

PROYEK AKHIR
“PENGEMBANGAN DATA WAREHOUSE DAN DASHBOARD”
SEMESTER GANJIL 2024/2025



KELAS A

DOSEN PENGAMPU :

MOHAMAD IRWAN AFANDI, ST.,M.Sc

DISUSUN OLEH :

RISMA PARAMESTI (22082010014)

DWI SEPTIAJAYANTI (22082010042)

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA TIMUR

2024

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, kami dapat menyelesaikan laporan proyek akhir yang berjudul "Pengembangan Data Warehouse dan Dashboard". Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan mata kuliah Data Warehouse & Olap di Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur.

Penyusunan laporan ini tidak terlepas dari bantuan dan dukungan berbagai pihak. Oleh karena itu, kami ingin mengucapkan terima kasih kepada:

1. Bapak **MOHAMAD IRWAN AFANDI, ST.,M.Sc**, selaku dosen pengampu mata kuliah Data Warehouse & Olap yang telah memberikan bimbingan dan arahan dalam penyusunan laporan ini.
2. Rekan tim proyek akhir, yang telah bekerja sama dengan penuh dedikasi dan semangat untuk menyelesaikan laporan ini

Kami menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun sangat kami harapkan untuk perbaikan di masa mendatang. Semoga laporan ini dapat memberikan manfaat bagi pembaca.

Surabaya, 27 Desember 2024

Penyusun

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I.....	1
PENDAHULUAN.....	1
A. Latar Belakang.....	1
B. Tujuan.....	2
C. Manfaat.....	3
BAB II.....	4
METODOLOGI.....	4
A. Metodologi.....	4
B. Landasan Teori.....	6
C. Sumber Data & Tools Yang Digunakan.....	7
BAB III.....	10
HASIL & PEMBAHASAN.....	10
A. Implementasi atau Model Multidimensi.....	10
B. Implementasi ETL & Olap.....	25
C. Implementasi Dashboard & Hasil Insight Analisis.....	36
BAB IV.....	43
KESIMPULAN.....	43
A. Kesimpulan.....	43
B. Saran.....	43
LAMPIRAN.....	44

BAB I

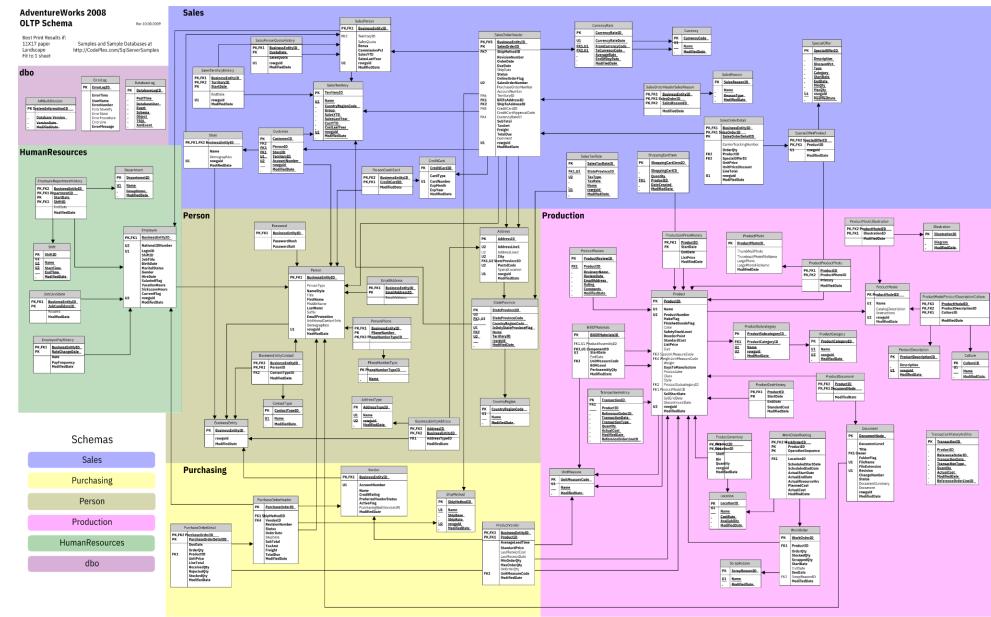
PENDAHULUAN

A. Latar Belakang

Seiring dengan perkembangan organisasi yang terus meningkat, kebutuhan akan pengelolaan data juga semakin bertambah secara signifikan. Setiap langkah kemajuan dalam suatu organisasi, baik dalam skala kecil maupun besar, umumnya diiringi dengan pertumbuhan data yang eksponensial. Pertumbuhan data ini meliputi berbagai aspek, mulai dari data operasional harian, informasi pelanggan, laporan keuangan, hingga data strategis lainnya yang mendukung pengambilan keputusan. Jika tidak dikelola dengan baik, akumulasi data yang besar dan kompleks ini dapat memberikan tekanan pada sistem basis data yang digunakan. Kondisi tersebut dapat mengakibatkan penurunan kinerja sistem secara keseluruhan, memperlambat waktu respons, dan menimbulkan hambatan dalam pengolahan data yang diperlukan untuk operasional sehari-hari. Hal ini menjadi tantangan yang semakin penting, terutama ketika sistem basis data organisasi berkembang menjadi lebih besar, dengan struktur yang semakin rumit serta kebutuhan akan kecepatan dan akurasi yang lebih tinggi. Oleh karena itu, diperlukan upaya yang terencana dan berkesinambungan untuk memastikan bahwa kinerja sistem basis data tetap optimal, sekaligus mampu mendukung kebutuhan organisasi secara efektif di masa depan.

Perusahaan Microsoft menyediakan **Database AdventureWorks 2019** sebagai contoh *database* yang dirancang untuk merepresentasikan data bisnis *fiktif* dalam skala yang cukup *realistik*. Adventure Works 2019 merupakan perusahaan grosir sepeda dengan memiliki 97 merk sepeda yang dikelompokan ke dalam tiga kategori yaitu sepeda gunung, sepeda jalan raya, dan sepeda tour. Adventure works tidak hanya menjual sepeda, tetapi juga menyediakan aksesoris, pakaian, dan komponen. Aksesoris yang tersedia seperti botol, rak, sepeda, rem, dan lain-lain. Pakaian tersedia seperti topi, sarung tangan, jersey, dan lain-lain. Sedangkan untuk komponen, Adventure works menjual rem, rantai, derailleur, dan lain-lain.

Adventure works melayani pelanggan secara global, termasuk Australia, Kanada, Perancis, Jerman, Inggris, Amerika Serikat.



Gambar 1.1 Struktur Database AdventureWorks

Pada proyek ini penyusun ingin mengembangkan Data Warehouse yang mengambil data dari database Adventure works 2019. Tabel utama di database Adventure Works yang menjadi sumber data adalah tabel **sales_salesorderdetail** dan tabel **purchase_purchaseorderdetail** yang memiliki data **OrderQty** (jumlah pembelian), **LineTotal** (Total harga pembelian) yang merupakan measure yang dibutuhkan. Selain itu hasil dari data warehouse yang sudah dibangun di visualisasikan dengan membuat dashboard. Dashboard digunakan untuk mempermudah dalam menganalisis dan pengambilan keputusan. Terdapat 2 yang penyusun analisis yaitu **analisis penjualan produk** dan **analisis purchase** yang dilakukan oleh **sales**.

B. Tujuan

1. Mengembangkan data dari database Adventure Works 2019 yang relevan dan terintegrasi untuk keperluan analisis penjualan dan pembelian produk.

2. Membuat dashboard sebagai alat visualisasi data untuk mendukung pengambilan keputusan yang cepat dan akurat.
3. Memberikan insight mendalam tentang pola penjualan produk dan pembelian oleh tim sales melalui analisis data berbasis dashboard.

C. Manfaat

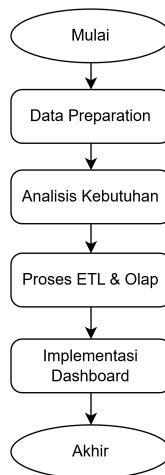
1. **Kemudahan Analisis Data**, Dengan data warehouse yang dirancang khusus, proses analisis data menjadi lebih cepat, akurat, dan terorganisir, sehingga mendukung kebutuhan bisnis.
2. **Peningkatan Pengambilan Keputusan**, Dashboard menyediakan visualisasi yang mudah dipahami, membantu manajemen dalam mengambil keputusan yang berbasis data (data-driven decision-making).
3. **Efisiensi Operasional**, Data warehouse mengurangi redundansi dan kompleksitas pengelolaan data, sehingga mendukung efisiensi operasional perusahaan.

BAB II

METODOLOGI

A. Metodologi

Pada bab ini, dijelaskan mengenai tahapan-tahapan yang dilakukan dalam projek ini, yang meliputi langkah-langkah dalam pengembangan data warehouse dan dashboard berdasarkan studi kasus pada database AdventureWorks 2019. Proses ini terbagi menjadi beberapa fase, mulai data preparation, analisis kebutuhan, proses ETL & Olap, hingga implementasi dashboard. Adapun alur penelitian ini terdiri dari tahapan-tahapan sebagai berikut:



Gambar 2.1 Alur Tahapan Pengerjaan

a. Data Preparation

Pada tahap ini fokus utama adalah mempersiapkan data yang akan digunakan dalam pengembangan data warehouse. Data Adventure Works 2019 di *extract* dan di masukkan ke dalam *tools* dbeaver untuk analisis lebih lanjut. Selain itu, Pada tahap ini juga dilakukan pemodelan awal struktur data untuk memudahkan integrasi ke dalam data warehouse.

b. Analisis Kebutuhan

Tahap ini bertujuan untuk memahami secara mendalam serta tujuan yang ingin dicapai dari sistem yang dikembangkan. Kebutuhan informasi diidentifikasi dengan menganalisis data sumber yang tersedia dalam database *Adventure Works*. Dari hasil analisis ini, ditentukan jenis laporan, visualisasi yang akan digunakan dalam dashboard. Selain itu, pada tahap ini dirancang skema data warehouse menggunakan model seperti star schema agar struktur data lebih optimal untuk mendukung proses analisis.

c. Proses ETL dan Olap

Proses ETL (Extract, Transform, Load) dilakukan untuk mengekstrak data dari database *Adventure Works*, mentransformasi data sesuai kebutuhan analisis, dan memuat data tersebut ke dalam data warehouse dengan menggunakan tools pentaho. Data yang sudah dimasukkan ke dalam data warehouse kemudian diolah menggunakan konsep OLAP (Online Analytical Processing), yang memungkinkan pengaturan data dalam format multidimensi sehingga lebih mudah untuk dianalisis secara cepat dan mendalam. Tahap ini memastikan bahwa data warehouse siap digunakan sebagai sumber utama dalam pengembangan dashboard.

d. Implementasi Dashboard

Implementasi Dashboard, di mana data warehouse yang telah dibangun diintegrasikan. Pada tahap ini, dashboard dirancang dengan menyusun elemen-elemen visual yang sesuai dengan kebutuhan analisis yang telah dilakukan, seperti grafik, tabel, dan lainnya. Dashboard ini dirancang agar dapat menyajikan informasi secara real-time, akurat, dan mudah dipahami oleh pengguna.

