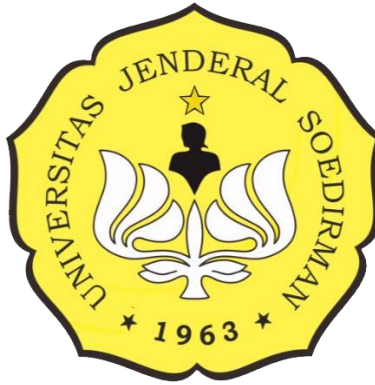


LAPORAN PROJECT BASIS DATA

JASA ANTAR MAKANAN ONLINE



DISUSUN OLEH:

- | | |
|---------------------------|-------------|
| 1. Naufal Althafi Handoyo | (H1D021087) |
| 2. Maulana Zhahran | (H1D021070) |
| 3. Taufik Satria Nugraha | (H1D020028) |

JURUSAN INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS JENDERAL SOEDIRMAN

TAHUN AJARAN 2021/2022

BAB I

PENDAHULUAN

A. Latar Belakang

Basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh suatu informasi dari database tersebut. Perangkat lunak yang digunakan untuk mengolah dan mengambil query basis data disebut sistem manajemen basis data. Pemrosesan basis data sebagai perangkat andalan sangat diperlukan oleh berbagai institusi dan perusahaan. Dalam pengembangan sistem informasi diperlukan basis data sebagai media penyimpanan data. Kehadiran basis data dapat meningkatkan daya saing perusahaan tersebut. Basis data dapat mempercepat upaya pelayanan kepada pelanggan, menghasilkan informasi dengan cepat dan tepat sehingga membantu pengambilan keputusan untuk segera memutuskan suatu masalah berdasarkan informasi yang ada. Banyak aplikasi yang dibuat dengan berlandaskan pada basis data antara lain semua transaksi perbankan, aplikasi pemesanan, penjadwalan penerbangan, proses registrasi dan pencatatan data mahasiswa pada perguruan tinggi, aplikasi pemrosesan penjualan, pembelian dan pencatatan data barang pada perusahaan dagang, pencatatan data pegawai beserta aktivitasnya termasuk operasi penggajian pada suatu perusahaan, dan sebagainya. Beberapa informasi pada perusahaan retail seperti jumlah penjualan, mencari jumlah stok yang tersedia, barang apa yang paling laku dijual pada bulan ini, dan berapa laba bersih perusahaan dapat diketahui dengan mudah menggunakan basis data.

Di zaman yang serba digital ini banyak sekali kegiatan yang dapat dilakukan dengan mudah hanya dengan menggunakan sistem informasi yang terhubung pada jaringan internet dan masyarakat cenderung menginginkan hal-hal yang tidak merepotkan. Misalnya berbelanja online, membeli makanan online, melakukan kegiatan pembelajaran daring, melakukan pekerjaan daring. Tentunya setiap sistem informasi memiliki sistem basis datanya sendiri sendiri, oleh karena itu kami mencoba membuat sistem basis data berdasarkan sistem informasi apa yang banyak dibutuhkan masyarakat salah satunya yaitu membeli makanan secara online, maka dibutuhkan juga jasa antar makanan oleh sebab itu kami akan membuat sistem basis data untuk jasa antar makanan.

B. Rumusan Masalah

Pada Laporan Tugas Project ini akan dirumuskan masalah mengenai “Bagaimana Proses atau Tahapan Pembuatan Sistem Basis Data Jasa Antar Makanan”

C. Tujuan

1. Memahami Bagaimana Proses Perancangan Sistem Basis Data
2. Memahami Bagaimana Proses Pembuatan Sistem Basis Data
3. Memahami Bagaimana Proses Penerapan Sistem Basis Data
4. Sebagai Tugas Project Mata Kuliah Basis Data II

BAB II

PEMBAHASAN

A. Role Utama Basis Data

Role merupakan pengaturan hak akses user pada database Oracle dengan mengelompokkan beberapa privileges. Terdapat predefined roles dan user defined roles. Predefined roles merupakan role yang sudah disiapkan oleh DBMS Oracle, sedangkan user defined roles merupakan role yang ditentukan sendiri oleh programmer. User defined role dalam proyek basis data ini ada 3 yaitu :

1. Admin : Mengatur dan merawat basis data.
2. Driver : Membeli makanan yang sebelumnya sudah dipesan oleh pembeli sekaligus diantarkan ke alamat pembeli.
3. Pembeli : Memesan makanan yang diinginkan untuk kemudian diantarkan oleh driver.

