTEGER
, customer_name TEXT
, customer_email VARCHAR(200)
, customer_address VARCHAR(200)
, customer_postalcode VARCHAR(200)
, customer_district VARCHAR(200)
, customer_city VARCHAR(200)
, customer_country VARCHAR(200)
, customer_phone VARCHAR(200)
, version INTEGER
, date_from TIMESTAMP
, date_to TIMESTAMP
)
;CREATE INDEX idx_dim_customer_lookup ON dwh.dim_customer(customer_id)
;
CREATE INDEX idx_dim_customer_tk ON dwh.dim_customer(sk_customer)
;
```

Gambar 3.1 *Query* Tabel Dimensi *Customer*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.2 Proses Data Warehouse Tabel Dimensi *Customer*

c) Hasil pembuatan kerangka tabel dan penginputan data

sk_customer	customer_id	customer_name	customer_email	customer_address	customer_postalcode	customer_district	customer
1	524	Jared Ely	jared.ely@sakilacustomer.org	1003 Qinghuangdao Street	25972	West Java	Purwakarta
2	1	Mary Smith	mary.smith@sakilacustomer.org	1913 Hancu Way	35200	Nagasaki	Sasebo
3	2	Patricia Johnson	patricia.johnson@sakilacustomer.org	1121 Loja Avenue	17886	California	San Benarc
4	3	Linda Williams	linda.williams@sakilacustomer.org	692 Joliet Street	83579	Attika	Athenai
5	4	Barbara Jones	barbara.jones@sakilacustomer.org	1566 Inegl Manor	53561	Mandalay	Myingyan
6	5	Elizabeth Brown	elizabeth.brown@sakilacustomer.org	53 Iofu Parkway	42399	Nantou	Nantou
7	6	Jennifer Davis	jennifer.davis@sakilacustomer.org	1795 Santiago de Compostela Way	16743	Texas	Laredo
8	7	Maria Miller	maria.miller@sakilacustomer.org	900 Santiago de Compostela Parkway	93896	Central Serbia	Kragujevac
9	8	Susan Wilson	susan.wilson@sakilacustomer.org	478 Joliet Way	77948	Hamilton	Hamilton
10	9	Margaret Moore	margaret.moore@sakilacustomer.org	613 Korolev Drive	45844	Masqat	Masqat
11	10	Dorothy Taylor	dorothy.taylor@sakilacustomer.org	1531 Sai Drive	53628	Esfahan	Esfahan
12	11	Lisa Anderson	lisa.anderson@sakilacustomer.org	1542 Tarbac Parkway	10027	Kanagawa	Sagamihara
13	12	Nancy Thomas	nancy.thomas@sakilacustomer.org	608 Bhopal Manor	10672	Haryana	Yamuna Na
14	13	Karen Jackson	karen.jackson@sakilacustomer.org	270 Amroha Parkway	29610	Osmaniye	Osmaniye
15	14	Betty White	betty.white@sakilacustomer.org	770 Bydgoszcz Avenue	16266	California	Citrus Heigl
16	15	Helen Harris	helen.harris@sakilacustomer.org	419 Iligan Lane	72878	Madhya Pradesh	Bhopal
17	16	Sandra Martin	sandra.martin@sakilacustomer.org	360 Toulouse Parkway	54308	England	Southend-o
18	17	Doreen Thompson	doreen.thompson@sakilacustomer.org	270 Toulon Boulevard	81786	Kalmiyya	Elista
19	18	Carol Garcia	carol.garcia@sakilacustomer.org	520 Brest Avenue	43331	Kaduna	Kaduna
20	19	Ruth Martinez	ruth.martinez@sakilacustomer.org	1417 Lancaster Avenue	72192	Northern Cape	Kimberley
21	20	Sharon Robinson	sharon.robinson@sakilacustomer.org	1688 Okara Way	21954	Northwest Border Prov	Mardan
22	21	Michelle Clark	michelle.clark@sakilacustomer.org	262 A Corua La Corua Parkway	34418	Dhaka	Tangail
23	22	Laura Rodriguez	laura.rodriguez@sakilacustomer.org	28 Charlotte Amalie Street	37551	Rabat-Sal-Zammour-Z	Sal
24	23	Sarah Lewis	sarah.lewis@sakilacustomer.org	1780 Hino Boulevard	7716	Liepaja	Liepaja
25	24	Kimberly Lee	kimberly.lee@sakilacustomer.org	96 Tafuna Way	99865	Croacia	Croacia
26	25	Deborah Walker	deborah.walker@sakilacustomer.org	934 San Felipe de Puerto Plata Street	99780	Sind	Shikarpur
27	26	Jessica Hall	jessica.hall@sakilacustomer.org	18 Duisburg Boulevard	58327	Citt del Vati	Citt del Vati
28	27	Shirley Allen	shirley.allen@sakilacustomer.org	217 Botshabelo Place	49521	Southern Mindanao	Davao
29	28	Cynthia Young	cynthia.young@sakilacustomer.org	1425 Shikarpur Manor	65599	Bihar	Munger (M

Gambar 3.3 Data Tabel Dimensi *Customer*

2. Tabel Dimensi *Date* (dim_date)

Tabel ini menyimpan informasi waktu yang terstruktur seperti tanggal, bulan, tahun, kuartal, dan minggu. Tabel ini digunakan untuk analisis berbasis waktu.

a) Query untuk kerangka data

```

Simple SQL editor

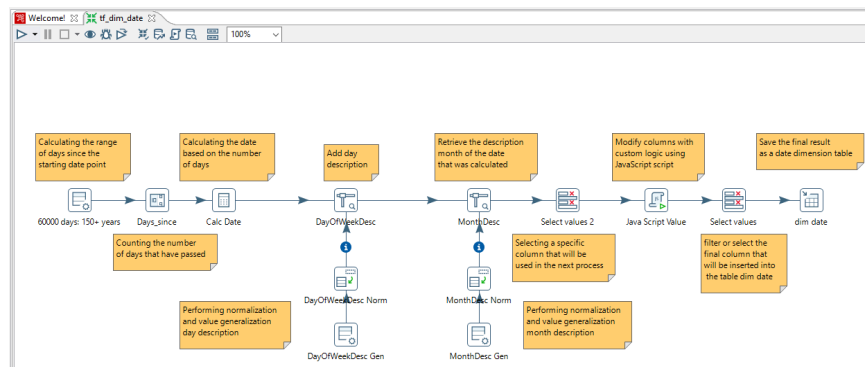
SQL statements, separated by semicolon ';'