B. Landasan Teori

a. Database

Database adalah kumpulan data yang terorganisir dengan baik dan terstruktur sedemikian rupa sehingga memungkinkan pengelolaan, penyimpanan, dan pengambilan kembali data secara efisien. Database didefinisikan sebagai suatu koleksi data yang saling berkaitan dan disusun untuk memenuhi berbagai kebutuhan. Data yang disimpan dalam database umumnya memiliki keterkaitan yang membentuk suatu sistem informasi, yang dapat digunakan untuk mendukung proses pengambilan keputusan. Secara umum, database tidak hanya menyimpan data dalam jumlah besar, tetapi juga memungkinkan manipulasi data seperti menambah, menghapus, atau memperbarui informasi dengan cepat dan akurat. Sehingga, database menjadi komponen penting dalam pengembangan sistem informasi, baik itu untuk keperluan administratif, bisnis, maupun analisis data.

b. Data Warehouse

Data warehouse adalah koleksi data yang mempunyai sifat subject-oriented, Integrated, nonvolatile, dan time-variant untuk mendukung proses pengambilan keputusan dalam manajemen. Menurut Mannino, data warehouse adalah tempat penyimpanan data terpusat, dimana data dari basis data operasional dan sumber lainnya diintegrasikan, dibersihkan dan diarsipkan untuk mendukung pengambilan keputusan. Menurut McLeod, data warehouse adalah sebuah sistem penyimpanan data yang berkapasitas besar, dimana data dikumpulkan dengan menambahkan record baru daripada meng-update record yang sudah ada dengan informasi baru. Data jenis ini digunakan hanya untuk proses pengambilan keputusan dan bukan untuk kegiatan operasional perusahaan sehari-hari

C. Sumber Data & Tools Yang Digunakan

Tabel data yang digunakan dalam proyek ini adalah sebagai berikut :

1. **Sales.SalesOrderHeader:** Berisi informasi tentang pesanan penjualan, termasuk tanggal pemesanan, tanggal pengiriman, dan total penjualan.
2. **Sales.SalesOrderDetail:** Menyediakan rincian tentang produk individual yang dijual di setiap pesanan penjualan, seperti jumlah, harga, dan diskon.
3. **Sales.Customer:** Menyimpan informasi terkait pelanggan, seperti customerID, storeID, territoryID, dan nomor akun.
4. **Sales.SalesPerson:** Berisi informasi tentang tenaga penjualan yang terkait dengan perusahaan. Termasuk kolom seperti BusinessEntityID, TerritoryID, SalesQuota, SalesYTD.
5. **Sales.SalesTerritory:** Berisi informasi tentang berbagai wilayah dan kawasan penjualan. Termasuk kolom-kolom seperti TerritoryID, Nama wilayah, CountryRegionCode
6. **Person.Person:** Berisi informasi tentang individu yang terkait dengan perusahaan seperti nama lengkap, jenis kelamin, alamat email, nomor telepon, dan detail pribadi lainnya.
7. **Production.Produk:** Memberikan rincian tentang produk, termasuk nama dan deskripsi lainnya.
8. **Production.ProductCategory:** Berisi informasi tentang kategori produk
9. **Purchasing.Vendor:** Tabel ini berisi informasi tentang pemasok, termasuk nama dan informasi kontak. Tabel ini terhubung ke tabel Purchasing.PurchaseOrderHeader melalui ID vendor.
10. **Purchasing.PurchaseOrderHeader:** Tabel ini berisi informasi tentang purchase order, termasuk tanggal order, nomor order, dan ID pemasok. Tabel ini terhubung ke tabel Purchasing.Vendor melalui ID pemasok. Pembelian.Tabel vendor melalui ID pemasok.
11. **Purchase.PurchaseOrderDetail:** Tabel ini berisi informasi terperinci tentang pesanan pembelian, termasuk ID pesanan, ID produk, kuantitas, harga satuan, dan diskon.

12. **Purchasing.shipmethod** : berisi informasi tentang metode pengiriman atau perusahaan pengiriman dalam sistem. Tabel ini berisi detail biaya dasar pengiriman, biaya per berat (pound).

Dalam proses pengerjaan kami menggunakan beberapa tools untuk mempermudah proses analisis data, mulai dari memasukkan data atau melakukan re-store data, pengolahan data, hingga visualisasi hasil analisis. Berikut tools yang kami gunakan dan alasan pemilihannya.

a. Xampp

XAMPP adalah sebuah perangkat lunak yang berfungsi sebagai server lokal untuk pengembangan website. Nama XAMPP merupakan singkatan dari Cross-platform (X), Apache (A), MySQL/MariaDB (M), PHP (P), dan Perl (P). Tools ini menyediakan lingkungan pengembangan web lokal di mana pengembang dapat membuat, menguji, dan menjalankan aplikasi berbasis PHP dan database MySQL/MariaDB secara offline sebelum mengunggahnya ke server online. Dengan XAMPP, server Apache dan layanan database dapat berjalan secara simultan sehingga sangat memudahkan dalam membangun dan mengembangkan aplikasi web.

b. Dbeaver

DBeaver adalah alat manajemen database universal yang mendukung berbagai jenis database seperti MySQL, PostgreSQL, Oracle, SQL Server, SQLite, dan lainnya. DBeaver menyediakan antarmuka grafis yang memudahkan pengguna dalam mengelola database, menulis dan menjalankan query SQL, serta melakukan analisis dan visualisasi data. Tools ini memungkinkan pengguna untuk melakukan operasi kompleks seperti ekspor dan impor data, pembuatan skema database, serta pemantauan struktur dan performa database dengan efisien.

c. Pentaho

Pentaho adalah perangkat lunak yang digunakan dalam solusi Business Intelligence (BI) dan pengolahan data, terutama untuk proses ETL (Extract, Transform, Load). Pentaho membantu dalam mengambil data dari berbagai sumber, mengolah atau mengubah data menjadi format yang diinginkan, dan memuatnya ke dalam database atau data warehouse. Selain itu, Pentaho memungkinkan pembuatan laporan bisnis, dashboard visual, serta analisis data untuk mendukung pengambilan keputusan berbasis data. Tools ini sering digunakan oleh perusahaan untuk mengintegrasikan data dan menyediakan informasi yang relevan dalam bentuk yang mudah dipahami.

d. Visual Studio Code

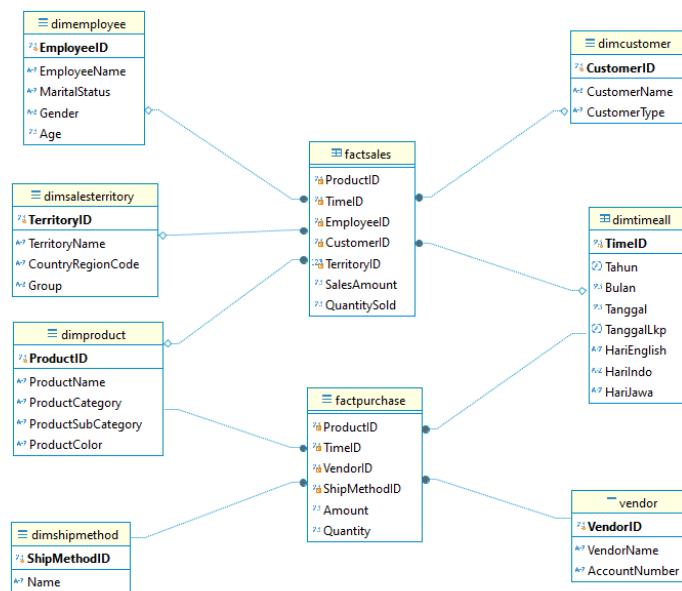
Visual Studio Code atau sering disebut VS Code adalah editor kode sumber yang dikembangkan oleh Microsoft. VS Code dikenal sebagai editor kode yang ringan, cepat, dan mendukung berbagai bahasa pemrograman seperti Python, JavaScript, PHP, Java, dan banyak lagi. Selain itu, VS Code dilengkapi dengan fitur-fitur canggih seperti intellisense (autocompletion dan suggestion), debugging, serta integrasi Git. Dengan dukungan ekstensi yang beragam, pengguna dapat menambahkan fungsionalitas sesuai kebutuhan mereka, seperti linting, integrasi tools eksternal, dan manajemen proyek, menjadikannya alat yang ideal untuk pengembangan aplikasi web, desktop, maupun mobile.

BAB III

HASIL & PEMBAHASAN

A. Implementasi atau Model Multidimensi

Pada tahap ini kami mendesain atau memodelkan data warehouse yang akan dikembangkan. Ada 2 kebutuhan analisis yang dibangun yaitu performa sales yang dilakukan oleh customer, dan performa purchase yang dilakukan oleh employee. Desain data warehouse dapat dilihat pada Gambar 3.1 sebagai berikut :



Gambar 3.1 *Design Star-schema Data Warehouse*

Pada Gambar 3.1 menunjukkan desain model data warehouse dalam bentuk star-schema. Adapun tabel dimensi dan tabel measure atau fakta yang dibuat yaitu :

1. Tabel Dimensi

Struktur tabel dimensi sebagai berikut :

a. Tabel dimcustomer

```
CREATE TABLE DimCustomer (
    CustomerID INT PRIMARY KEY,
    CustomerName VARCHAR(100),
```

```
CustomerType CHAR(1)  
);
```

b. Tabel dimemployee

```
CREATE TABLE DimEmployee (  
    EmployeeID INT PRIMARY KEY,  
    EmployeeName VARCHAR(100),  
    MaritalStatus VARCHAR(1),  
    Gender VARCHAR(1),  
    Age INT  
);
```

c. Tabel dimproduct

```
CREATE TABLE DimProduct (  
    ProductID INT PRIMARY KEY,  
    ProductName VARCHAR(100),  
    ProductCategory VARCHAR(50),  
    ProductSubCategory VARCHAR(50),  
    ProductColor VARCHAR(30)  
);
```

d. Tabel dimsalesterritory

```
CREATE TABLE DimSalesTerritory (  
    TerritoryID INT PRIMARY KEY,  
    TerritoryName VARCHAR(100),  
    CountryRegionCode VARCHAR(5),  
    Group VARCHAR(50)  
);
```

e. Tabel dimtimeall

```
CREATE TABLE dimtimeall (
    TimeID INT PRIMARY KEY,
    Tahun YEAR,
    Bulan INT,
    Tanggal INT,
    TanggalLkp DATE,
    HariEnglish VARCHAR(15),
    HariIndo VARCHAR(15),
    HariJawa VARCHAR(15)
);
```

f. Tabel vendor

```
CREATE TABLE vendor(
    VendorID INT,
    VendorName varchar(225)
);
```

g. Tabel dimshipmethod

```
CREATE TABLE DimShipMethod (
    ShipMethodID INT primary key,
    Name varchar(100)
);
```