B. Entitas Basis Data

Entitas atau entity dalam database adalah benda, orang, tempat, unit, objek atau hal lainnya yang mempresentasikan data dan data tersebut akan disimpan ke dalam pangkal data. Pengertian lain dari Entitas adalah objek yang mewakili sesuatu dalam dunia nyata dan dapat dibedakan antara satu dengan lainnya (unique). Setiap entitas memiliki beberapa atribut yang mendeskripsikan karakteristik dari objek. Entitas dalam proyek basis data ini adalah sebagai berikut :

1. Pembeli(id_pembeli, nama_pembeli, alamat_pembeli, no_tel_pembeli, email)
2. Pesanan(id_pesanan, tanggal_pemesanan, total_harga)
3. Driver(id_driver, nama_driver, no_tel_driver, rating_driver)
4. Makanan(id_makanan, nama_makanan, harga_makanan, rating_makanan)
5. Restoran(id_restoran, nama_restoran, alamat_restoran, no_tel_restoran)

C. Aturan Bisnis

Menurut Business Rules Group (1993), aturan bisnis adalah pernyataan yang mendefinisikan atau membatasi beberapa aspek bisnis. Adanya aturan-aturan bisnis ini dimaksudkan untuk menegaskan struktur bisnis atau untuk mengendalikan atau mempengaruhi perilaku bisnis. Aturan bisnis biasanya dituliskan dalam dokumentasi sederhana terkait sistem yang dibangun dan menghubungkannya dengan spesifikasi kebutuhan fungsional. Aturan bisnis dalam basis data ini adalah sebagai berikut :

1. Satu pembeli bisa melakukan banyak pesanan.
Satu pesanan hanya bisa dilakukan oleh satu pembeli.
2. Satu driver bisa mengantar banyak pesanan.
Satu pesanan hanya bisa diantar oleh satu driver.

3. Satu pesanan dapat memiliki banyak makanan.
Satu makanan dapat dimiliki oleh banyak pesanan.
4. Satu restoran menyediakan banyak makanan.
Satu makanan hanya bisa disediakan oleh satu restoran.
5. Satu pembeli menilai banyak driver.
Satu driver dapat dinilai oleh banyak pembeli.
6. Satu pembeli menilai banyak makanan.
Satu makanan dapat dinilai oleh banyak pembeli.

D. Relasi Basis Data

Secara umum, relasi adalah hubungan antara tabel yang mempresentasikan hubungan antar objek di dunia nyata. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur mengatur operasi suatu database.

Jadi relasi merupakan penghubung antar tabel satu dengan tabel lainnya yang mana tabel tersebut memiliki data yang berhubungan di dunia nyata untuk mengatur operasi suatu database. Pada sebuah database, relasi dihubungkan dengan cara memberikan satu kolom dengan value yang sama dengan tabel yang berhubungan, ini disebut foreign key. Foreign key sendiri merupakan sebuah kolom yang hanya bisa menyimpan data yang sama dengan primary key yang berhubungan dengan tabel tersebut. Artinya foreign key hanya bisa diisi dengan data yang sudah ada pada primary key. Relasi yang ada di basis data ini adalah sebagai berikut :

1. Driver mengantar Pesanan



Tabel utama : Driver, Pesanan

Tabel kedua : Pesanan

Relasi : One to Many (1-N)

Atribut penghubung : id_driver (Foregin Key id_driver di Pesanan)

2. Pembeli melakukan Pesanan



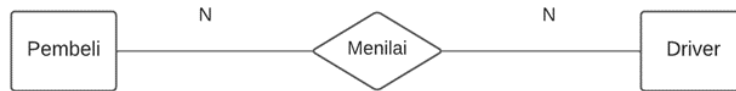
Tabel utama : Pembeli, Pesanan

Tabel kedua : Pesanan

Relasi : One to Many (1-N)

Atribut penghubung : id_pembeli (Foregin Key id_pembeli di Pesanan)

3. Pembeli menilai driver



Tabel utama : Pembeli, Driver

Tabel kedua : rating_driver

Relasi : Many to Many (N-N)

Atribut penghubung : id_driver, id_pembeli (Foreign Key id_driver, id_pembeli di rating_driver)

4. Pembeli menilai makanan



Tabel utama : Pembeli, Makanan

Tabel kedua : rating_makanan

Relasi : Many to Many (N-N)

Atribut penghubung : id_makanan, id_pembeli (Foreign Key id_makanan, id_pembeli di rating_makanan)

5. Pesanan memiliki makanan



Tabel utama : Pesanan, makanan

Tabel kedua : detail_pesanan

Relasi : Many to Many (N-N)

Atribut penghubung : id_pesanan, id_makanan (Foreign Key id_pesanan, id_makanan di detail_pesanan)