CREATE TABLE dwb.dim_date
(
    sk_waktu DOUBLE PRECISION
, tanggal TIMESTAMP
, deskripsi_tanggal TEXT
, sk_tahun DOUBLE PRECISION
, tahun_angka SMALLINT
, sk_kuartal DOUBLE PRECISION
, kuartal_angka DOUBLE PRECISION
, kuartal VARCHAR(200)
, kuartal_tahun VARCHAR(200)
, sk_bulan DOUBLE PRECISION
, bulan_angka SMALLINT
, bulan VARCHAR(200)
, sk_minggu DOUBLE PRECISION
, minggu VARCHAR(32)
, minggu_dalam_tahun_angka SMALLINT
, minggu_dalam_bulan_angka DOUBLE PRECISION
, minggu_dalam_bulan TEXT
, hari VARCHAR(200)
, hari_dalam_tahun_angka SMALLINT
, hari_dalam_bulan_angka SMALLINT
, hari_dalam_minggu_angka SMALLINT
, is_weekend VARCHAR(200)
, banyak_hari_dalam_bulan SMALLINT
, tahun_sort VARCHAR(200)
, hari_dalam_minggu_sort VARCHAR(200)
)
;

```

Gambar 3.4 Query Tabel Dimensi Date

b) Proses data warehouse untuk penginputan data



Gambar 3.5 Proses Data Warehouse Tabel Dimensi Date

c) Hasil pembuatan kerangka tabel dan penginputan data

dim_date	sk_waktu	tanggal	deskripsi_tanggal	sk_tahun	tahun_angka	sk_kuartal	kuartal_angka	kuartal	kuartal_tahun	sk_bulan	bulan
1	19450817	1945-08-17 00:00:00.0000	17-08-1945	1945	1945	3	3	Q3	Q31945	8	8
2	19450818	1945-08-18 00:00:00.0000	18-08-1945	1945	1945	3	3	Q3	Q31945	8	8
3	19450819	1945-08-19 00:00:00.0000	19-08-1945	1945	1945	3	3	Q3	Q31945	8	8
4	19450820	1945-08-20 00:00:00.0000	20-08-1945	1945	1945	3	3	Q3	Q31945	8	8
5	19450821	1945-08-21 00:00:00.0000	21-08-1945	1945	1945	3	3	Q3	Q31945	8	8
6	19450822	1945-08-22 00:00:00.0000	22-08-1945	1945	1945	3	3	Q3	Q31945	8	8
7	19450823	1945-08-23 00:00:00.0000	23-08-1945	1945	1945	3	3	Q3	Q31945	8	8
8	19450824	1945-08-24 00:00:00.0000	24-08-1945	1945	1945	3	3	Q3	Q31945	8	8
9	19450825	1945-08-25 00:00:00.0000	25-08-1945	1945	1945	3	3	Q3	Q31945	8	8
10	19450826	1945-08-26 00:00:00.0000	26-08-1945	1945	1945	3	3	Q3	Q31945	8	8
11	19450827	1945-08-27 00:00:00.0000	27-08-1945	1945	1945	3	3	Q3	Q31945	8	8
12	19450828	1945-08-28 00:00:00.0000	28-08-1945	1945	1945	3	3	Q3	Q31945	8	8
13	19450829	1945-08-29 00:00:00.0000	29-08-1945	1945	1945	3	3	Q3	Q31945	8	8
14	19450830	1945-08-30 00:00:00.0000	30-08-1945	1945	1945	3	3	Q3	Q31945	8	8
15	19450831	1945-08-31 00:00:00.0000	31-08-1945	1945	1945	3	3	Q3	Q31945	8	8
16	19450901	1945-09-01 00:00:00.0000	01-09-1945	1945	1945	3	3	Q3	Q31945	9	9
17	19450902	1945-09-02 00:00:00.0000	02-09-1945	1945	1945	3	3	Q3	Q31945	9	9
18	19450903	1945-09-03 00:00:00.0000	03-09-1945	1945	1945	3	3	Q3	Q31945	9	9
19	19450904	1945-09-04 00:00:00.0000	04-09-1945	1945	1945	3	3	Q3	Q31945	9	9
20	19450905	1945-09-05 00:00:00.0000	05-09-1945	1945	1945	3	3	Q3	Q31945	9	9
21	19450906	1945-09-06 00:00:00.0000	06-09-1945	1945	1945	3	3	Q3	Q31945	9	9
22	19450907	1945-09-07 00:00:00.0000	07-09-1945	1945	1945	3	3	Q3	Q31945	9	9
23	19450908	1945-09-08 00:00:00.0000	08-09-1945	1945	1945	3	3	Q3	Q31945	9	9
24	19450909	1945-09-09 00:00:00.0000	09-09-1945	1945	1945	3	3	Q3	Q31945	9	9
25	19450910	1945-09-10 00:00:00.0000	10-09-1945	1945	1945	3	3	Q3	Q31945	9	9
26	19450911	1945-09-11 00:00:00.0000	11-09-1945	1945	1945	3	3	Q3	Q31945	9	9
27	19450912	1945-09-12 00:00:00.0000	12-09-1945	1945	1945	3	3	Q3	Q31945	9	9
28	19450913	1945-09-13 00:00:00.0000	13-09-1945	1945	1945	3	3	Q3	Q31945	9	9
29	19450914	1945-09-14 00:00:00.0000	14-09-1945	1945	1945	3	3	Q3	Q31945	9	9
30	19450915	1945-09-15 00:00:00.0000	15-09-1945	1945	1945	3	3	Q3	Q31945	9	9

Gambar 3.6 Data Tabel Dimensi Date

3. Tabel Dimensi Film (dim_film)

Tabel ini menyimpan informasi detail film, seperti judul, kategori, rating, dan aktor. Tabel ini digunakan untuk analisis performa film, popularitas kategori, dan preferensi pelanggan.

a) Query untuk kerangka data

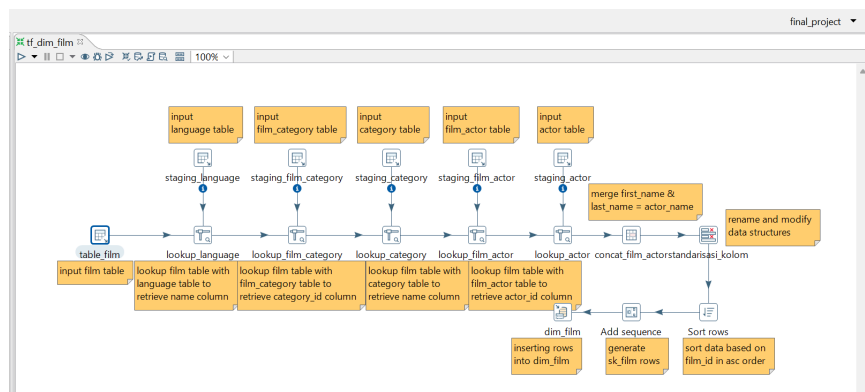
```
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.dim_film
(
    film_id INTEGER
, sk_film INTEGER
, film_title VARCHAR(255)
, film_description TEXT
, film_year INTEGER
, film_language VARCHAR(200)
, film_category VARCHAR(200)
, film_actor TEXT
, film_rentalduration SMALLINT
, film_rentalrate NUMERIC(6, 2)
, film_duration INTEGER
, film_replacementcost NUMERIC(7, 2)
, film_rating TEXT
, film_specialfeatures TEXT
)
;CREATE INDEX idx_dim_film_lookup ON dwh.dim_film(film_id)
;
```

Gambar 3.7 Query Tabel Dimensi Film

b) Proses data warehouse untuk penginputan data



Gambar 3.8 Proses Data Warehouse Tabel Dimensi Film

c) Hasil pembuatan kerangka tabel dan penginputan data

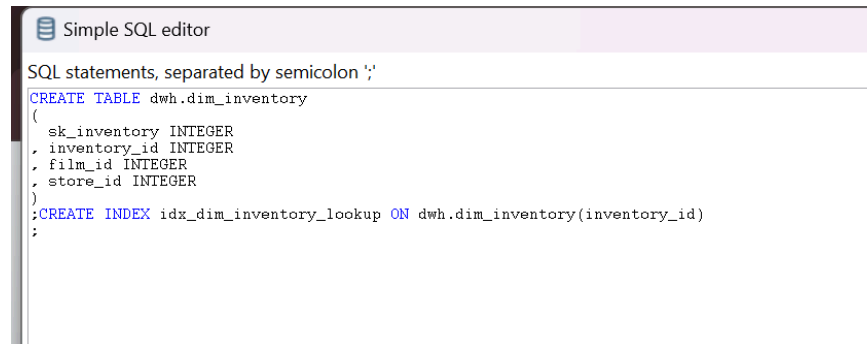
sk_film	film_id	film_title	film_description	film_year	film_language	film_category	film_actor
1	1	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a	2006	English	Documentary	Mary Keitel
2	2	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer v	2006	English	Horror	Chris Depp
3	3	Adaptation Hides	A Astounding Reflection of a Lumberjack And a Cat who must Sink	2006	English	Documentary	Julianne Dench
4	4	Affair Prejudice	A Fasciful Documentary of a Frisbee And a Lumberjack who must C	2006	English	Horror	Oprah Kimer
5	5	African Egg	A Fast-Paced Documentary of a Pastry Chef And a Dentist who muz	2006	English	Family	Thora Temple
6	6	Agent Truman	A Intrepid Panorama of a Robot And a Boy who must Escape a Sur	2006	English	Foreign	Reese West
7	7	Airplane Sierra	A Touching Saga of a Hunter And a Butler who must Discover a Bu	2006	English	Comedy	Michael Bolger
8	8	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in	2006	English	Horror	Lucille Dee
9	9	Alabama Devil	A Thoughtful Panorama of a Database Administrator And a Mad Si	2006	English	Horror	Merly Allen
10	10	Aladdin Calendar	A Action-Packed Tale of a Man And a Lumberjack who must Reach	2006	English	Sports	Rock Dukakis
11	11	Alamo Videotape	A Boring Epistle of a Butler And a Cat who must Fight a Pastry Chef	2006	English	Foreign	Michael Bening
12	12	Alaska Phantom	A Fanciful Saga of a Hunter And a Pastry Chef who must Vanquish i	2006	English	Music	Jeff Silverstone
13	13	Alli Forever	A Action-Packed Drama of a Dentist And a Crocodile who must Ba	2006	English	Horror	Jon Chase
14	14	Alice Fantasia	A Emotional Drama of a A Shark And a Database Administrator w	2006	English	Classics	Rock Dukakis
15	15	Alien Center	A Brilliant Drama of a Cat And a Mad Scientist who must Settle a F	2006	English	Foreign	Mena Hopper
16	16	Alley Evolution	A Fast-Paced Drama of a Robot And a Composer who must Battle	2006	English	Foreign	John Suvari
17	17	Alone Trip	A Fast-Paced Character Study of a Composer And a Dog who must	2006	English	Music	Renee Ball
18	18	Alter Victory	A Thoughtful Drama of a Composer And a Feminist who must Mee	2006	English	Animation	Oprah Kimer
19	19	Amadeus Holy	A Emotional Display of a Pioneer And a Technical Writer who must	2006	English	Action	Penelope Crumyn
20	20	Amelie Hellfighters	A Boring Drama of a Woman And a Squirrel who must Conquer a S	2006	English	Music	Laura Brody

Gambar 3.9 Data Tabel Dimensi Film

4. Tabel Dimensi *Inventory* (dim_inventory)

Tabel ini menyimpan informasi tentang persediaan film di setiap toko. Tabel ini digunakan untuk melacak stok film dan mendukung analisis.

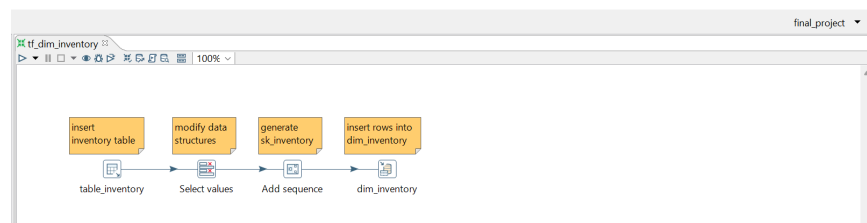
a) *Query* untuk kerangka data