2. Analisis kebutuhan data tabel dimensi dari sumber data pada database AdventureWorks

a. Tabel dimcustomer

Data yang dibutuhkan	Sumber data	Keterangan
CustomerID	Tabel sales_customer kolom PersonID	CustomerID diperoleh langsung dari tabel

	→ CusttomerID	sales_customer kolom PersonID-
CustomerName	Tabel person_person kolom FirstName dan LastName → CustomerName	Menggunakan fungsi CONCAT untuk menggabungkan FirstName dan LastName dan dilakukan join antara tabel sales_customer dan tabel person_person
CustomerType	Tabel person_person kolom PersonType → CustomerType	CustomerType diperoleh langsung dari tabel sales_customer kolom PersonType-

Query:

```

INSERT INTO projectuas.dimcustomer
(customerID, CustomerName, CustomerType)
SELECT
    sc.PersonID AS CustomerID,
    CONCAT(pp.FirstName, ' ', pp.LastName)
AS CustomerName,
    pp.PersonType AS CustomerType
FROM
    sales_customer sc
INNER JOIN
    person_person pp ON sc.PersonID =
pp.BusinessEntityID
WHERE
    sc.PersonID IS NOT NULL;

```

b. Tabel dimemployee

Data yang dibutuhkan	Sumber data	Keterangan
EmployeeID	Tabel Salessales_person kolom BusinessEntityID → EmployeeID	EmployeeID diperoleh langsung dari tabel salessales_person kolom BusinessEntityID dan dilakukan join antara tabel salessales_person dan tabel humanresources_employee
EmployeeName	Tabel person_person kolom FirstName, MiddleName & LastName → EmployeeName	Menggunakan fungsi CONCAT untuk menggabungkan FirstName, MiddleName dan LastName dari tabel person_person
MaritalStatus	Tabel humanresources_employee → MaritalStatus	EmployeeID diperoleh langsung dari tabel humanresources_employee kolom MaritalStatus
Gender	Tabel humanresources_employee → Gender	EmployeeID diperoleh langsung dari tabel humanresources_employee kolom Gender
Age	Tabel humanresources_employee → BirthDate	Menggunakan fungsi TIMESTAMPDIFF untuk menghitung umur berdasarkan

		tanggal lahir dari kolom BirthDate dengan tanggal acuan '2001-01-01'
--	--	--

Query:

```

INSERT INTO projectuas.dimEmployee
(EmployeeID, EmployeeName, MaritalStatus,
Gender, Age)
SELECT
    ss.BusinessEntityID,
    CONCAT(pp.FirstName, " ",
COALESCE(pp.MiddleName, ""), " ",
pp.LastName) AS FullName,
    he.MaritalStatus,
    he.Gender,
    TIMESTAMPDIFF(YEAR, he.BirthDate,
'2001-01-01') AS Age
FROM
    sales_salesperson ss
JOIN
    humanresources_employee he
    ON ss.BusinessEntityID =
he.BusinessEntityID
JOIN
    person_person pp
    ON pp.BusinessEntityID =
he.BusinessEntityID;

```

c. Tabel dimproduct

Data yang dibutuhkan	Sumber data	Keterangan
ProductID	Tabel production_product → ProductID	ProductID diperoleh langsung dari tabel production_product
ProductName	Tabel production_product kolom Name → ProductName	ProductID diperoleh langsung dari tabel production_product kolom Name.

ProductCategory	Tabel production_productcategory kolom Name → ProductCategory	Harus melakukan join antara tabel production_product, production_productcategory, dan production_products ubcategory.
ProductSubCategory	Tabel production_productsubcategory kolom Name → ProductSubCategory	Harus melakukan join antara tabel production_product dan production_productcategory.
ProductColor	Tabel production_product kolom Color → ProductColor	ProductColor diperoleh langsung dari tabel production_product kolom Color.

Query:

```

INSERT INTO projectuas.dimProduct
(ProductID, ProductName, ProductCategory,
ProductSubCategory, ProductColor)
SELECT
    p.ProductID,
    p.Name AS ProductName,
    pc.Name AS ProductCategory,
    psc.Name AS ProductSubCategory,
    p.Color AS ProductColor
FROM
    production_product p
LEFT JOIN
    production_productsubcategory psc ON
    p.ProductSubcategoryID =
    psc.ProductSubcategoryID
LEFT JOIN
    production_productcategory pc ON
    psc.ProductCategoryID =
    pc.ProductCategoryID;

```

d. Tabel dimsalesterritory

Data yang dibutuhkan	Sumber data	Keterangan
TerritoryID	Tabel sales_salesterritory → TerritoryID	TerritoryID diperoleh langsung dari tabel sales_salesterritory
TerritoryName	Tabel sales_salesterritory kolom Name → TerritoryName	TerritoryName diperoleh langsung dari tabel sales_salesterritory kolom Name
CountryRegionCode	Tabel sales_salesterritory → CountryRegionCode	CountryRegionCode diperoleh langsung dari tabel sales_salesterritory kolom CountryRegionCode
Group	Tabel sales_salesterritory → Group	Group diperoleh langsung dari tabel sales_salesterritory kolom Group

Query:

```

INSERT INTO projectuas.dimsalesterritory
(TerritoryID , TerritoryName ,
CountryRegionCode, Group )
SELECT
    st.TerritoryID,
    st.Name AS TerritoryName,
    st.CountryRegionCode,
    st.Group
FROM
    sales_salesterritory st;

```

e. Tabel dimtimeall

Pada tabel dimtimeall data satuan waktu terkecil yang dibutuhkan adalah harian, artinya hanya sampai tanggal. Tabel dimensi time adalah tabel acuan yang tidak tergantung pada data transaksi, akan tetapi untuk baris pertama-nya perlu dilihat data transaksi di dalam database adventureworks pada tabel salesorderdetail kolom ModifiedDate.

Dengan perintah SQL berikut ini akan diketahui tanggal paling awal dalam ModifiedDate yaitu tanggal 16 April 2014 :

```
SELECT ModifiedDate  
FROM salesorderdetail  
ORDER BY ModifiedDate ASC  
LIMIT 1;
```

Sedangkan tanggal paling akhir dalam tabel rental yaitu tanggal 22 September 2014 :

```
SELECT ModifiedDate  
FROM salesorderdetail  
ORDER BY ModifiedDate DESC  
LIMIT 1;
```

Jadi data dalam tabel timedate akan dimulai pada **16 April 2014** sampai tanggal **22 September 2014**. Setelah mengetahui waktu awal dan akhir, lalu melakukan pengisian data dengan menggunakan Stored Procedures, buat procedures dengan nama ‘isitime’ lalu masukan query berikut :

```
CREATE DEFINER=root@localhost PROCEDURE  
projectuas.isitimeall(  
    IN waktuMulai DATE,  
    IN waktuSelesai DATE,  
    IN HariJawa VARCHAR(20),
```

```

        IN HariIndo VARCHAR(20)    -- Tambahkan
        input untuk HariIndo
    )
begin

        -- Label untuk loop
    mulaiLoop: LOOP

        -- Memasukkan data ke dalam tabel
    dimtime
        INSERT INTO dimtimeall (Tahun,
        Bulan, Tanggal, TanggalLkp, HariEnglish,
        HariIndo, HariJawa)
        VALUES (
            YEAR(waktuMulai),
            MONTH(waktuMulai),
            DAY(waktuMulai),
            waktuMulai,
            DAYNAME(waktuMulai),
            HariIndo,
            HariJawa
        );

        -- Menambah satu hari ke
    waktuMulai
        SET waktuMulai =
DATE_ADD(waktuMulai, INTERVAL 1 DAY);

```

```

-- Mengubah HariJawa berdasarkan
siklus 5 hari
SET HariJawa =
CASE
WHEN HariJawa = 'pon'
THEN 'wage'
WHEN HariJawa = 'wage'
THEN 'kliwon'
WHEN HariJawa = 'kliwon'
THEN 'legi'
WHEN HariJawa = 'legi'
THEN 'pahing'
WHEN HariJawa = 'pahing'
THEN 'pon'
END;

```

```

-- Mengubah HariIndo berdasarkan
siklus 7 hari
SET HariIndo =
CASE
WHEN HariIndo = 'Minggu'
THEN 'Senin'
WHEN HariIndo = 'Senin'
THEN 'Selasa'
WHEN HariIndo = 'Selasa'
THEN 'Rabu'
WHEN HariIndo = 'Rabu'
THEN 'Kamis'
WHEN HariIndo = 'Kamis'
THEN 'Jumat'

```

```

WHEN HariIndo = 'Jumat'
THEN 'Sabtu'

WHEN HariIndo = 'Sabtu'
THEN 'Minggu'

END;

-- Memeriksa apakah waktuMulai
melebihi waktuSelesai
IF DATEDIFF(waktuMulai,
waktuSelesai) > 0 THEN
    LEAVE mulaiLoop; -- Keluar
dari loop jika waktuMulai sudah lebih
dari waktuSelesai
END IF;

END LOOP mulaiLoop;
END

```

Jika sudah save, dan panggil :

```

CALL projectuas.isitimeall('2011-04-16',
'2014-09-22', 'wage', 'sabtu');

```

f. Tabel vendor

Data yang dibutuhkan	Sumber data	Keterangan
VendorID	Tabel purchasing_vendor kolom BusinessEntityID → VendorID	VendorID diperoleh langsung dari tabel purchasing_vendor kolom BusinessEntityID
VendorName	Tabel	VendorName

	purchasing_vendor kolom Name → VendorName	diperoleh langsung dari tabel purchasing_vendor kolom Name
--	---	---

Query:

```
INSERT INTO projectuas.vendor (VendorID,
VendorNam)
SELECT
P.BusinessEntityID as VendorID,
p.Name as VendorName
FROM
purchasing_vendor p;
```

g. Tabel dimshipmethod

Data yang dibutuhkan	Sumber data	Keterangan
ShipMethodID	Tabel purchasing_shipm ethod → ShipMethodID	ShipMethodID diproleh langsung dari tabel purchasing_shipmet hod
Method_Name	Tabel purchasing_shipm ethod kolom Name → Method_Name	Method_Name diproleh langsung dari tabel purchasing_shipmet hod kolom Name

Query:

```
INSERT INTO projectuas.dimshipmethod
(ShipMethodID, Name)
SELECT
    p.ShipMethodID,
    p.Name as Method_Name
FROM
    purchasing_shipmethod p;
```