6. Restoran sedia makanan



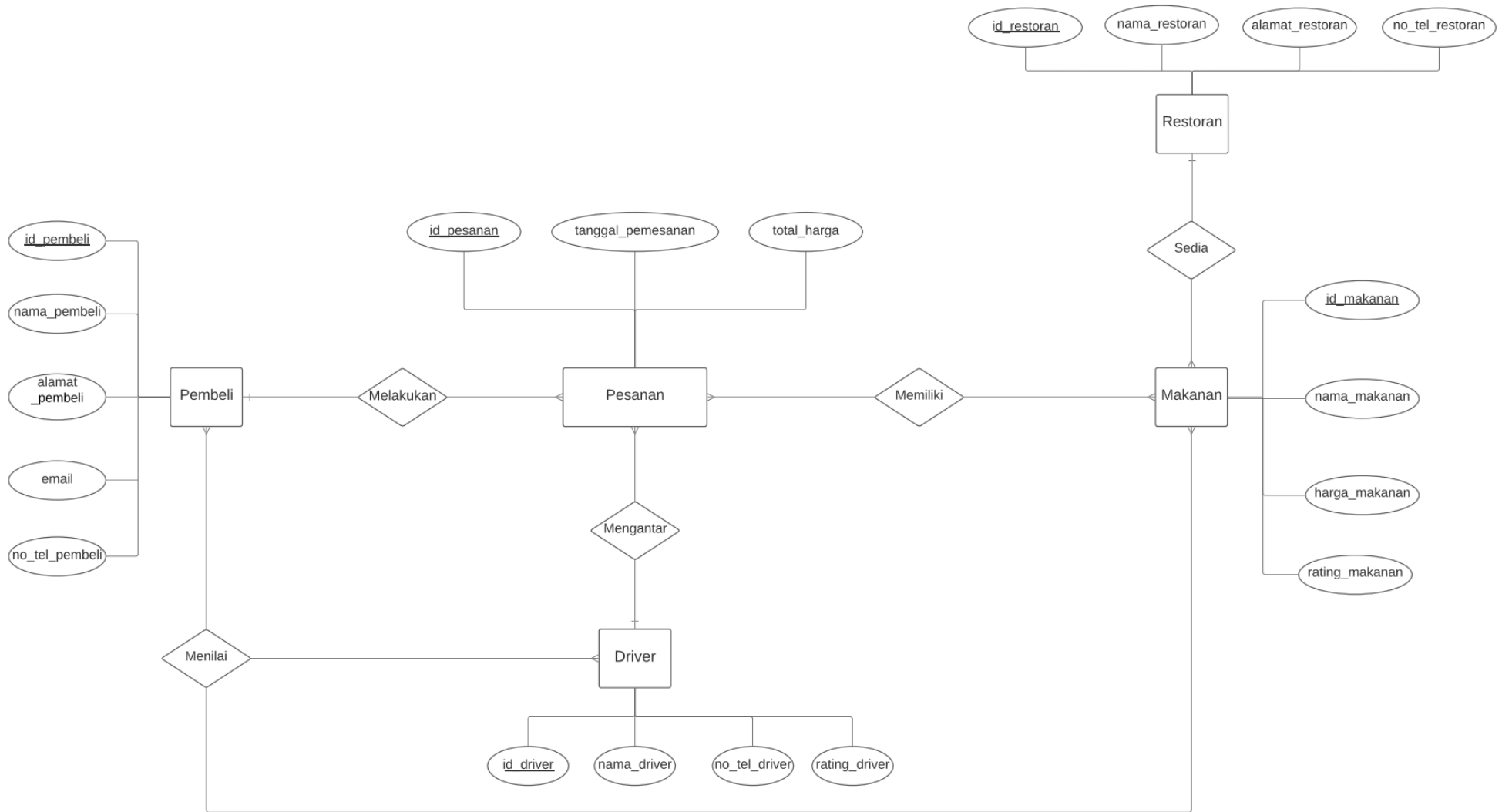
Tabel utama : Restoran, Makanan

Tabel kedua : Makanan

Relasi : One to Many (1-N)

Atribut penghubung : id_restoran (Foreign Key id_restoran di Makanan)