```
Simple SQL editor
SQL statements, separated by semicolon ';'
CREATE TABLE dwh.dim_inventory
(
  sk_inventory INTEGER
, inventory_id INTEGER
, film_id INTEGER
, store_id INTEGER
)
;CREATE INDEX idx_dim_inventory_lookup ON dwh.dim_inventory(inventory_id)
;
```

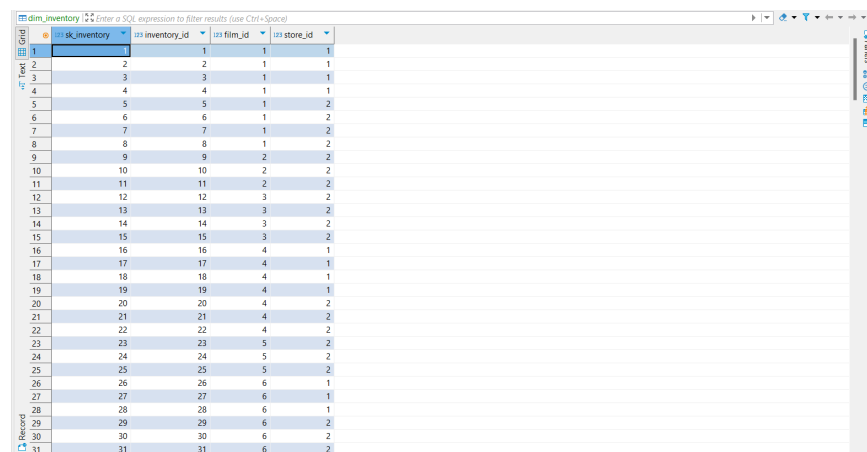
Gambar 3.10 *Query* Tabel Dimensi *Inventory*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.11 Proses *Data Warehouse* Tabel Dimensi *Inventory*

c) Hasil pembuatan kerangka tabel dan penginputan data



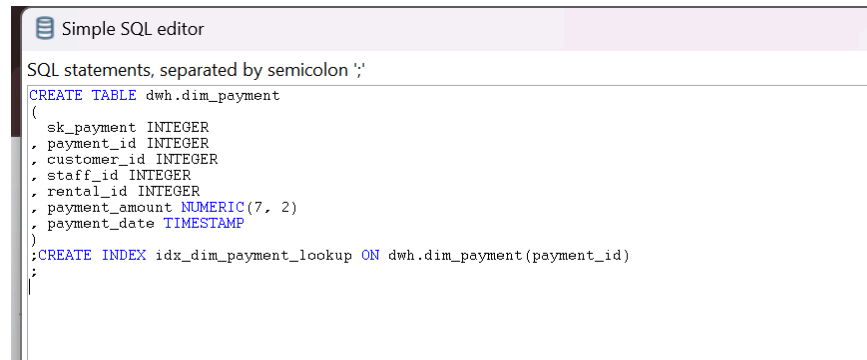
sk_inventory	inventory_id	film_id	store_id
1	1	1	1
2	2	2	1
3	3	3	1
4	4	4	1
5	5	5	2
6	6	6	2
7	7	7	2
8	8	8	2
9	9	9	2
10	10	10	2
11	11	11	2
12	12	12	2
13	13	13	2
14	14	14	2
15	15	15	2
16	16	16	2
17	17	17	2
18	18	18	2
19	19	19	2
20	20	20	2
21	21	21	2
22	22	22	2
23	23	23	2
24	24	24	2
25	25	25	2
26	26	26	2
27	27	27	2
28	28	28	2
29	29	29	2
30	30	30	2
31	31	31	2

Gambar 3.12 Data Tabel Dimensi *Inventory*

5. Tabel Dimensi *Payment* (dim_payment)

Tabel ini menyimpan informasi pembayaran, seperti jumlah pembayaran dan tanggal transaksi. Tabel ini digunakan untuk analisis performa keuangan dan pendapatan.

a) Query untuk kerangka data



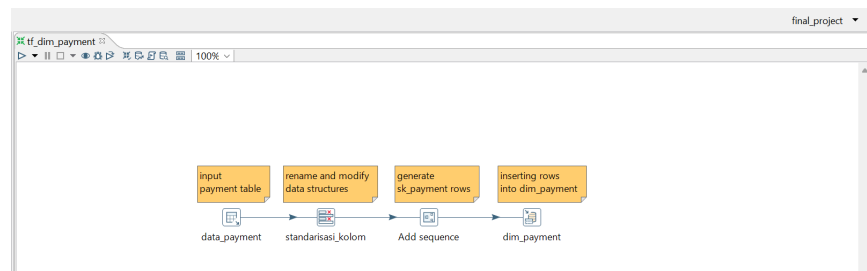
```
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.dim_payment
(
  sk_payment INTEGER
, payment_id INTEGER
, customer_id INTEGER
, staff_id INTEGER
, rental_id INTEGER
, payment_amount NUMERIC(7, 2)
, payment_date TIMESTAMP
)
;CREATE INDEX idx_dim_payment_lookup ON dwh.dim_payment(payment_id)
;
```

Gambar 3.13 Query Tabel Dimensi *Payment*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.14 Proses *Data Warehouse* Tabel Dimensi *Payment*

c) Hasil pembuatan kerangka tabel dan penginputan data

dim_payment

 Enter a SQL expression to filter results (use Ctrl-Space)

Grid

Text

	sk_payment	payment_id	customer_id	staff_id	rental_id	payment_amount	payment_date
1	17503	341	2	1,520	7.99	2007-02-15 22:25:46.996	
2	17504	341	1	1,778	1.99	2007-02-16 17:23:14.996	
3	17505	341	1	1,949	7.99	2007-02-16 22:41:45.996	
4	17506	341	2	2,829	2.99	2007-02-19 19:39:56.996	
5	17507	341	2	3,130	7.99	2007-02-20 17:31:48.996	
6	17508	341	1	3,382	5.99	2007-02-21 12:33:49.996	
7	17509	342	2	2,190	5.99	2007-02-17 23:58:17.996	
8	17510	342	1	2,914	5.99	2007-02-20 02:11:44.996	
9	17511	342	1	2,081	2.99	2007-02-20 18:57:39.996	
10	17512	343	2	1,547	4.99	2007-02-16 00:10:50.996	
11	17513	343	1	1,564	6.99	2007-02-16 01:15:33.996	
12	17514	343	2	1,879	0.99	2007-02-17 01:26:00.996	
13	17515	343	2	1,922	0.99	2007-02-17 04:32:51.996	
14	17516	343	2	2,461	6.99	2007-02-18 18:26:38.996	
15	17517	343	1	2,980	8.99	2007-02-20 07:03:29.996	
16	17518	343	1	3,407	0.99	2007-02-21 14:42:28.996	
17	17519	344	1	1,341	3.99	2007-02-15 10:54:44.996	
18	17520	344	2	1,475	4.99	2007-02-15 19:36:27.996	
19	17521	344	1	1,731	0.99	2007-02-16 14:00:38.996	
20	17522	345	2	1,210	0.99	2007-02-15 01:26:17.996	
21	17523	345	1	1,457	4.99	2007-02-15 18:34:15.996	
22	17524	345	2	1,550	0.99	2007-02-16 00:27:01.996	
23	17525	345	2	2,766	4.99	2007-02-19 16:13:41.996	
24	17526	346	1	1,994	5.99	2007-02-17 09:35:32.996	
25	17527	346	2	3,372	2.99	2007-02-21 12:02:45.996	
26	17528	346	1	3,421	2.99	2007-02-21 15:51:24.996	
27	17529	347	2	1,711	8.99	2007-02-16 12:40:18.996	
28	17530	347	2	2,274	0.99	2007-02-18 04:59:41.996	