3. Tabel Fakta

1. Tabel factsales

```
CREATE TABLE FactSales (
    ProductID INT,
    TimeID INT,
    EmployeeID INT,
    CustomerID INT,
    TerritoryID INT,
    SalesAmount DOUBLE,
    QuantitySold SMALLINT(6),
    FOREIGN KEY (ProductID) REFERENCES DimProduct(ProductID),
    FOREIGN KEY (TimeID) REFERENCES DimTimeAll(TimeID),
    FOREIGN KEY (EmployeeID) REFERENCES DimEmployee(EmployeeID),
    FOREIGN KEY (CustomerID) REFERENCES DimCustomer(CustomerID),
    FOREIGN KEY (TerritoryID) REFERENCES DimSalesTerritory(TerritoryID)
);
```

2. Tabel factpurchase

```
CREATE TABLE FactPurchase (
    ProductID INT NOT NULL,
    TimeID INT NOT NULL,
    VendorID INT NOT NULL,
    ShipMethodID INT not null,
    Amount DOUBLE,
    Quantity SMALLINT,
```

```

        FOREIGN KEY (ProductID) REFERENCES
DimProduct (ProductID),
        FOREIGN KEY (TimeID) REFERENCES
DimTimeAll (TimeID),
        FOREIGN KEY (VendorID) REFERENCES
vendor (VendorID),
    foreign key (ShipMethodID) references
dimshipmethod (ShipMethodID)
);

```

3. Relevansi Tabel fakta dengan tabel dimensi

a. Tabel factsales :

Untuk relevansi lakukanlah dengan perumpamaan sebagai berikut :

“ Pada Tanggal 26 Oktober 2024 (TimeID, When) Seorang karyawan bernama Budi (EmployeeID, Who) menjual sepeda (ProdukId, What) sejumlah 2 unit (Quantity, How) kepada pelanggan atau customer bernama Dinda (CustomerID, Who) di daerah Rungkut-Surabaya (TeritoryID, Where) dengan total penjualan sebesar 6.000.000 (Amount, How) ”

b. Tabel factpurchase

Untuk relevansi lakukanlah dengan perumpamaan sebagai berikut :

“ Pada tanggal 16 April 2011 (TimeID, When) seorang karyawan bernama Budi (EmployeeID, Who) membeli komponen sepeda (ProductID, What) sejumlah 1 unit (Quantity, How) di perusahaan atau toko Allenson Cycles (VendorID, Who) di kirim dengan jasa pengiriman ZY - Express (ShipmethodID, How) dengan total pembelian sebesar 5.000.000 (Amount, How) ”

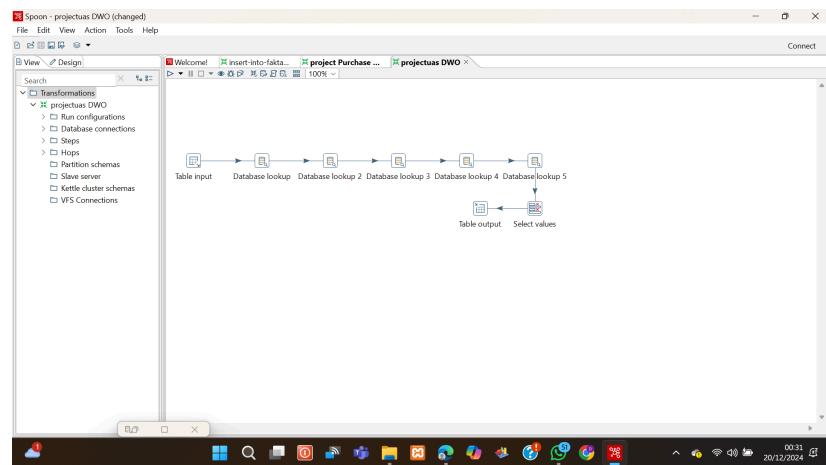
B. Implementasi ETL & Olap

Pada tahap ini dilakukan Data Integration dengan menggunakan tools Pentaho, untuk penjelasan dan langkah - langkah sebagai berikut :

a. Proses ETL & Olap untuk tabel factsales

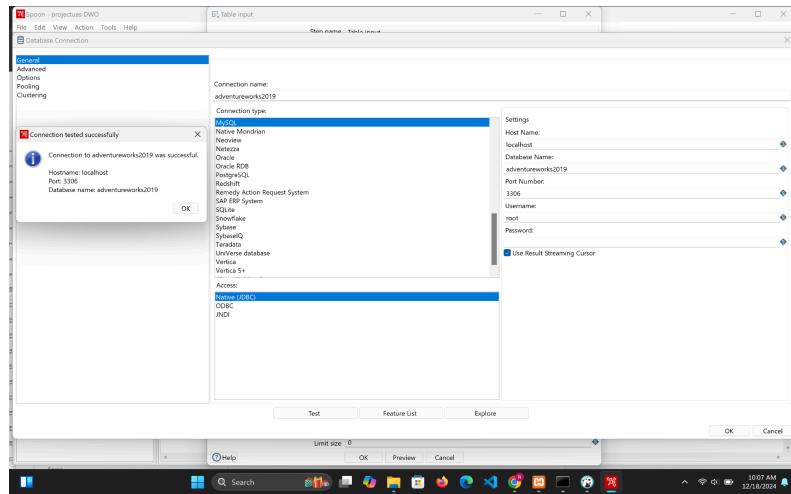
1. Koneksi Database

- Pertama buat file transformation baru dan berikan nama sesuai kebutuhan.
- Setelah itu buat design sesuai kebutuhan, dapat dilihat pada gambar berikut :



- Double click pada tabel input, lalu klik New

- Isi koneksi dan sesuaikan kebutuhan, pada kasus ini table input terkoneksi dengan database adventureworks dapat dilihat pada gambar berikut :



- Pada Database Lookup 1, 2, 3, 4, 5, select values, table input pastuikan terkoneksi dengan database projectus yang telah dibuat sebelumnya, untuk selengkapnya bisa dilihat di tahap proses.

2. Query di Table Input

Pada table input isikan query seperti berikut :

SELECT

```
sop.ProductID,
sc.PersonID AS CustomerID,
sol.BusinessEntityID AS EmployeeID,
soh.TerritoryID,
soh.OrderDate,
sop.LineTotal AS SalesAmount,
sop.OrderQty AS QuantitySold
```

FROM

```
sales_salesorderdetail sop
```

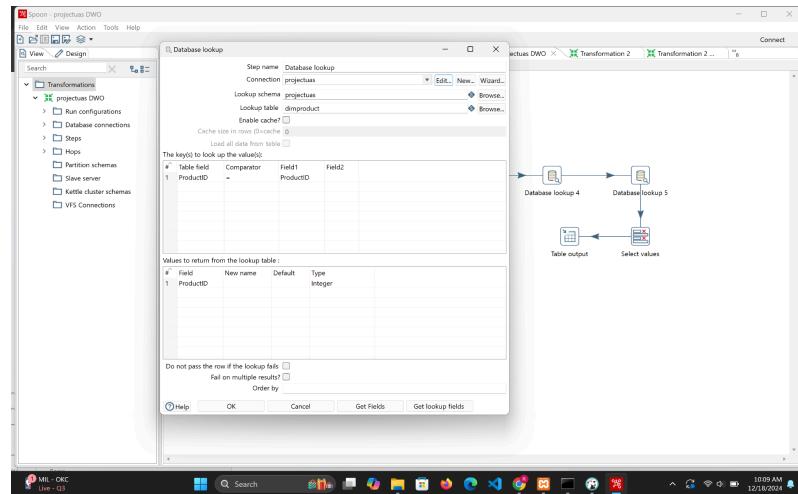
JOIN

```
sales_salesorderheader soh
```

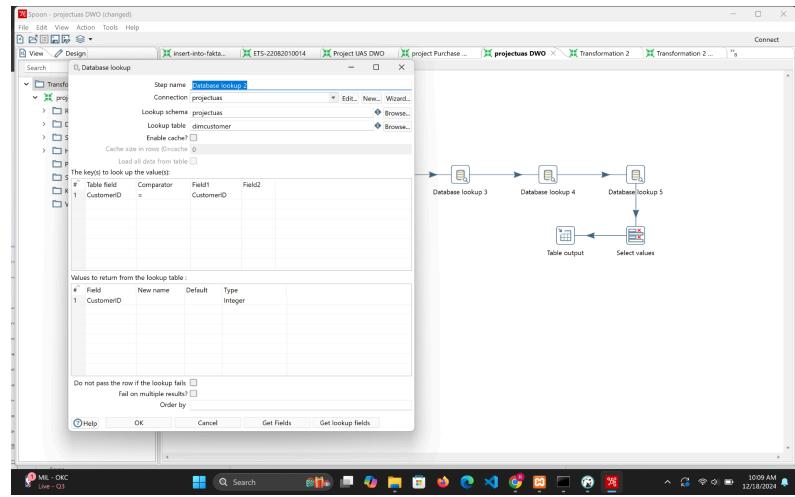
```
    ON sop.SalesOrderID = soh.SalesOrderID  
JOIN  
    sales_customer sc  
    ON soh.CustomerID = sc.CustomerID  
JOIN  
    sales_salesperson sol  
    ON soh.SalesPersonID =  
sol.BusinessEntityID  
JOIN  
    sales_salesterritory st  
    ON soh.TerritoryID = st.TerritoryID  
JOIN  
    production_product sod  
    ON sop.ProductID = sod.ProductID;
```

3. Proses

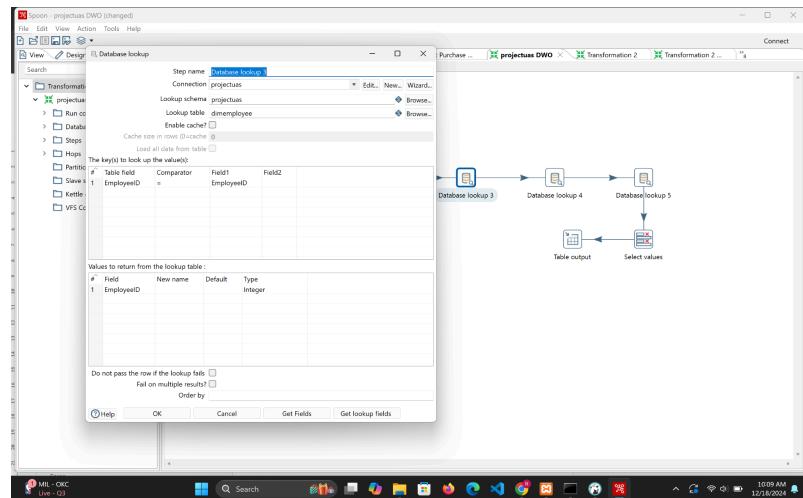
- Pada Database lookup 1 bertujuan untuk melakukan lookup product untuk di lookup berdasarkan productID, dapat dilihat pada gambar berikut :