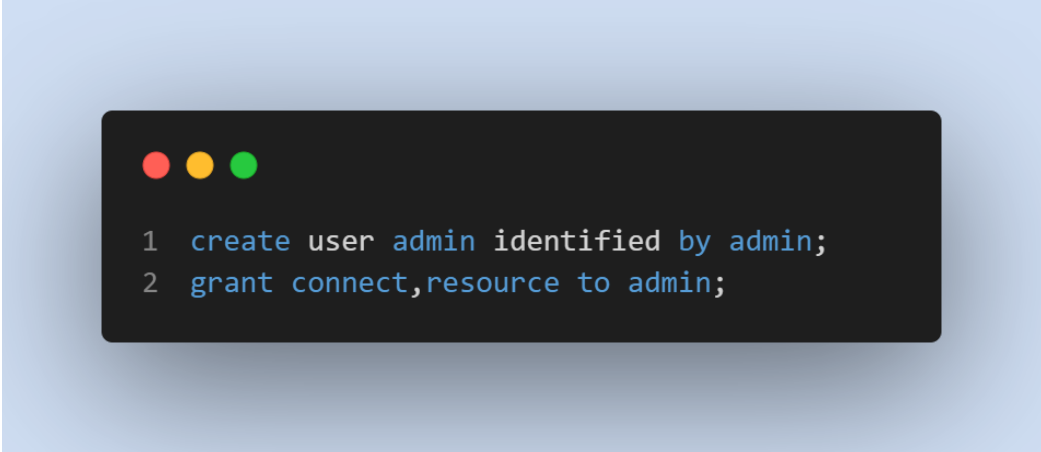
E. Rancangan ERD



F. Langkah-langkah Pembuatan Sistem Basis Data Jasa Antar Makanan

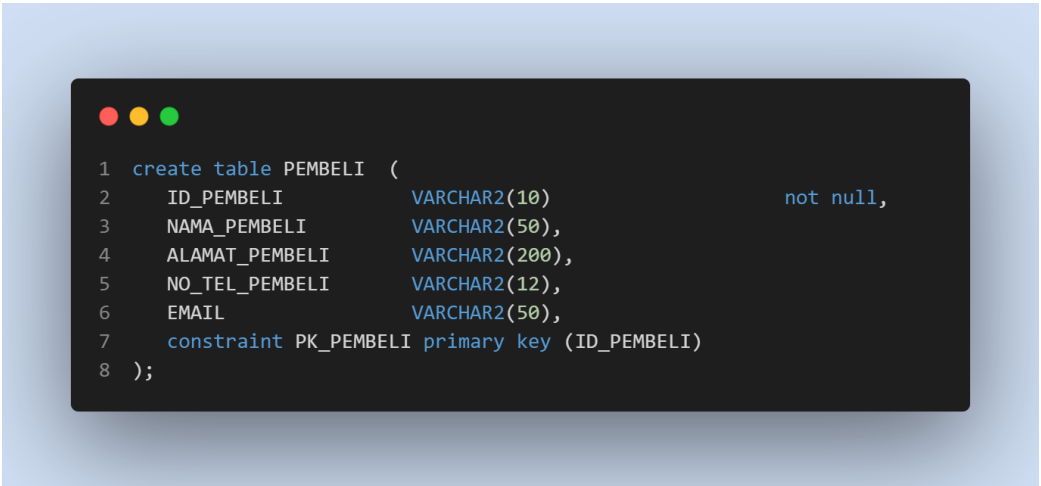
Pada pembuatan basis data ini, kami menggunakan dbms oracle version 10g express edition. Berikut ini langkah-langkah yang dilakukan dalam pembuatan basis data.

1. Membuat dan memberikan privileges ke user admin



```
1 create user admin identified by admin;
2 grant connect,resource to admin;
```

2. Membuat tabel pembeli



```
1 create table PEMBELI (
2     ID_PEMBELI          VARCHAR2(10)                not null,
3     NAMA_PEMBELI        VARCHAR2(50),
4     ALAMAT_PEMBELI      VARCHAR2(200),
5     NO_TEL_PEMBELI      VARCHAR2(12),
6     EMAIL                VARCHAR2(50),
7     constraint PK_PEMBELI primary key (ID_PEMBELI)
8 );
```

3. Membuat tabel driver

```
1 create table DRIVER (  
2     ID_DRIVER          VARCHAR2(5)                not null,  
3     NAMA_DRIVER         VARCHAR2(50),  
4     NO_TEL_DRIVER       VARCHAR2(12),  
5     RATING_DRIVER       FLOAT,  
6     constraint PK_DRIVER primary key (ID_DRIVER)  
7 );
```

4. Membuat tabel restoran

```
1 create table RESTORAN (  
2     ID_RESTORAN         VARCHAR2(10)                not null,  
3     NAMA_RESTORAN       VARCHAR2(50),  
4     ALAMAT_RESTORAN     VARCHAR2(200),  
5     NO_TEL_RESTORAN     VARCHAR2(12),  
6     constraint PK_RESTORAN primary key (ID_RESTORAN)  
7 );
```

5. Membuat tabel makanan

```
1 create table MAKANAN (  
2     ID_MAKANAN          VARCHAR2(10)                not null,  
3     ID_RESTORAN          VARCHAR2(10),  
4     NAMA_MAKANAN         VARCHAR2(100),  
5     HARGA_MAKANAN        INTEGER,  
6     RATING_MAKANAN       FLOAT,  
7     constraint PK_MAKANAN primary key (ID_MAKANAN)  
8 );
```