29	17531	347	1	3,026	4.99	2007-02-20 10:16:26.996	
30	17532	347	1	3,092	8.99	2007-02-20 14:33:08.996	
31	17533	347	1	3,326	7.99	2007-02-21 07:33:16.996	

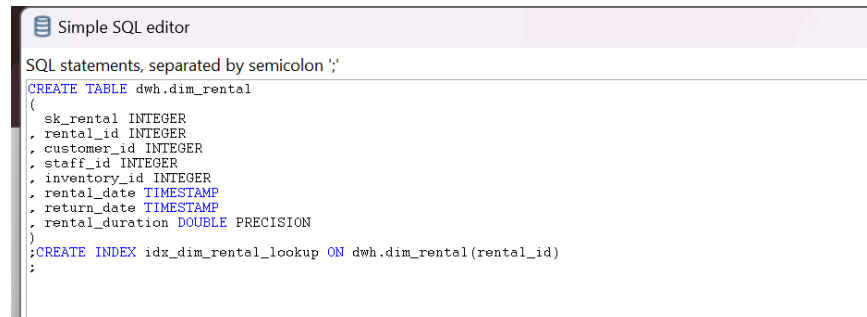
2 Record

Gambar 3.15 Data Tabel Dimensi *Payment*

6. Tabel Dimensi *Rental* (dim_rental)

Tabel ini menyimpan informasi detail transaksi penyewaan, seperti tanggal penyewaan, tanggal pengembalian, dan durasi penyewaan. Tabel ini digunakan untuk analisis aktivitas penyewaan.

a) Query untuk kerangka data



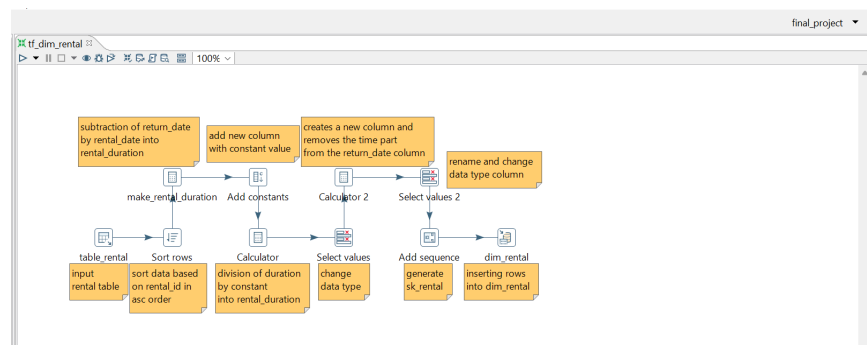
```
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.dim_rental
(
  sk_rental INTEGER
, rental_id INTEGER
, customer_id INTEGER
, staff_id INTEGER
, inventory_id INTEGER
, rental_date TIMESTAMP
, return_date TIMESTAMP
, rental_duration DOUBLE PRECISION
)
;CREATE INDEX idx_dim_rental_lookup ON dwh.dim_rental(rental_id)
;
```

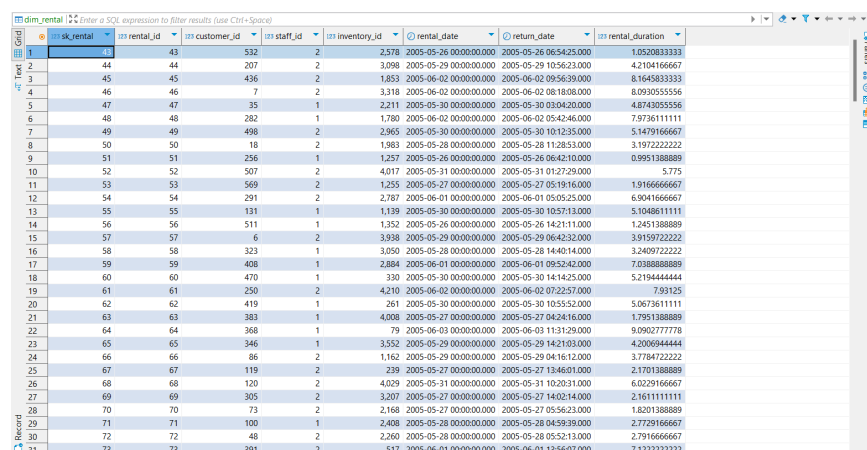
Gambar 3.16 Query Tabel Dimensi *Rental*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.17 Proses *Data Warehouse* Tabel Dimensi *Rental*

c) Hasil pembuatan kerangka tabel dan penginputan data



	sk_rental	rental_id	customer_id	staff_id	inventory_id	rental_date	return_date	rental_duration
1	43	43	532	2	2,578	2005-05-26 00:00:00.000	2005-05-26 06:54:25.000	1.052083333
2	44	44	207	2	3,098	2005-05-29 00:00:00.000	2005-05-29 10:56:23.000	4.210416667
3	45	45	436	2	1,853	2005-06-02 00:00:00.000	2005-06-02 09:56:39.000	8.164583333
4	46	46	7	2	3,318	2005-06-02 00:00:00.000	2005-06-02 08:16:08.000	8.095055556
5	47	47	35	1	2,211	2005-06-30 00:00:00.000	2005-06-30 03:04:20.000	4.874385556
6	48	48	282	1	1,780	2005-06-02 00:00:00.000	2005-06-02 05:42:46.000	7.973611111
7	49	49	498	2	2,965	2005-05-30 00:00:00.000	2005-05-30 10:12:35.000	5.147916667
8	50	50	18	2	1,983	2005-05-28 00:00:00.000	2005-05-28 11:28:53.000	3.197222222
9	51	51	256	1	1,257	2005-05-26 00:00:00.000	2005-05-26 06:42:10.000	0.995138889
10	52	52	507	2	4,017	2005-05-31 00:00:00.000	2005-05-31 01:27:29.000	5.775
11	53	53	569	2	1,255	2005-05-27 00:00:00.000	2005-05-27 05:19:16.000	1.916666667
12	54	54	291	2	2,787	2005-06-01 00:00:00.000	2005-06-01 05:05:25.000	6.904166667
13	55	55	131	1	1,139	2005-05-30 00:00:00.000	2005-05-30 10:57:13.000	5.104861111
14	56	56	511	1	1,352	2005-05-26 00:00:00.000	2005-05-26 14:21:11.000	1.245138889
15	57	57	6	2	3,938	2005-05-29 00:00:00.000	2005-05-29 06:42:32.000	3.915972222
16	58	58	323	1	3,050	2005-05-28 00:00:00.000	2005-05-28 14:40:14.000	3.244972222
17	59	59	408	1	2,884	2005-06-01 00:00:00.000	2005-06-01 09:52:42.000	7.038888889
18	60	60	470	1	330	2005-05-30 00:00:00.000	2005-05-30 14:14:25.000	5.219444444
19	61	61	250	2	4,210	2005-06-02 00:00:00.000	2005-06-02 07:22:57.000	7.93125
20	62	62	419	1	261	2005-05-30 00:00:00.000	2005-05-30 10:55:52.000	5.067361111
21	63	63	383	1	4,808	2005-05-27 00:00:00.000	2005-05-27 04:24:16.000	1.795138889
22	64	64	368	1	79	2005-06-03 00:00:00.000	2005-06-03 11:31:29.000	9.090277778
23	65	65	346	1	3,552	2005-05-29 00:00:00.000	2005-05-29 14:21:03.000	4.200694444
24	66	66	86	2	1,162	2005-05-29 00:00:00.000	2005-05-29 04:16:12.000	3.778472222
25	67	67	119	2	239	2005-05-27 00:00:00.000	2005-05-27 13:46:01.000	2.170138889
26	68	68	120	2	4,029	2005-05-31 00:00:00.000	2005-05-31 10:20:31.000	6.022916667
27	69	69	305	2	3,207	2005-05-27 00:00:00.000	2005-05-27 14:02:14.000	2.161111111