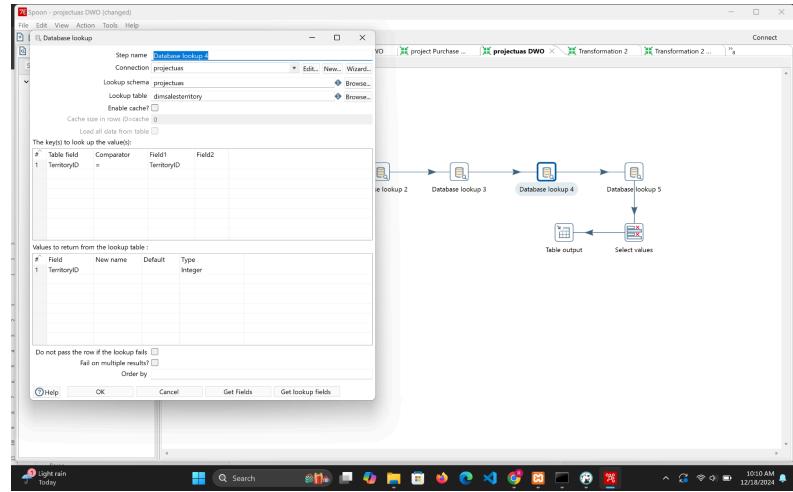
- Pada Database lookup 2 bertujuan untuk melakukan lookup Customer untuk di lookup berdasarkan CustomerID, dapat dilihat pada gambar berikut :



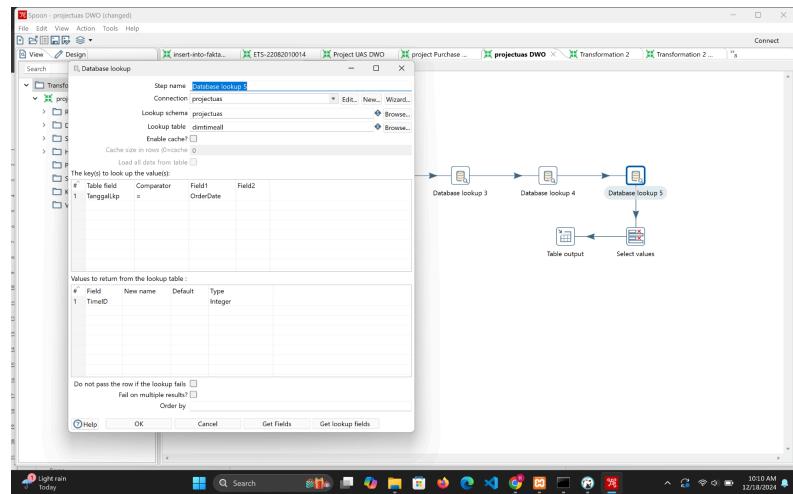
- Pada Database lookup 3 bertujuan untuk melakukan lookup Employee untuk di lookup berdasarkan EmployeeID, dapat dilihat pada gambar berikut :



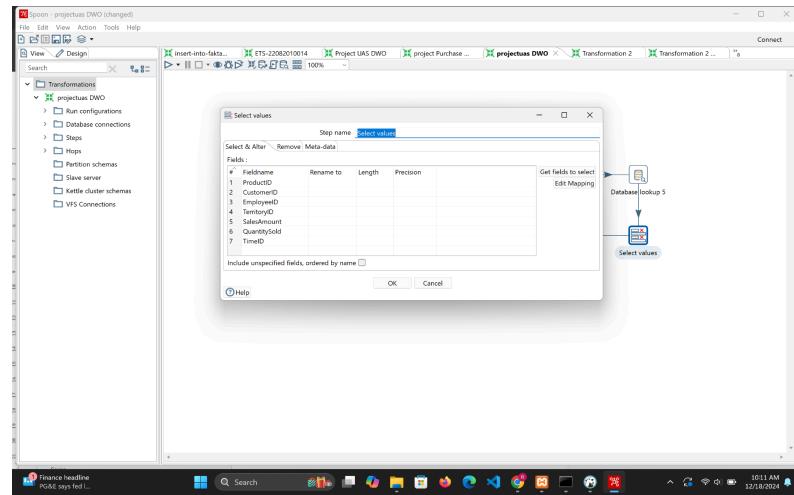
- Pada Database lookup 4 bertujuan untuk melakukan lookup Territory untuk di lookup berdasarkan TerritoryID, dapat dilihat pada gambar berikut :



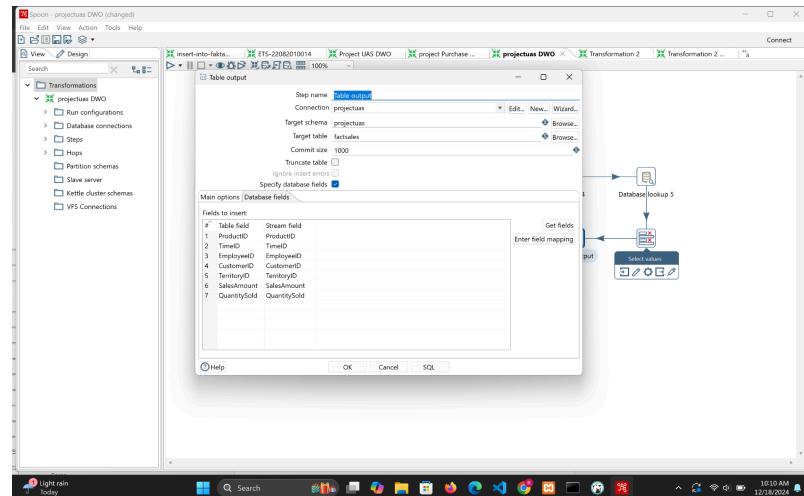
- Pada Database lookup 5 bertujuan untuk melakukan lookup dimtimeall untuk di lookup berdasarkan TimeID, dapat dilihat pada gambar berikut :



- Pada Select Values untuk memasukan data yang dibutuhkan di tabel factsales dapat dilihat pada gambar berikut :



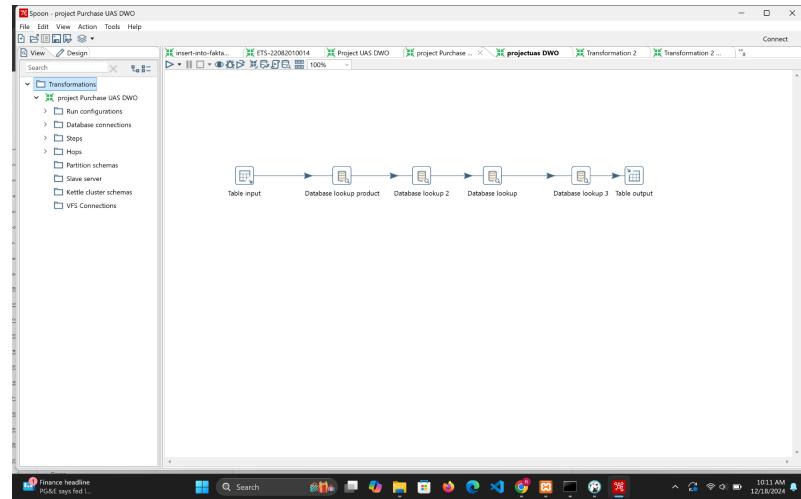
- Pada table output bertujuan untuk memasukan data yang dibutuhkan di tabel factsales, bisa dilihat pada gambar berikut:



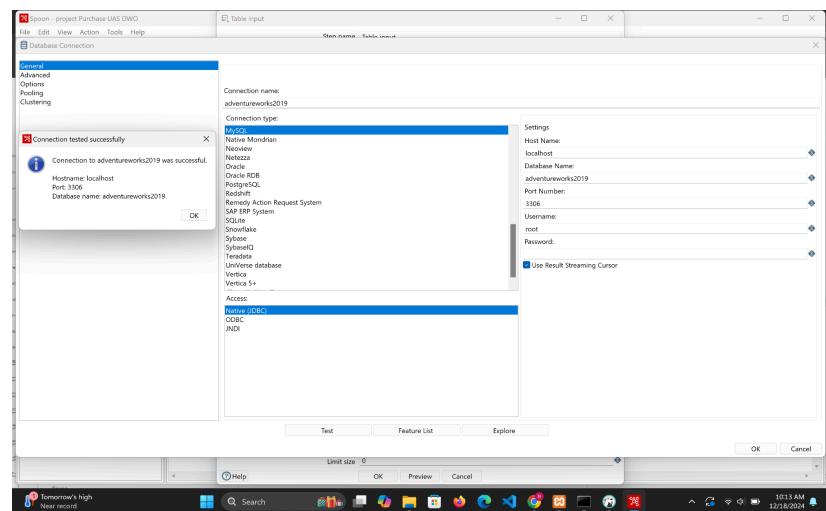
b. Proses ETL & Olap untuk tabel factpurchase

1. Koneksi Database

- Pertama buat file transformation baru dan berikan nama sesuai kebutuhan.
- Setelah itu buat design sesuai kebutuhan, dapat dilihat pada gambar berikut :



- Double click pada tabel input, lalu klik New
- Isi koneksi dan sesuaikan kebutuhan, pada kasus ini table input terkoneksi dengan database adventureworks dapat dilihat pada gambar berikut :



- Pada Database Lookup 1, 2, 3, 4 dan table input terkoneksi dengan database projectus yang telah dibuat sebelumnya, untuk selengkapnya bisa dilihat di tahap proses.