6. Membuat tabel pesanan

```
1 create table PESANAN (  
2   ID_PESANAN      VARCHAR2(10)          not null,  
3   ID_PEMBELI      VARCHAR2(10),  
4   ID_DRIVER       VARCHAR2(5),  
5   TANGGAL_PEMESANAN  TIMESTAMP,  
6   TOTAL_HARGA      INTEGER,  
7   constraint PK_PESANAN primary key (ID_PESANAN)  
8 );
```

7. Membuat tabel detail pesanan

```
1 create table DETAIL_PESANAN (  
2   ID_PESANAN      VARCHAR2(10)          not null,  
3   ID_MAKANAN      VARCHAR2(10)          not null,  
4   JUMLAH_MAKANAN  NUMBER(2),  
5   constraint PK_DETAIL_PESANAN primary key (ID_PESANAN, ID_MAKANAN)  
6 );
```

8. Membuat tabel rating makanan

```
1 create table RATING_MAKANAN (  
2   ID_PEMBELI      VARCHAR2(10)          not null,  
3   ID_MAKANAN      VARCHAR2(10)          not null,  
4   RATING_MAKANAN  NUMBER(5),  
5   constraint PK_RATING_MAKANAN primary key (ID_PEMBELI, ID_MAKANAN)  
6 );
```

9. Membuat tabel rating driver

```
1 create table RATING_DRIVER (
2   ID_PEMBELI      VARCHAR2(10)          not null,
3   ID_DRIVER       VARCHAR2(5)           not null,
4   RATING_DRIVER   NUMBER(5),
5   constraint PK_RATING_DRIVER primary key (ID_PEMBELI, ID_DRIVER)
6 );
```

10. Membuat tabel c_rating_driver

```
1 create table C_RATING_DRIVER (
2   ID_PEMBELI      VARCHAR2(10)          not null,
3   ID_DRIVER       VARCHAR2(5)           not null,
4   RATING_DRIVER   NUMBER(5),
5   constraint PK_C_RATING_DRIVER primary key (ID_PEMBELI, ID_DRIVER)
6 );
```

11. Membuat tabel c_rating_makanan

```
1 create table C_RATING_MAKANAN (
2   ID_PEMBELI      VARCHAR2(10)          not null,
3   ID_MAKANAN      VARCHAR2(10)          not null,
4   RATING_MAKANAN  NUMBER(5),
5   constraint PK_C_RATING_MAKANAN primary key (ID_PEMBELI, ID_MAKANAN)
6 );
```

12. Membuat tabel c_detail_pesanan

```
1 create table C_DETAIL_PESANAN (
2   ID_PESANAN          VARCHAR2(10)          not null,
3   ID_MAKANAN          VARCHAR2(10)          not null,
4   JUMLAH_MAKANAN      NUMBER(2),
5   constraint PK_C_DETAIL_PESANAN primary key (ID_PESANAN, ID_MAKANAN)
6 );
```

13. Membuat procedure add_makanan

```
1 create or replace procedure add_makanan(nama_makanan in varchar2
2 , harga_makanan in number, id_restoran in varchar2)
3 as
4 begin
5   insert into
6     MAKANAN(id_makanan, nama_makanan, harga_makanan, id_restoran) values
7     ('MKN' || MAKANAN_SEQ.nextval,
8      nama_makanan,
9      harga_makanan,
10     id_restoran);
11 end;
```

Procedure ini berguna untuk menginsertkan data ke tabel makanan tanpa harus menulis query yang panjang.

14. Membuat procedure add_restoran

```
1 create or replace procedure add_restoran(nama_restoran in varchar2
2 , alamat_restoran in varchar2, no_tel_restoran in varchar2)
3 as
4 begin
5     insert into RESTORAN values ('RSTRN' || RESTORAN_SEQ.nextval,
6     nama_restoran,
7     alamat_restoran,
8     no_tel_restoran);
9 end;
```

Procedure ini berguna untuk menginsertkan data ke tabel restoran tanpa harus menulis query yang panjang.

15. Membuat procedure add_pesanan

```
1 create or replace procedure add_pesanan(id_pembeli in varchar2
2 , id_driver in varchar2)
3 as
4 begin
5     insert into
6     PESANAN(id_pesanan, id_pembeli, id_driver, tanggal_pemesanan) values
7     ('PSN' || PESANAN_SEQ.nextval,
8     id_pembeli,
9     id_driver,
10    systimestamp);
11 end;
```

Procedure ini berguna untuk menginsertkan data ke tabel pesanan tanpa harus menulis query yang panjang.