28	70	70	73	2	2,168	2005-05-27 00:00:00.000	2005-05-27 05:56:23.000	1.820138889
29	71	71	100	1	2,408	2005-05-28 00:00:00.000	2005-05-28 04:59:39.000	2.772916667
30	72	72	48	2	2,260	2005-05-28 00:00:00.000	2005-05-28 05:52:13.000	2.791666667
31	73	73	391	2	517	2005-06-01 00:00:00.000	2005-06-01 13:56:07.000	7.122222222

Gambar 3.18 Data Tabel Dimensi *Rental*

7. Tabel Dimensi *Staff* (dim_staff)

Tabel ini menyimpan informasi tentang staf, termasuk nama, alamat, dan lokasi kerja. Tabel ini digunakan untuk analisis kinerja staf dan manajemen sumber daya manusia.

a) *Query* untuk kerangka data

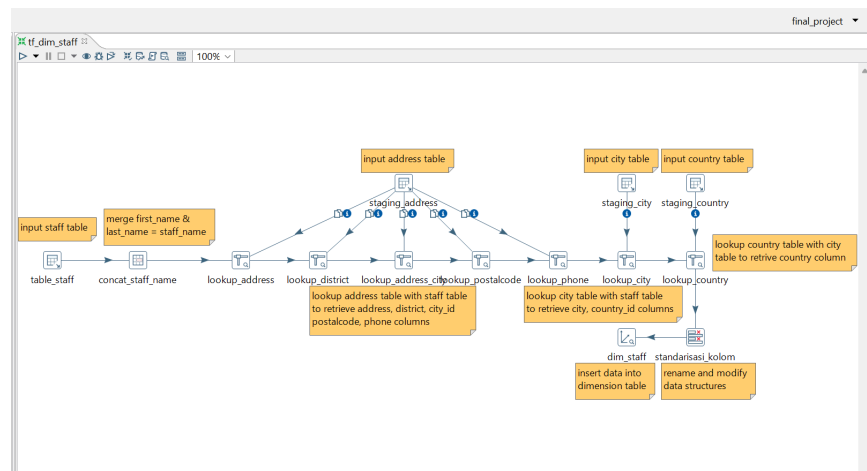
```
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.dim_staff
(
  sk_staff INTEGER
, staff_id INTEGER
, staff_name TEXT
, staff_email VARCHAR(200)
, staff_address VARCHAR(200)
, staff_postalcode VARCHAR(200)
, staff_district VARCHAR(200)
, staff_city VARCHAR(200)
, staff_country VARCHAR(200)
, staff_phone VARCHAR(200)
, version INTEGER
, date_from TIMESTAMP
, date_to TIMESTAMP
)
;
CREATE INDEX idx_dim_staff_lookup ON dwh.dim_staff(staff_id)
;
CREATE INDEX idx_dim_staff_tk ON dwh.dim_staff(sk_staff)
;
```

Gambar 3.19 *Query* Tabel Dimensi *Staff*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.20 Proses *Data Warehouse* Tabel Dimensi *Staff*

c) Hasil pembuatan kerangka tabel dan penginputan data

dim_staff	sk_staff	staff_id	staff_name	staff_email	staff_address	staff_postalcode	staff_district	staff_city	staff_country	staff_phone
1	1	1	Mike Hillyer	Mike.Hillyer@sakilastaff.com	23 Workhaven Lane	[NULL]	Alberta	Lethbridge	Canada	14033335568
2	2	2	Jon Stephens	Jon.Stephens@sakilastaff.com	1411 Lillydale Drive	[NULL]	QLD	Woodridge	Australia	6172233589

Gambar 3.21 Data Tabel Dimensi *Staff*

8. Tabel Dimensi *Store* (dim_store)

Tabel ini menyimpan informasi tentang staf, termasuk nama, alamat, dan lokasi kerja. Tabel ini digunakan untuk analisis kinerja staf dan manajemen sumber daya manusia.

a) *Query* untuk kerangka data

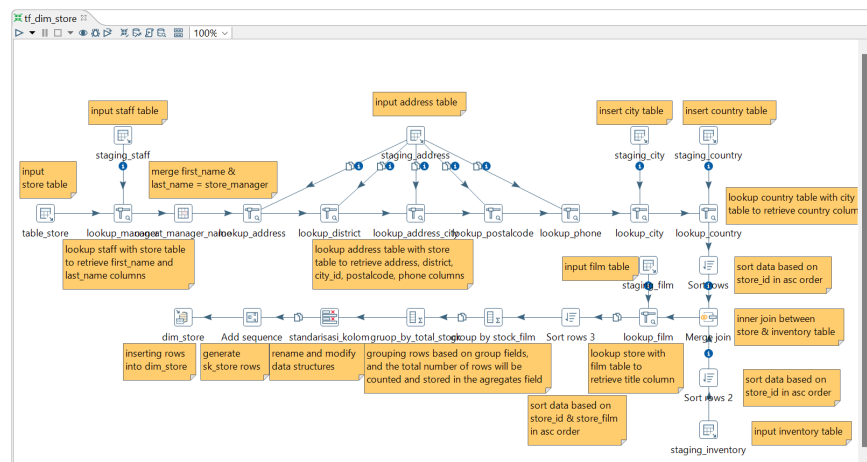
```
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.dim_store
(
  sk_store INTEGER
, store_id INTEGER
, store_manager TEXT
, film_list_inventory TEXT
, total_kind_films INTEGER
, total_stock_inventory INTEGER
, store_address VARCHAR(200)
, store_postalcode VARCHAR(200)
, store_district VARCHAR(200)
, store_city VARCHAR(200)
, store_country VARCHAR(200)
, store_phone VARCHAR(200)
)
;
CREATE INDEX idx_dim_store_lookup ON dwh.dim_store(store_id)
```

Gambar 3.22 *Query* Tabel Dimensi *Store*

b) Proses *data warehouse* untuk penginputan data



Gambar 3.23 Proses *Data Warehouse* Tabel Dimensi *Store*

c) Hasil pembuatan kerangka tabel dan penginputan data

sk_store	store_id	store_manager	film_list_inventory	total_kind_films	total_stock_inventory	store_address	store_postcode
1	1	Mike Hillier	Academy Dinosaur, Affair Prejudice, Agent Truman, Airplane Sierra,	759	2,270	47 MySakila Drive	[NULL]
2	2	Jon Stephens	Academy Dinosaur, Ace Goldfinger, Adaptation Holes, Affair Prejud	762	2,311	28 MySQL Boulevard	[NULL]

Gambar 3.24 Data Tabel Dimensi *Store*

9. Tabel Fakta Rental (fact_rentals)

Tabel ini menyimpan data utama transaksi penyewaan, termasuk hubungan dengan tabel dimensi. Tabel ini digunakan untuk analisis menyeluruh tentang penyewaan, pembayaran, dan durasi transaksi.

a) Query untuk kerangka data