2. Query di Table Input

Pada table input isikan query seperti berikut :

```

SELECT

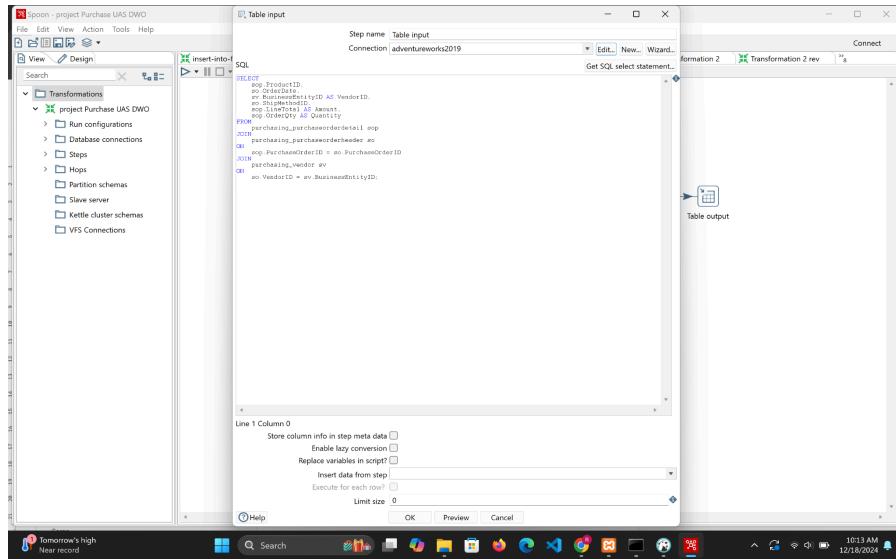
    sop.ProductID,
    so.OrderDate,
    sv.BusinessEntityID AS VendorID,
    so.ShipMethodID,
    sop.LineTotal AS Amount,
    sop.OrderQty AS Quantity

FROM

    purchasing_purchaseorderdetail sop
JOIN
    purchasing_purchaseorderheader so
ON
    sop.PurchaseOrderID = so.PurchaseOrderID
JOIN
    purchasing_vendor sv
ON

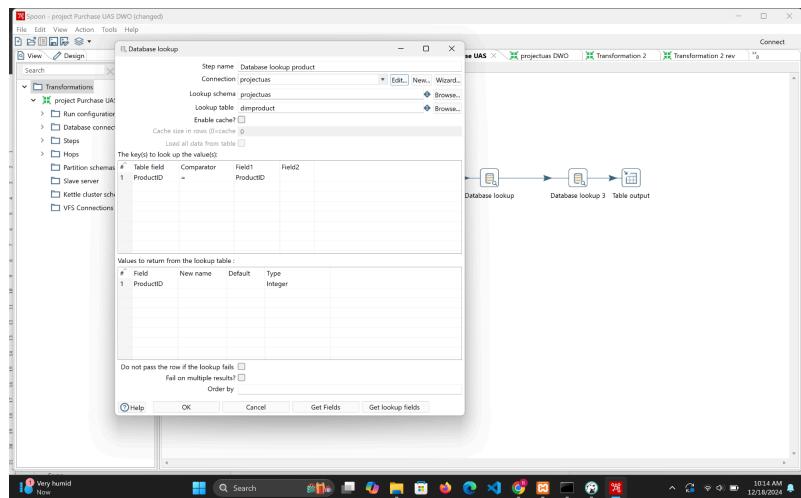
```

`so.VendorID = sv.BusinessEntityID;`

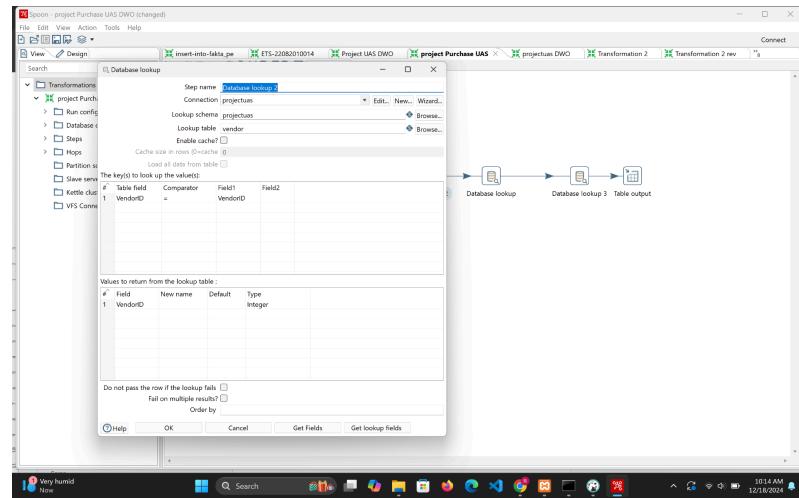


3. Proses

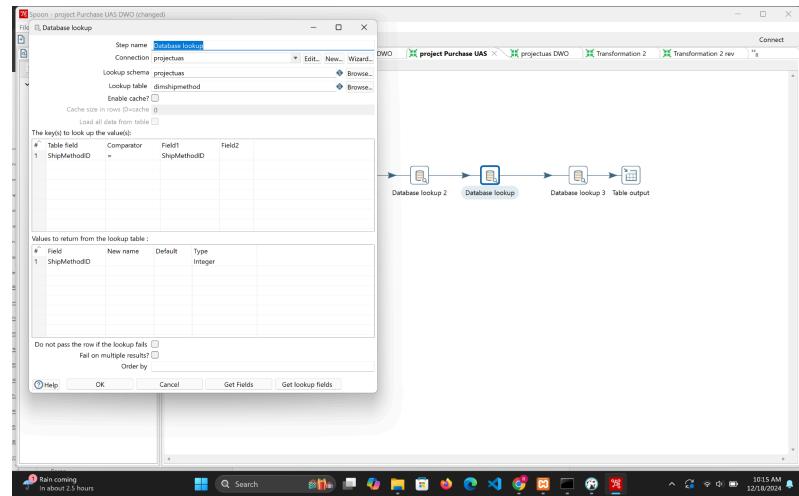
- Pada Database lookup 1 bertujuan untuk melakukan lookup product untuk di lookup berdasarkan productID, dapat dilihat seperti gambar berikut:



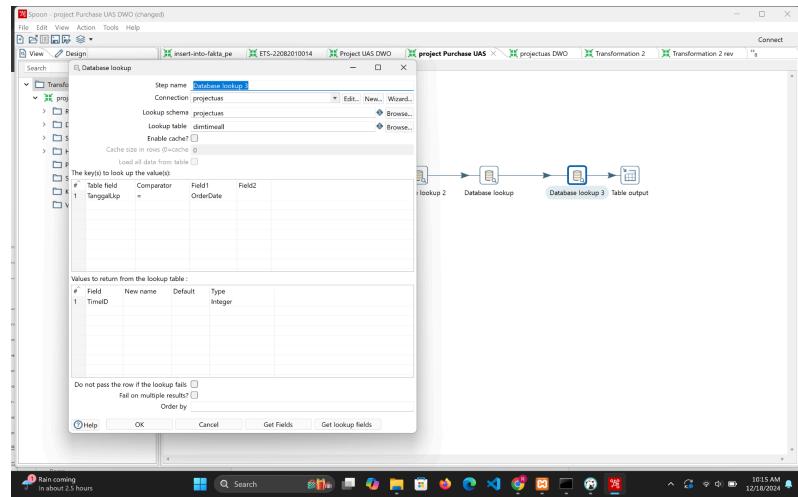
- Pada Database lookup 2 bertujuan untuk melakukan lookup Vendor untuk di lookup berdasarkan VendorID, dapat dilihat seperti gambar berikut :



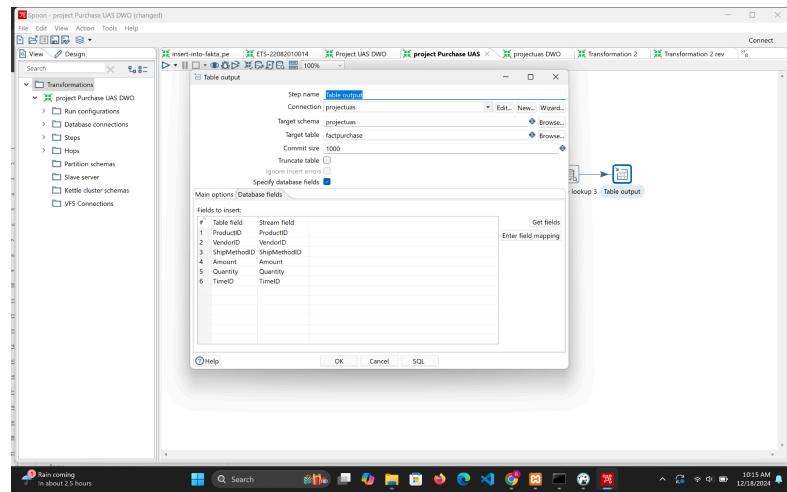
- Pada Database lookup 3 bertujuan untuk melakukan lookup ShipMethod untuk di lookup berdasarkan ShipMethodID, dapat dilihat seperti gambar berikut:



- Pada Database lookup 4 bertujuan untuk melakukan lookup dimtimeall untuk di lookup berdasarkan TimeID, dapat dilihat seperti gambar berikut:

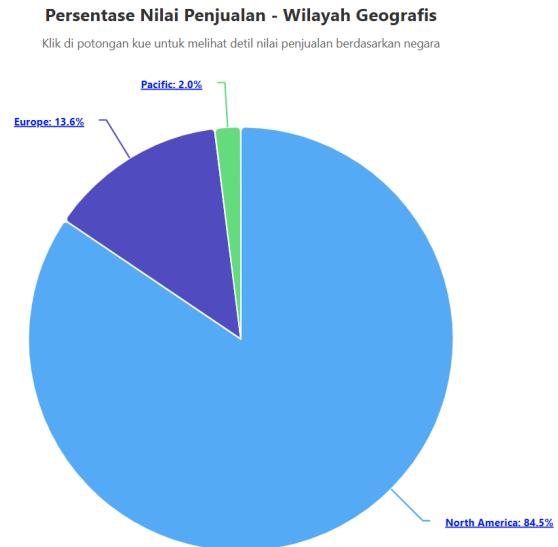


- Pada table output bertujuan untuk memasukan data yang dibutuhkan di tabel factpurchase, bisa dilihat pada gambar berikut:



C. Implementasi Dashboard & Hasil Insight Analisis

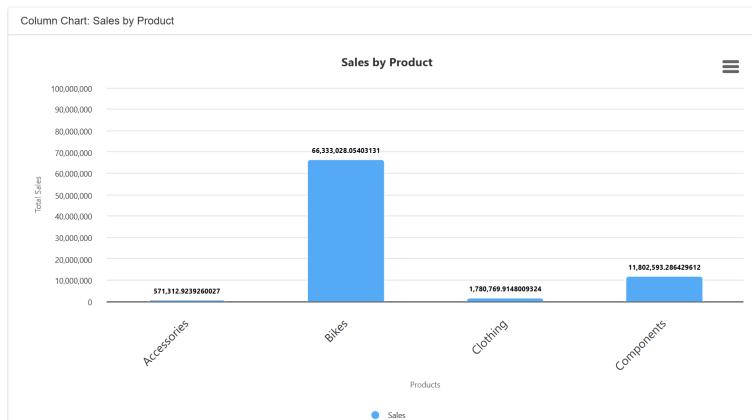
- Analisis total penjualan yang dilakukan customer di setiap wilayah



Dari hasil analisis yang dilakukan dan hasil grafik diatas menunjukkan bahwa distribusi nilai penjualan sangat didominasi oleh wilayah Amerika Utara dengan kontribusi sebesar 84,5% dari total penjualan. Wilayah ini menunjukkan bahwa pasar Amerika Utara menjadi urutan pertama dalam menghasilkan pendapatan. Wilayah Eropa berada di posisi kedua dengan kontribusi sebesar 13,6%, yang mencerminkan adanya potensi pasar yang cukup signifikan, meskipun jauh lebih kecil dibandingkan dengan Amerika Utara. Sementara itu, wilayah Pasifik memiliki kontribusi yang paling kecil, yaitu hanya sebesar 2% dari total penjualan, yang menunjukkan bahwa pasar di wilayah ini masih memiliki ruang yang sangat besar untuk ditingkatkan atau belum menjadi prioritas utama dalam strategi penjualan. Secara keseluruhan, grafik ini mengindikasikan ketergantungan yang besar pada pasar Amerika Utara. Oleh karena itu, jika perusahaan ingin mengurangi risiko ketergantungan terhadap satu wilayah, perlu dilakukan diversifikasi pasar dengan lebih mengembangkan wilayah Eropa dan Pasifik. Hal ini juga bisa menjadi

peluang untuk eksplorasi lebih lanjut guna memperluas basis pelanggan dan meningkatkan pangsa pasar di wilayah-wilayah tersebut.