16. Membuat procedure add_detail_pesanan

```
1 create or replace procedure add_detail_pesanan(id_pesanan in varchar2
, id_makanan in varchar2, jumlah_makanan in number)
2 as
3 begin
4     insert into
        DETAIL_PESANAN(id_pesanan, id_makanan, jumlah_makanan) values (
5         id_pesanan,
6         id_makanan,
7         jumlah_makanan);
8 end;
9 /
```

Procedure ini berguna untuk menginsertkan data ke tabel detail_pesanan tanpa harus menulis query yang panjang.

17. Membuat procedure add_rating_driver

```
1 create or replace procedure add_rating_driver(id_pembeli in varchar2
, id_driver in varchar2, rating in number)
2 as
3 begin
4     if (rating > 0 and rating <= 5) then
5         insert into RATING_DRIVER(ID_PEMBELI, ID_DRIVER, RATING_DRIVER)
6         values(id_pembeli, id_driver, rating);
7     else
8         raise_application_error(-20000, 'Rating harus antara 1-5');
9     end if;
10 end;
11 /
```

Procedure ini berguna untuk menginsertkan data ke tabel rating_driver tanpa harus menulis query yang panjang sekaligus memberi peringatan apabila rating yang diberikan tidak sesuai.

18. Membuat procedure add_rating_makanan

```
1 create or replace procedure add_rating_makanan(id_pembeli in varchar2
2 , id_makanan in varchar2, rating in number)
3 as
4 begin
5     if (rating > 0 and rating <= 5) then
6         insert into RATING_MAKANAN(ID_PEMBELI, ID_MAKANAN, RATING_MAKANAN)
7         values(id_pembeli, id_makanan, rating);
8     else
9         raise_application_error(-20000, 'Rating harus antara 1-5');
10    end if;
11 end;
```

Procedure ini berguna untuk menginsertkan data ke tabel rating_makanan tanpa harus menulis query yang panjang sekaligus memberi peringatan apabila rating yang diberikan tidak sesuai.

19. Membuat procedure calculate_rating_driver

```
1 create or replace procedure calculate_rating_driver(id_driver1 in
2 varchar2)
3 as
4 begin
5     update DRIVER
6     set RATING_DRIVER = (select round(avg(RATING_DRIVER),2)
7                          from C_RATING_DRIVER
8                          where ID_DRIVER = id_driver1)
9     where ID_DRIVER = id_driver1;
10 end;
```

Prosedur ini berguna untuk menghitung rata-rata rating dari driver yang sebelumnya telah dinilai oleh pembeli dan mengupdate rating driver dengan rata-rata rating tersebut ke tabel driver.

20. Membuat procedure calculate_rating_makanan

```
1  create or replace procedure calculate_rating_makanan(id_makanan1 in
2  varchar2)
3  as
4  begin
5      update MAKANAN
6          set RATING_MAKANAN = (select round(avg(RATING_MAKANAN),2)
7                                from C_RATING_MAKANAN
7                                where ID_MAKANAN = id_makanan1)
8      where ID_MAKANAN = id_makanan1;
9  end;
10 /
```

Prosedur ini berguna untuk menghitung rata-rata rating dari makanan yang sebelumnya telah dinilai oleh pembeli dan mengupdate rating makanan dengan rata-rata rating tersebut ke tabel makanan.

21. Membuat function calculate_harga_makanan

```
1 create or replace function calculate_harga_makanan(id_pesanan1 in
  varchar2)
2   return INTEGER as
3     total INTEGER := 0;
4     i INTEGER;
5     cursor c_makanan is
6       select HARGA_MAKANAN*JUMLAH_MAKANAN
7       from C_DETAIL_PESANAN join MAKANAN on
8       C_DETAIL_PESANAN.ID_MAKANAN = MAKANAN.ID_MAKANAN
9       where ID_PESANAN = id_pesanan1;
10    begin
11      open c_makanan;
12      loop
13        fetch c_makanan into i;
14        exit when c_makanan%notfound;
15        total := total + i;
16      end loop;
17      close c_makanan;
18      return total;
19    end;
```

Fungsi ini berguna untuk menghitung total harga dari pesanan yang sebelumnya telah dipesan oleh pembeli.

22. Membuat trigger rating_driver_trigger

```
1 create or replace trigger rating_driver_trigger
2   after insert or update on RATING_DRIVER
3   for each row
4   declare
5     jumlah integer;
6     cursor c_jumlah is select count(*) from C_RATING_DRIVER where
ID_DRIVER = :new.ID_DRIVER and id_pembeli = :new.ID_PEMBELI;
7   begin
8     open c_jumlah;
9     fetch c_jumlah into jumlah;
10    if jumlah = 1 then
11      update C_RATING_DRIVER
12        set RATING_DRIVER = :new.RATING_DRIVER
13        where ID_DRIVER = :new.ID_DRIVER and id_pembeli = :new.ID_PEMBELI;
14      calculate_rating_driver(:new.ID_DRIVER);
15    else
16      insert into C_RATING_DRIVER values
17        (:new.ID_PEMBELI, :new.ID_DRIVER, :new.RATING_DRIVER);
18      calculate_rating_driver(:new.ID_DRIVER);
19    end if;
20    close c_jumlah;
21  end;
```

Trigger ini akan aktif setiap tabel rating_driver di insert-kan data, trigger ini kemudian memanggil prosedur calculate_rating_driver agar rata-rata rating dari driver dapat langsung dihitung sekaligus diupdate ke tabel driver.