```

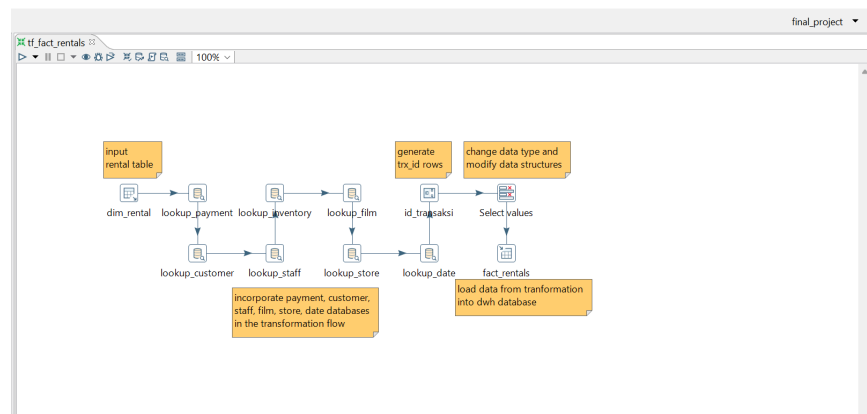
Simple SQL editor

SQL statements, separated by semicolon ';'

CREATE TABLE dwh.fact_rentals
(
  trx_id INTEGER
, sk_rental INTEGER
, sk_payment INTEGER
, sk_customer INTEGER
, sk_inventory INTEGER
, sk_film INTEGER
, sk_store INTEGER
, sk_staff INTEGER
, sk_rentaldate INTEGER
, rental_duration DOUBLE PRECISION
, payment_amount NUMERIC(9, 2)
, payment_date TIMESTAMP
)
;
  
```

Gambar 3.25 Query Tabel Fakta Rental

b) Proses ETL untuk penginputan data



Gambar 3.26 Proses Data Warehouse Tabel Fakta Rental

c) Hasil pembuatan kerangka tabel dan penginputan data

fact_rentals	trx_id	sk_rental	sk_payment	sk_customer	sk_inventory	sk_film	sk_store	sk_staff	sk_rentaldate	rental_duration	payment_amount
1	1	1	[NULL]	131	367	80	1	1	20,050,526	1.9659722222	[N]
2	2	2	[NULL]	460	1,525	333	2	1	20,050,528	3.8652777778	[N]
3	3	3	[NULL]	409	1,711	373	2	1	20,050,601	7.9645833333	[N]
4	4	4	[NULL]	334	2,452	535	1	2	20,050,603	9.1104166667	[N]
5	5	5	[NULL]	223	2,078	450	2	1	20,050,602	8.2277777778	[N]
6	6	6	[NULL]	549	2,792	613	1	1	20,050,527	2.1	[N]
7	7	7	[NULL]	270	3,995	870	2	2	20,050,529	4.8909722222	[N]
8	8	8	[NULL]	240	2,346	510	1	2	20,050,527	3.0013888889	[N]
9	9	9	[NULL]	127	2,580	565	1	1	20,050,528	3.0152777778	[N]
10	10	10	[NULL]	400	1,824	396	2	2	20,050,531	6.0458333333	[N]
11	11	11	[NULL]	143	4,443	971	1	2	20,050,602	8.8659722222	[N]
12	12	12	[NULL]	262	1,584	347	1	2	20,050,530	5.2256944444	[N]
13	13	13	[NULL]	335	2,294	499	1	1	20,050,530	5.1708333333	[N]
14	14	14	[NULL]	447	2,701	593	2	1	20,050,526	1.1006944444	[N]
15	15	15	[NULL]	320	3,049	670	1	1	20,050,603	8.118175	[N]
16	16	16	[NULL]	317	389	86	1	2	20,050,526	1.1659722222	[N]
17	17	17	[NULL]	575	830	181	2	1	20,050,527	1.9840277778	[N]
18	18	18	[NULL]	20	3,376	741	1	2	20,050,531	6.2256944444	[N]
19	19	19	[NULL]	457	1,941	422	2	1	20,050,531	6.1965277778	[N]
20	20	20	[NULL]	186	3,517	770	2	2	20,050,527	2.0222222222	[N]
21	21	21	[NULL]	389	146	31	1	2	20,050,526	0.9597222222	[N]
22	22	22	[NULL]	510	727	159	1	2	20,050,526	1.10625	[N]
23	23	23	[NULL]	439	4,441	971	1	1	20,050,529	4.1625	[N]
24	24	24	[NULL]	351	3,273	721	1	1	20,050,527	1.9319444444	[N]
25	25	25	[NULL]	38	3,961	863	2	2	20,050,527	2.7527777778	[N]
26	26	26	[NULL]	372	4,371	953	1	1	20,050,531	5.8736111111	[N]
27	27	27	[NULL]	302	1,225	271	2	2	20,050,530	4.8972222222	[N]
28	28	28	[NULL]	233	4,068	887	1	2	20,050,526	1.2388888889	[N]

Gambar 3.27 Data Tabel Fakta Rental

3.2 Implementasi *Cube* dan OLAP

3.2.1 *Cube* dan OLAP (dwh)

Pada implementasi *Cube* dan OLAP terdapat *query* dalam file Fact.xml yang digunakan untuk mendefinisikan struktur dasar *cube* OLAP yang akan digunakan dalam analisis data. Query ini memetakan hubungan antara tabel fakta (fact_rentals) dan tabel-tabel dimensi yang relevan. Setiap hubungan didefinisikan secara jelas untuk memastikan data yang dimuat ke *cube* dapat diakses secara multidimensional. File ini menjadi inti dari konfigurasi *cube*, dan memungkinkan pengguna untuk melihat data dari berbagai sudut pandang, seperti analisis berdasarkan pelanggan, film, toko, atau staf. Dengan menggunakan file ini, sistem OLAP dapat memproses data yang ada dalam *data warehouse* untuk mendukung eksplorasi data.

```
<Cube name="Fact" defaultMeasure="Amount">
  <Table name="fact_rentals"/>
```

Gambar 3.28 Query dalam File Fact.xml

Query dalam file Fact.jsp berfungsi sebagai penghubung antara *cube* Mondrian yang telah didefinisikan dalam Fact.xml dan aplikasi OLAP berbasis web. *Query* ini memungkinkan pengguna untuk menampilkan hasil analisis secara visual. Dalam file ini, berbagai operasi OLAP seperti pengambilan data, agregasi, dan pembuatan hierarki dimensi diterjemahkan ke dalam format yang dapat diakses.

```
<%@ page session="true" contentType="text/html; charset=ISO-8859-1" %>
<%@ taglib uri="http://www.tonbeller.com/jpivot" prefix="jp" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<jp:mondrianQuery id="query01" jdbcDriver="com.mysql.jdbc.Driver"
jdbcUrl="jdbc:mysql://localhost/dwh?user=root&password"
catalogUri="/WEB-INF/queries/fact.xml">

select {[Measures].[Total Amount], [Total Sales]} ON COLUMNS,
{[Customers Segmentation], [Film Rating], [Store], [Staff]} ON ROWS
from [Fact]

</jp:mondrianQuery>

<c:set var="title01" scope="session">Query SALES using Mondrian OLAP</c:set>
```

Gambar 3.29 Query dalam File Fact.jsp

Berikut ini adalah hasil dari proses implementasi tabel-tabel dimensi yang relevan ke dalam struktur Mondrian OLAP. Implementasi ini mencakup integrasi setiap tabel dimensi, seperti `dim_customer`, `dim_film`, `dim_store`, dan `dim_staff` ke dalam *cube* OLAP yang telah dirancang. Dengan implementasi ini, data dari setiap tabel dimensi dapat diakses dan dianalisis secara multidimensional, memungkinkan pengguna untuk menggali informasi yang lebih mendalam sesuai kebutuhan analisis.

1. `fact_rentals` (Dimensi *Customer*)

Dimensi *customer* menghubungkan tabel fakta dengan data pelanggan untuk analisis yang mendalam berdasarkan informasi demografi atau perilaku pelanggan. Hubungan antara tabel fakta dan tabel dimensi *customer* ini sangat penting untuk memberikan wawasan yang lebih personal tentang pelanggan dan mendukung strategi pemasaran yang lebih efektif.

a) *Query* pengisian data

```
<Dimension name="Customers Segmentation" foreignKey="sk_customer">
  <Hierarchy hasAll="true" allMemberName="All Customers" primaryKey="sk_customer">
    <Table name="dim_customer" />

    <!-- Menentukan level 1 -->
    <Level name="CustomerCountry" column="customer_country"/>

    <!-- Menentukan level 2 -->
    <Level name="CustomerDistrict" column="customer_district"/>