b. Analisis total penjualan sales berdasarkan produk

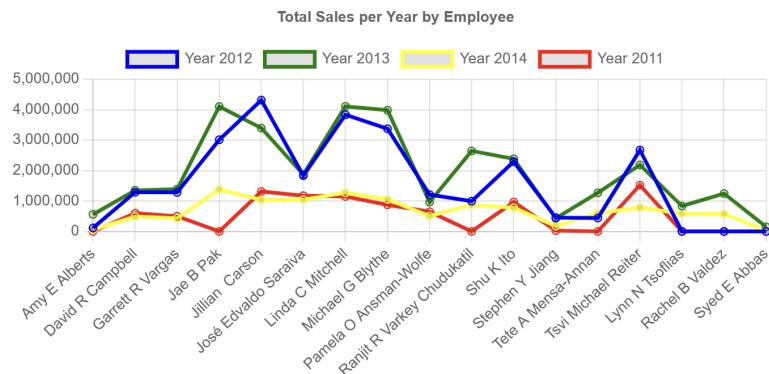


Berdasarkan grafik "Sales by Product," penjualan tertinggi berasal dari kategori Bikes, dengan total sebesar \$64,35 juta, menjadikannya penyumbang utama pendapatan perusahaan. Kategori Components menempati posisi kedua dengan penjualan sebesar \$14,88 juta, diikuti oleh Clothing sebesar \$11,74 juta, dan Accessories dengan angka terendah, yaitu \$9,32 juta. Dominasi Bikes yang sangat signifikan menunjukkan bahwa produk ini memiliki permintaan pasar yang tinggi dan menjadi kekuatan utama bisnis perusahaan.

Dari data ini, dapat diperoleh insight bahwa perusahaan perlu memprioritaskan strategi pengembangan dan inovasi untuk mempertahankan dominasi Bikes, misalnya melalui peningkatan kualitas, peluncuran varian baru, atau penguatan distribusi. Sementara itu, kategori dengan penjualan lebih rendah seperti Accessories, Clothing, dan Components memiliki potensi peningkatan melalui riset pasar untuk memahami kebutuhan konsumen, promosi yang lebih menarik, serta diversifikasi produk. Dengan memaksimalkan potensi dari semua kategori,

perusahaan dapat meningkatkan pendapatan secara keseluruhan dan memperluas pangsa pasarnya.

c. Analisis Trend Penjualan sales berdasarkan Employee di setiap tahun

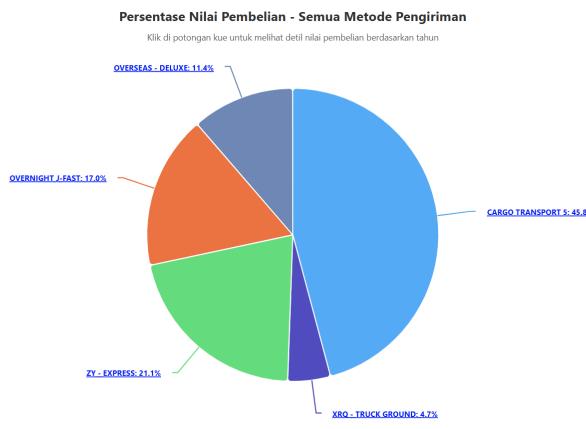


Berdasarkan grafik "Total Sales per Year by Employee," terlihat perbedaan signifikan dalam kinerja penjualan karyawan dari tahun 2011 hingga 2014. Pada tahun 2012 dan 2013, beberapa karyawan seperti Jillian Carson, Linda C. Mitchell, dan Pamela O. Ansman-Wolfe menunjukkan performa penjualan yang sangat tinggi, dengan angka mendekati atau melebihi \$4 juta. Sementara itu, pada tahun 2011 dan 2014, penjualan sebagian besar karyawan cenderung lebih rendah, di bawah \$2 juta. Hal ini menunjukkan adanya fluktuasi kinerja karyawan yang cukup signifikan setiap tahunnya.

Insight yang dapat diambil dari data ini adalah pentingnya mengevaluasi faktor-faktor yang memengaruhi peningkatan performa di tahun-tahun tertentu, seperti strategi penjualan, promosi, atau pelatihan yang diterapkan pada tahun-tahun tersebut. Fokus perusahaan sebaiknya diarahkan pada replikasi strategi yang berhasil pada karyawan dengan penjualan tertinggi untuk mendorong performa karyawan lainnya. Selain itu, karyawan dengan kinerja rendah dapat diberikan pelatihan tambahan atau strategi yang lebih terarah untuk meningkatkan kontribusinya. Dengan

cara ini, perusahaan dapat menjaga stabilitas dan meningkatkan total penjualan di masa depan.

d. Analisis Pembelian Purchase di semua kategori pengiriman

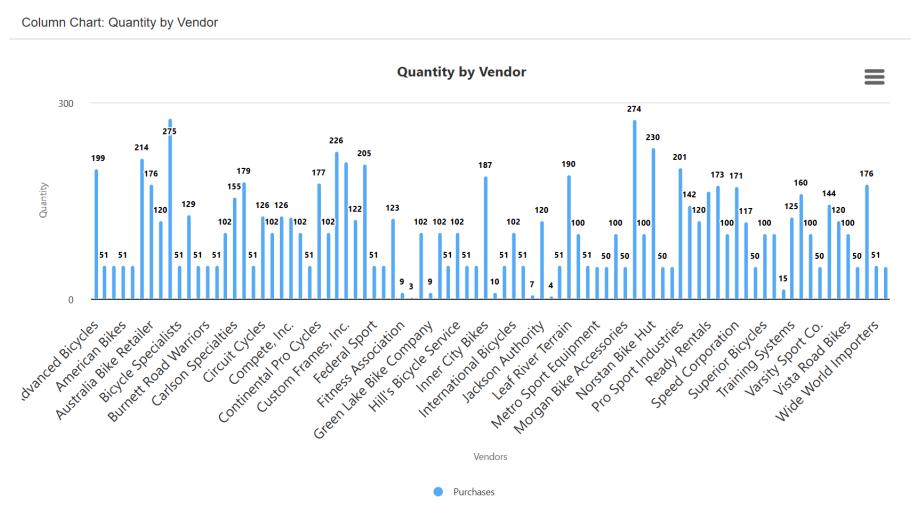


Berdasarkan diagram "Percentase Nilai Pembelian - Semua Metode Pengiriman," metode pengiriman Cargo Transport mendominasi dengan kontribusi terbesar sebesar 45,8% dari total nilai pembelian. Metode ini diikuti oleh ZY - Express dengan 21,3% dan Overnight - Fast sebesar 17,0%. Sementara itu, Overseas - Deluxe menyumbang 11,4%, dan metode pengiriman dengan kontribusi terkecil adalah XRO - Truck Ground, hanya sebesar 4,5%. Data ini menunjukkan bahwa Cargo Transport merupakan metode pengiriman yang paling banyak digunakan atau bernilai tinggi, mungkin karena kapasitas besar atau efisiensi biaya yang ditawarkan.

Insight yang dapat diperoleh adalah bahwa Cargo Transport sangat penting untuk dipertahankan dan diperkuat, mengingat kontribusinya yang signifikan terhadap nilai pembelian. Perusahaan juga dapat mengevaluasi strategi untuk meningkatkan penggunaan metode pengiriman lain yang lebih kecil kontribusinya, seperti XRO - Truck Ground, dengan mengidentifikasi kendala yang ada, seperti biaya, waktu pengiriman, atau jangkauan layanan. Selain itu, metode dengan kontribusi menengah seperti ZY - Express dan Overnight - Fast memiliki potensi untuk ditingkatkan

melalui promosi layanan yang lebih luas atau penawaran khusus. Dengan strategi yang tepat, diversifikasi metode pengiriman dapat memberikan nilai tambah bagi pelanggan dan meningkatkan efisiensi logistik secara keseluruhan.

e. Analisis Jumlah Quantity berdasarkan Vendor

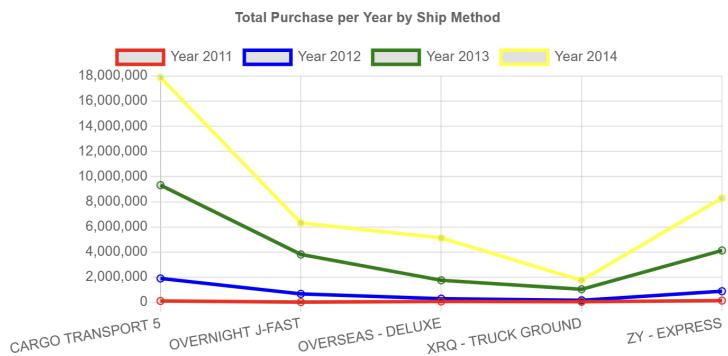


Dari hasil analisis, terlihat bahwa terdapat perbedaan signifikan dalam jumlah kuantitas pembelian berdasarkan vendor. Vendor seperti "Metro Sport Terrain" dan "Pro Sports Industries" mencatat kuantitas tertinggi, masing-masing dengan 274 dan 230 unit, menunjukkan dominasi mereka dalam pasar. Hal ini dapat disebabkan oleh faktor seperti kualitas produk yang lebih baik, reputasi yang kuat, atau strategi pemasaran yang efektif. Sebaliknya, vendor seperti "Advanced Bicycles" dan "Wide World Importers" memiliki kuantitas paling rendah dengan 51 unit, mengindikasikan pangsa pasar yang kecil atau kurangnya daya saing dibandingkan vendor lain.

Insight ini menunjukkan bahwa distribusi pasar tidak merata, dengan sebagian besar kuantitas dikuasai oleh beberapa vendor tertentu. Vendor dengan kuantitas rendah dapat menggunakan data ini untuk

mengidentifikasi kelemahan mereka, seperti memperbaiki kualitas produk, menyesuaikan harga, atau meningkatkan upaya promosi. Sementara itu, vendor dengan kuantitas tinggi perlu mempertahankan keunggulan mereka dengan terus memberikan nilai tambah bagi pelanggan untuk menjaga posisi mereka di pasar. Analisis ini memberikan gambaran penting untuk pengambilan keputusan strategis bagi setiap vendor.