23. Membuat trigger rating_makanan_trigger

```
1 create or replace trigger rating_makanan_trigger
2   after insert or update on RATING_MAKANAN
3   for each row
4   declare
5     jumlah integer;
6     cursor c_jumlah is select count(*) from C_RATING_MAKANAN where
7       ID_MAKANAN = :new.ID_MAKANAN and id_pembeli = :new.ID_PEMBELI;
8   begin
9     open c_jumlah;
10    fetch c_jumlah into jumlah;
11    if jumlah = 1 then
12      update C_RATING_MAKANAN
13        set RATING_MAKANAN = :new.RATING_MAKANAN
14        where ID_MAKANAN = :new.ID_MAKANAN and id_pembeli = :new.ID_PEMBELI;
15      calculate_rating_makanan(:new.ID_MAKANAN);
16    else
17      insert into C_RATING_MAKANAN values
18        (:new.ID_PEMBELI, :new.ID_MAKANAN, :new.RATING_MAKANAN);
19      calculate_rating_makanan(:new.ID_MAKANAN);
20    end if;
21    close c_jumlah;
22  end;
```

Trigger ini akan aktif setiap tabel rating_makanan di insert-kan data, trigger ini kemudian memanggil prosedur calculate_rating_makanan agar rata-rata rating dari makanan dapat langsung dihitung sekaligus di update ke tabel driver.

24. Membuat trigger harga_makanan_trigger

```
1 create or replace trigger harga_makanan_trigger
2   after insert on DETAIL_PESANAN
3   for each row
4   begin
5       insert into C_DETAIL_PESANAN values
6       (:new.ID_PESANAN, :new.ID_MAKANAN, :new.JUMLAH_MAKANAN);
7       update PESANAN
8       set TOTAL_HARGA = calculate_harga_makanan(:new.ID_PESANAN)
9       where ID_PESANAN = :new.ID_PESANAN;
10  end;
```

Trigger ini akan aktif setiap tabel detail_pesanan di insert-kan data, trigger ini kemudian memanggil prosedur calculate_harga_makanan agar total harga dari setiap pesanan dapat langsung dihitung sekaligus di update ke tabel pesanan.

25. Menginsertkan data driver ke tabel driver

```
1 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Fred Siggins', '3565503488');
2 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Tuesday Bernette', '1626310621');
3 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Cati Malcher', '2476865272');
4 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Natka Scoterbosh', '9887739855');
5 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Jens Sepey', '8191173250');
6 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Nathalie Scyner', '9431529488');
7 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Nixie Dast', '9464049527');
8 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Margarethe Killelea', '4244328695');
9 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Gates O'Shiel', '5989615383');
10 insert into driver(id_driver,nama_driver,no_tel_driver) values ('DRV'|| DRIVER_SEQ.NEXTVAL,
  'Edie Keegan', '3121534792');
```

```
SQL> select * from driver;
```

ID_DR	NAMA_DRIVER	NO_TEL_DRIVE	RATING_DRIVER
DRV1	Fred Siggins	3565503488	5
DRV2	Tuesday Bernette	1626310621	5
DRV3	Cati Malcher	2476865272	4,5
DRV4	Natka Scoterbosh	9887739855	5
DRV5	Jens Sepey	8191173250	5
DRV6	Nathalie Scyner	9431529488	3,5
DRV7	Nixie Dast	9464049527	3,5
DRV8	Margarethe Killelea	4244328695	3
DRV9	Gates O'Shiel	5989615383	3,5
DRV10	Edie Keegan	3121534792	4

```
10 rows selected.
```