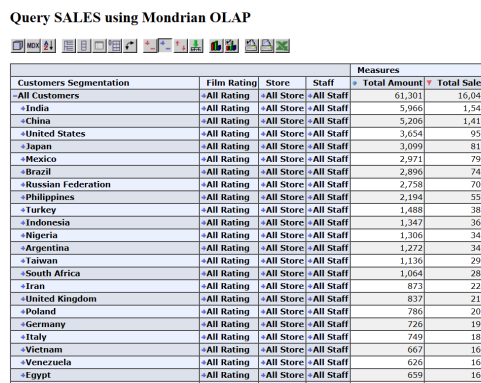
    <!-- Menentukan level 3 -->
    <Level name="CustomerCity" column="customer_city"/>

  </Hierarchy>
</Dimension>
```

Gambar 3.30 *Query* Dimensi *Customer*

b) Hasil implementasi pada Mondrian OLAP

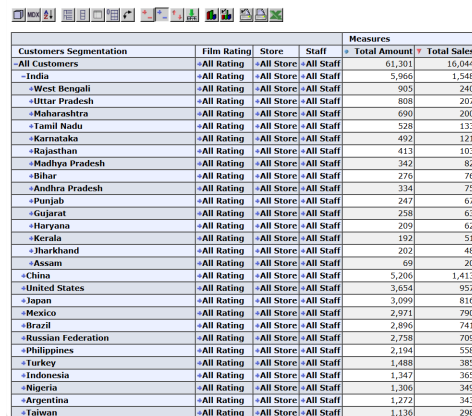
Query SALES using Mondrian OLAP



Customers Segmentation	Film Rating	Store	Staff	Measures	
				Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
India	All Rating	All Store	All Staff	5,566	1,548
China	All Rating	All Store	All Staff	5,206	1,413
United States	All Rating	All Store	All Staff	3,654	957
Japan	All Rating	All Store	All Staff	3,099	816
Mexico	All Rating	All Store	All Staff	2,971	790
Brazil	All Rating	All Store	All Staff	2,896	741
Russian Federation	All Rating	All Store	All Staff	2,758	709
Philippines	All Rating	All Store	All Staff	2,194	558
Turkey	All Rating	All Store	All Staff	1,488	385
Indonesia	All Rating	All Store	All Staff	1,347	365
Nigeria	All Rating	All Store	All Staff	1,306	349
Argentina	All Rating	All Store	All Staff	1,272	343
Taiwan	All Rating	All Store	All Staff	1,136	298
South Africa	All Rating	All Store	All Staff	1,064	284
Iran	All Rating	All Store	All Staff	873	224
United Kingdom	All Rating	All Store	All Staff	837	215
Poland	All Rating	All Store	All Staff	786	203
Germany	All Rating	All Store	All Staff	726	191
Italy	All Rating	All Store	All Staff	749	187
Vietnam	All Rating	All Store	All Staff	667	169
Venezuela	All Rating	All Store	All Staff	626	168
Egypt	All Rating	All Store	All Staff	659	161
Russia	All Rating	All Store	All Staff	676	160

Gambar 3.31 Hasil Implementasi Mondrian OLAP Dimensi
Customer Level Customer Country

Query SALES using Mondrian OLAP

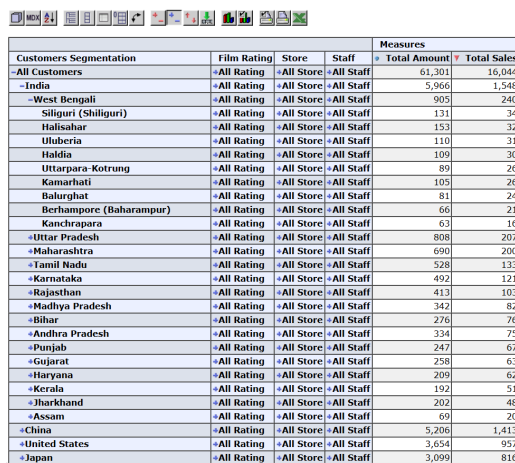


Customers Segmentation	Film Rating	Store	Staff	Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
India	All Rating	All Store	All Staff	5,966	1,548
West Bengal	All Rating	All Store	All Staff	905	240
Uttar Pradesh	All Rating	All Store	All Staff	808	207
Maharashtra	All Rating	All Store	All Staff	690	200
Tamil Nadu	All Rating	All Store	All Staff	528	133
Karnataka	All Rating	All Store	All Staff	492	121
Rajasthan	All Rating	All Store	All Staff	413	103
Madhya Pradesh	All Rating	All Store	All Staff	342	82
Bihar	All Rating	All Store	All Staff	276	76
Andhra Pradesh	All Rating	All Store	All Staff	334	75
Punjab	All Rating	All Store	All Staff	247	67
Gujarat	All Rating	All Store	All Staff	258	63
Haryana	All Rating	All Store	All Staff	209	62
Kerala	All Rating	All Store	All Staff	192	51
Jharkhand	All Rating	All Store	All Staff	202	48
Assam	All Rating	All Store	All Staff	69	20
China	All Rating	All Store	All Staff	5,206	1,413
United States	All Rating	All Store	All Staff	3,654	957
Japan	All Rating	All Store	All Staff	3,099	816
Mexico	All Rating	All Store	All Staff	2,971	790
Brazil	All Rating	All Store	All Staff	2,896	741
Russian Federation	All Rating	All Store	All Staff	2,758	709
Philippines	All Rating	All Store	All Staff	2,194	558
Turkey	All Rating	All Store	All Staff	1,488	385
Indonesia	All Rating	All Store	All Staff	1,347	365
Nigeria	All Rating	All Store	All Staff	1,306	349
Argentina	All Rating	All Store	All Staff	1,272	343
Taiwan	All Rating	All Store	All Staff	1,136	298

Gambar 3.32 Hasil Implementasi Mondrian OLAP Dimensi

Customer Level Customer District

Query SALES using Mondrian OLAP



Customers Segmentation	Film Rating	Store	Staff	Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
India	All Rating	All Store	All Staff	5,966	1,548
West Bengal	All Rating	All Store	All Staff	905	240
Shiliguri (Shiliguri)	All Rating	All Store	All Staff	131	34
Halisahar	All Rating	All Store	All Staff	153	32
Uluberia	All Rating	All Store	All Staff	110	31
Haldia	All Rating	All Store	All Staff	109	30
Uttarpara-Kotrung	All Rating	All Store	All Staff	89	26
Kamarhati	All Rating	All Store	All Staff	105	26
Balughat	All Rating	All Store	All Staff	81	24
Berhampore (Baharampur)	All Rating	All Store	All Staff	66	21
Kanchrapara	All Rating	All Store	All Staff	63	16
Uttar Pradesh	All Rating	All Store	All Staff	808	207
Maharashtra	All Rating	All Store	All Staff	690	200
Tamil Nadu	All Rating	All Store	All Staff	528	133
Karnataka	All Rating	All Store	All Staff	492	121
Rajasthan	All Rating	All Store	All Staff	413	103
Madhya Pradesh	All Rating	All Store	All Staff	342	82
Bihar	All Rating	All Store	All Staff	276	76
Andhra Pradesh	All Rating	All Store	All Staff	334	75
Punjab	All Rating	All Store	All Staff	247	67
Gujarat	All Rating	All Store	All Staff	258	63
Haryana	All Rating	All Store	All Staff	209	62
Kerala	All Rating	All Store	All Staff	192	51
Jharkhand	All Rating	All Store	All Staff	202	48
Assam	All Rating	All Store	All Staff	69	20
China	All Rating	All Store	All Staff	5,206	1,413
United States	All Rating	All Store	All Staff	3,654	957
Japan	All Rating	All Store	All Staff	3,099	816

Gambar 3.33 Hasil Implementasi Mondrian OLAP Dimensi

Customer Level Customer City

2. fact_rentals (Dimensi Film)

Dimensi film memungkinkan analisis performa film berdasarkan *rating* dan judul film. Dalam struktur *cube*, hubungan antara tabel fakta dan dimensi film memungkinkan pengguna untuk melihat data seperti *rating* film mana yang paling sering disewa atau judul film apa yang paling diminati oleh pelanggan.

a) *Query pengisian data*

```
<Dimension name="Film Rating" foreignKey="sk_film">
  <Hierarchy hasAll="true" allMemberName="All Rating" primaryKey="sk_film">
    <Table name="dim_film" />