- f. Analisis Trend Pembelian Purchase berdasarkan kategori pengiriman di setiap tahun



Dari hasil analisis, terlihat bahwa terdapat perbedaan signifikan dalam jumlah kuantitas pembelian berdasarkan vendor. Vendor seperti "Metro Sport Terrain" dan "Pro Sports Industries" mencatat kuantitas tertinggi, masing-masing dengan 274 dan 230 unit, menunjukkan dominasi mereka dalam pasar. Hal ini dapat disebabkan oleh faktor seperti kualitas produk yang lebih baik, reputasi yang kuat, atau strategi pemasaran yang efektif. Sebaliknya, vendor seperti "Advanced Bicycles" dan "Wide World Importers" memiliki kuantitas paling rendah dengan 51 unit, mengindikasikan pangsa pasar yang kecil atau kurangnya daya saing dibandingkan vendor lain.

Insight ini menunjukkan bahwa distribusi pasar tidak merata, dengan sebagian besar kuantitas dikuasai oleh beberapa vendor tertentu. Vendor dengan kuantitas rendah dapat menggunakan data ini untuk mengidentifikasi kelemahan mereka, seperti memperbaiki kualitas produk, menyesuaikan harga, atau meningkatkan upaya promosi. Sementara itu, vendor dengan kuantitas tinggi perlu mempertahankan keunggulan mereka dengan terus memberikan nilai tambah bagi pelanggan untuk menjaga posisi mereka di pasar. Analisis ini memberikan gambaran penting untuk pengambilan keputusan strategis bagi setiap vendor.

BAB IV

KESIMPULAN

A. Kesimpulan

Proyek pengembangan data warehouse dan dashboard berbasis database Adventure Works 2019 berhasil menghasilkan sistem yang mampu mengintegrasikan data secara efektif untuk kebutuhan analisis penjualan dan pembelian. Proses ETL (Extract, Transform, Load) yang diterapkan memastikan data yang digunakan relevan, terstruktur, dan mudah dianalisis. Dashboard yang dibuat mempermudah manajemen dalam memahami pola penjualan, pembelian, serta efisiensi logistik. Analisis menunjukkan bahwa wilayah Amerika Utara mendominasi penjualan, kategori produk "Bikes" menjadi sumber utama pendapatan, dan metode pengiriman Cargo Transport memberikan kontribusi terbesar. Sistem ini membuktikan bahwa penggunaan teknologi data warehouse dan dashboard mampu meningkatkan kualitas pengambilan keputusan yang berbasis data.

B. Saran

Untuk meningkatkan efektivitas dan efisiensi, perusahaan disarankan untuk memperluas fokus ke wilayah dengan kontribusi rendah seperti Pasifik dan Eropa guna mengurangi ketergantungan pada wilayah utama. Inovasi dan promosi produk pada kategori yang kurang dominan, seperti Accessories dan Clothing, dapat membantu meningkatkan diversifikasi produk dan pendapatan. Selain itu, evaluasi metode pengiriman yang kurang optimal perlu dilakukan untuk meningkatkan efisiensi logistik. Perusahaan juga dapat mempertimbangkan pelatihan tambahan bagi karyawan untuk meningkatkan kinerja penjualan serta melakukan riset pasar guna memahami kebutuhan pelanggan yang lebih spesifik. Dengan strategi ini, perusahaan diharapkan dapat memperkuat daya saing dan mencapai pertumbuhan bisnis yang lebih baik.

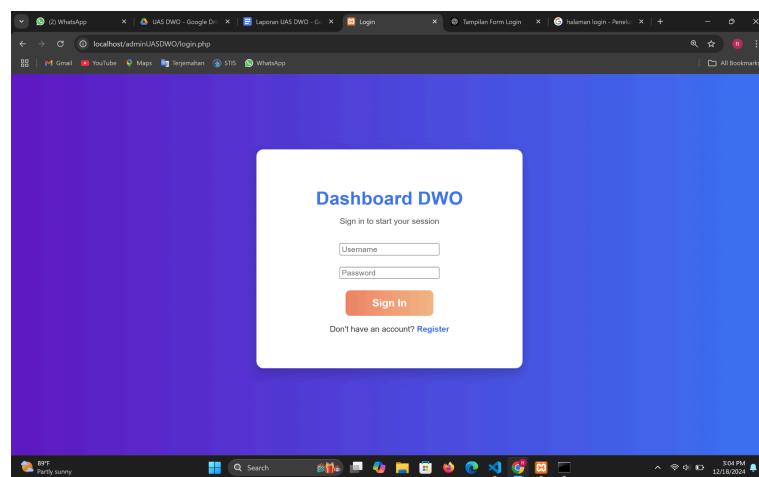
LAMPIRAN

A. Link GitHub

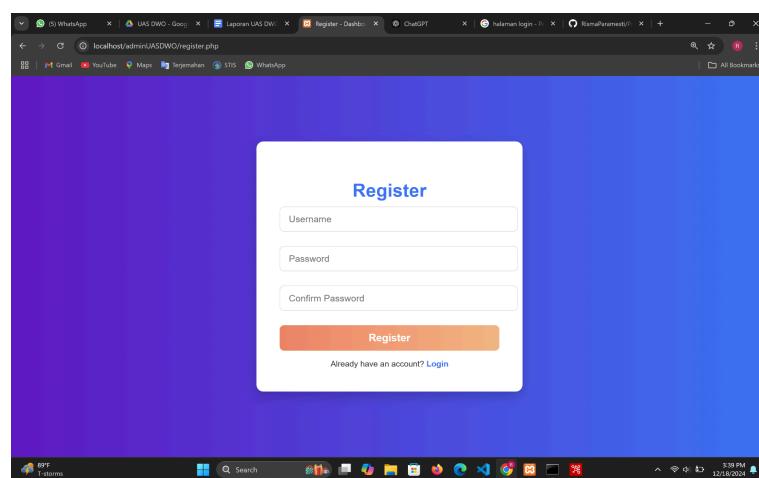
[Link GitHub Program UAS DWO](#)

B. Lampiran Hasil Tampilan Dashboard

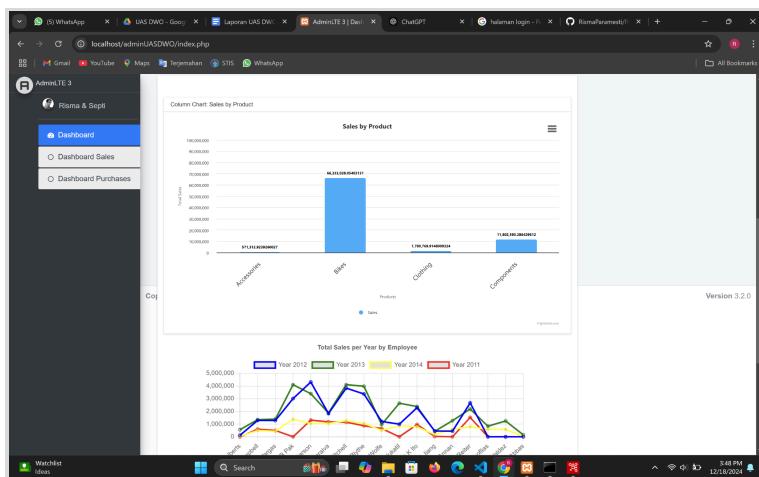
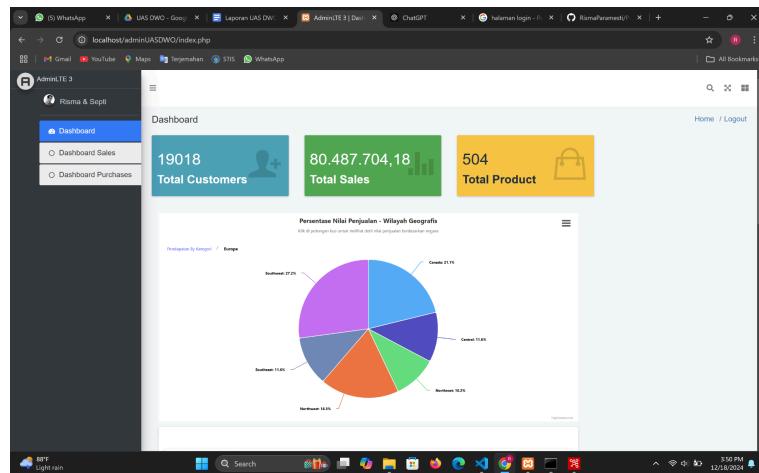
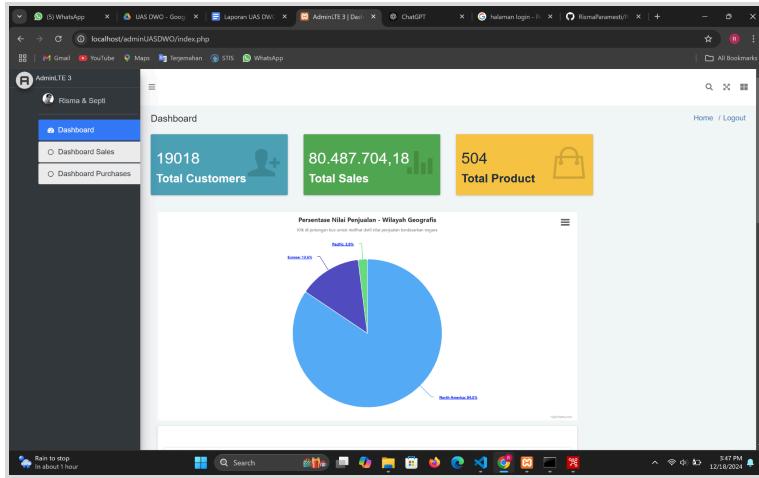
a. Login

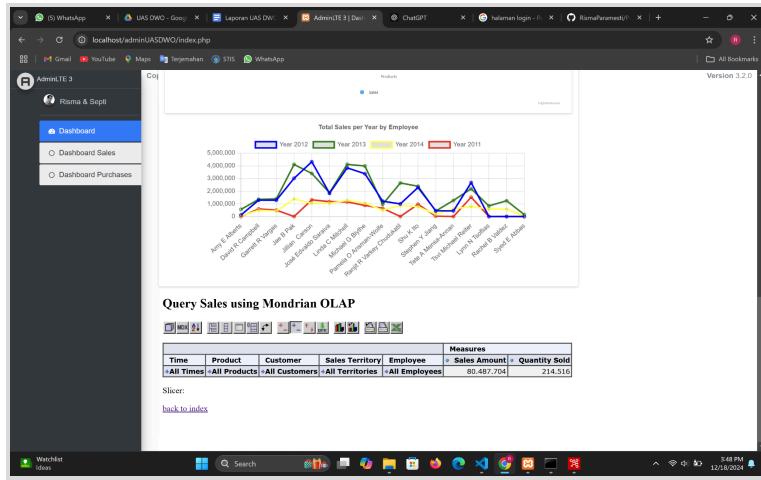


b. Register



c. Kategori Sales





d. Kategori Purchase