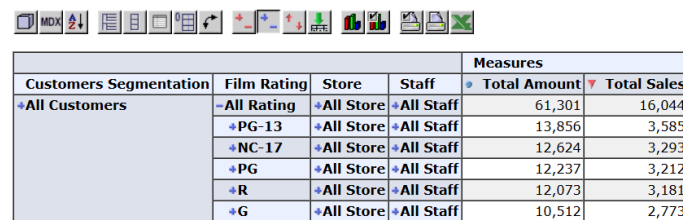
    <!-- Menentukan level 1 -->
    <Level name="FilmRating" column="film_rating"/>

    <!-- Menentukan level 2 -->
    <Level name="FilmTitle" column="film_title"/>
  </Hierarchy>
</Dimension>
```

Gambar 3.34 *Query Dimensi Film*

b) Hasil implementasi pada Mondrian OLAP

Query SALES using Mondrian OLAP



Customers Segmentation	Film Rating	Store	Staff	Measures	
				Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
	PG-13	All Store	All Staff	13,856	3,585
	NC-17	All Store	All Staff	12,624	3,293
	PG	All Store	All Staff	12,237	3,212
	R	All Store	All Staff	12,073	3,181
	G	All Store	All Staff	10,512	2,773

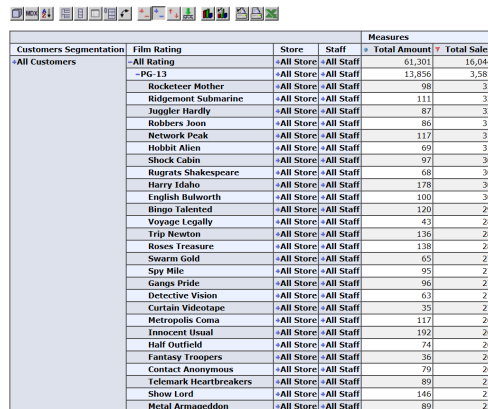
Slicer:

[back to index](#)

Gambar 3.35 Hasil Implementasi Mondrian OLAP Dimensi Film

Level Film Rating

Query SALES using Mondrian OLAP



Customers Segmentation	Film Rating	Store	Staff	Measures	
				Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
	PG-13	All Store	All Staff	13,856	3,585
	Rocketeer Mother	All Store	All Staff	98	33
	Ridgemont Submarine	All Store	All Staff	111	32
	Juggler Hardy	All Store	All Staff	87	32
	Robbers' Noon	All Store	All Staff	86	31
	Network Peak	All Store	All Staff	117	31
	Hobbit Alien	All Store	All Staff	69	31
	Shock Cabin	All Store	All Staff	97	30
	Rugrats Shakespeare	All Store	All Staff	68	30
	Harry Idaho	All Store	All Staff	178	30
	English Bulworth	All Store	All Staff	100	30
	Bingo Talented	All Store	All Staff	120	29
	Voyage Legally	All Store	All Staff	43	28
	Trip Newton	All Store	All Staff	136	28
	Roses Treasure	All Store	All Staff	138	28
	Swarm Gold	All Store	All Staff	65	27
	Spy Mile	All Store	All Staff	95	27
	Gangs Pride	All Store	All Staff	96	27
	Detective Vision	All Store	All Staff	63	27
	Curtain Videotape	All Store	All Staff	35	27
	Metropolis Coma	All Store	All Staff	117	26
	Innocent Usual	All Store	All Staff	192	26
	Half Outfield	All Store	All Staff	74	26
	Fantasy Troopers	All Store	All Staff	36	26
	Contact Anonymous	All Store	All Staff	79	26
	Telemark Heartbreakers	All Store	All Staff	89	25
	Show Lord	All Store	All Staff	146	25
	Metal Armageddon	All Store	All Staff	89	25

Gambar 3.35 Hasil Implementasi Mondrian OLAP Dimensi Film

Level Film Title

3. fact_rentals (Dimensi *Store*)

Dimensi *store* memberikan konteks geografis dan operasional dalam analisis data. Dengan adanya hubungan antara tabel fakta dan dimensi store, pengguna dapat mengevaluasi performa setiap toko. Analisis ini sangat berguna untuk mengidentifikasi toko dengan kinerja terbaik atau toko yang memerlukan perbaikan dalam operasionalnya.

a) *Query* pengisian data

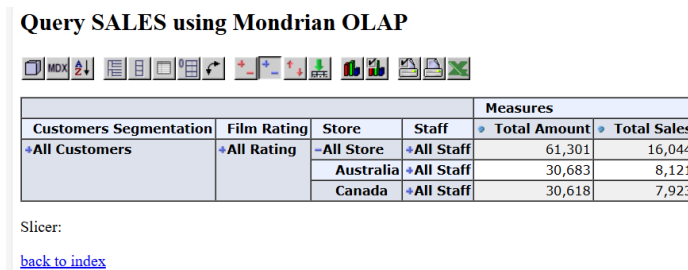
```
<Dimension name="Store" foreignKey="sk_store">
  <Hierarchy hasAll="true" allMemberName="All Store" primaryKey="sk_store">
    <Table name="dim_store" />

    <!-- Menentukan level 1" -->
    <Level name="StoreCountry" column="store_country"/>
  </Hierarchy>
</Dimension>
```

Gambar 3.36 *Query* Dimensi *Store*

b) Hasil implementasi pada Mondrian OLAP

Query SALES using Mondrian OLAP



				Measures	
Customers Segmentation	Film Rating	Store	Staff	Total Amount	Total Sales
+All Customers	+All Rating	+All Store	+All Staff	61,301	16,044
		Australia	+All Staff	30,683	8,121
		Canada	+All Staff	30,618	7,923

Slicer:
[back to index](#)

Gambar 3.37 Hasil Implementasi Mondrian OLAP Dimensi *Store*

4. fact_rentals (Dimensi *Staff*)

Dimensi *staff* memungkinkan analisis kinerja staf berdasarkan transaksi yang mereka kelola. Dalam *cube*, hubungan ini membantu pengguna memahami kontribusi setiap staf terhadap pendapatan. Dimensi ini juga mendukung evaluasi sumber daya manusia, seperti mengidentifikasi staf dengan kinerja paling tinggi atau paling rendah.

a) *Query* pengisian data

```
<Dimension name="Staff" foreignKey="sk_staff">
  <Hierarchy hasAll="true" allMemberName="All Staff" primaryKey="sk_staff">
    <Table name="dim_staff" />

    <!-- Menentukan level 1" -->
    <Level name="StaffName" column="staff_name"/>
  </Hierarchy>
</Dimension>
```

Gambar 3.38 *Query* Dimensi *Staff*

b) Hasil implementasi pada Mondrian OLAP

Query SALES using Mondrian OLAP



				Measures	
Customers Segmentation	Film Rating	Store	Staff	Total Amount	Total Sales
All Customers	All Rating	All Store	All Staff	61,301	16,044
			Jon Stephens	30,513	7,906
			Mike Hillyer	30,270	7,955

Slicer:

[back to index](#)

Gambar 3.39 Hasil Implementasi Mondrian OLAP Dimensi *Staff*

BAB IV KESIMPULAN DAN SARAN

4.1 Kesimpulan

Project pengembangan *data warehouse* dan implementasi OLAP ini berhasil menyediakan sistem yang terintegrasi untuk analisis data DVD Rental. Dengan memanfaatkan berbagai tabel dimensi, data yang sebelumnya belum terstruktur dapat dikelola secara terstruktur di dalam *data warehouse*. Implementasi *cube* OLAP menggunakan Mondrian memungkinkan analisis multidimensional yang mendalam, sehingga mendukung pengambilan keputusan strategis. Secara keseluruhan, sistem ini memberikan manfaat nyata dalam memahami pola transaksi, preferensi pelanggan, serta kinerja toko dan staf.

4.2 Saran

Pengembangan lebih lanjut dapat difokuskan pada pengintegrasian data secara *real-time*, yang memungkinkan sistem untuk mengolah dan menyajikan informasi terbaru secara langsung tanpa adanya jeda waktu. Dengan adanya integrasi data secara *real-time*, sistem ini dapat memberikan hasil analisis yang lebih akurat dan sesuai dengan kondisi terkini, sehingga mendukung pengambilan keputusan yang lebih cepat dan tepat. Langkah ini juga akan memastikan bahwa sistem *data warehouse* dan OLAP yang telah dikembangkan tetap relevan dan mampu memenuhi kebutuhan analisis data yang semakin kompleks.

LAMPIRAN

Lampiran 1. Link Github

https://github.com/fidyaahs/Konversi_DWO_22082010047_Fidyah-Salsabila-Putri-Sillehu.git