26. Menginsertkan data pembeli ke tabel pembeli

```
1 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Nate Creer', '9 Sheridan Hill', '5108504490',
  'ncreer0@blogtalkradio.com');
2 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Layney Claybourn', '5132 Killdeer Center', '9257139232',
  'lclaybourn1@elpais.com');
3 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Sim Lyford', '51992 Fisk Parkway', '2952790094',
  'slyford2@who.int');
4 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Ketty McNae', '48 Trailsway Road', '2508041550',
  'kmcnae3@topsy.com');
5 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Innis Warke', '73331 Valley Edge Terrace', '4154221081',
  'iwarke4@mayoclinic.com');
6 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Nert Downton', '57 Fallview Lane', '8609793000',
  'ndownton5@microsoft.com');
7 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Ruprecht Beany', '2 Beilfuss Crossing', '4031008779',
  'rbeany6@jiathis.com');
8 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Baillie Storey', '45 Corben Hill', '4911063374',
  'bstorey7@theforest.net');
9 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Nicholas Phelan', '32 Sachs Park', '2576878482',
  'nphelan8@reference.com');
10 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Angie Winnard', '6 Shelley Court', '5959395700',
  'awinnard9@deliciousdays.com');
11 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Raymond Sleaford', '603 Spohn Park', '3834942010',
  'rsleaforda@fastcompany.com');
12 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Hadria Twyford', '052 Lakewood Street', '4136752644',
  'htwyfordb@twitter.com');
13 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Tulley Choppin', '61497 Dennis Parkway', '4503701678',
  'tchoppinc@360.cn');
14 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Aeriell Izak', '29 South Drive', '3298733966',
  'aizakd@addthis.com');
15 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Dania Kirkness', '923 Sheridan Circle', '6697116708',
  'dkirknesse@apple.com');
16 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Merilee Baile', '36572 Declaration Park', '6367088258',
  'mbailef@baidu.com');
17 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Mariellen Anselmi', '63 Orin Parkway', '8921546652',
  'manselmig@nih.gov');
18 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Dacia Kield', '8538 David Court', '4231177942', 'dkieldh@furl.net'
  );
19 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Stanislaus Hendrichs', '0673 Trailsway Alley', '1886486922',
  'shendrichsi@wp.com');
20 insert into PEMBELI values ('PBLI'|| PEMBELI_SEQ.NEXTVAL,'Mollee Lamasna', '2239 Hallows Point', '2823261120',
  'mlamasnaj@tinypic.com');
```

SQL> select * from pembeli;

ID_PEMBELI	NAMA_PEMBELI	ALAMAT_PEMBELI	NO_TEL_PEMBELI	EMAIL
PBLI1	Nate Creer	9 Sheridan Hill	5108504490	ncreer0@blogtalkradio.com
PBLI2	Layney Claybourn	5132 Killdeer Center	9257139232	lclaybourn1@elpais.com
PBLI3	Sim Lyford	51992 Fisk Parkway	2952790094	slyford2@who.int
PBLI4	Ketty McNae	48 Trailsway Road	2508041550	kmcnae3@topsy.com
PBLI5	Innis Warke	73331 Valley Edge Terrace	4154221081	iwarke4@mayoclinic.com
PBLI6	Nert Downton	57 Fallview Lane	8609793000	ndownton5@microsoft.com
PBLI7	Ruprecht Beany	2 Beilfuss Crossing	4031008779	rbeany6@jiathis.com
PBLI8	Baillie Storey	45 Corben Hill	4911063374	bstorey7@theforest.net
PBLI9	Nicholas Phelan	32 Sachs Park	2576878482	nphelan8@reference.com
PBLI10	Angie Winnard	6 Shelley Court	5959395700	awinnard9@deliciousdays.com
PBLI11	Raymond Sleaford	603 Spohn Park	3834942010	rsleaforda@fastcompany.com
PBLI12	Hadria Twyford	052 Lakewood Street	4136752644	htwyfordb@twitter.com
PBLI13	Tulley Choppin	61497 Dennis Parkway	4503701678	tchoppinc@360.cn
PBLI14	Aeriell Izak	29 South Drive	3298733966	aizakd@addthis.com
PBLI15	Dania Kirkness	923 Sheridan Circle	6697116708	dkirknesse@apple.com
PBLI16	Merilee Baile	36572 Declaration Park	6367088258	mbailef@baidu.com
PBLI17	Mariellen Anselmi	63 Orin Parkway	8921546652	manselmig@nih.gov
PBLI18	Dacia Kield	8538 David Court	4231177942	dkieldh@furl.net
PBLI19	Stanislaus Hendrichs	0673 Trailsway Alley	1886486922	shendrichsi@wp.com
PBLI20	Mollee Lamasna	2239 Hallows Point	2823261120	mlamasnaj@tinypic.com

20 rows selected.

27. Menginsertkan data restoran ke tabel restoran

```
1 execute add_restoran('Soto Bancar Bu Misdar','Jl. Letjen S Parman No.14, Bancar, Kec. Purbalingga, Kabupaten Purbalingga',
2 '6210223144');
3 execute add_restoran('RM Wapo Kebon Kelapa',
4 'Jalan Ketuhu, Purbalingga Lor, Kecamatan Purbalingga, Wirasana, Kec. Purbalingga, Kabupaten Purbalingga, Jawa Tengah',
5 '081903024327');
6 execute add_restoran('Ayam Penyet Surabaya',
7 'Jl. Jend. Sudirman No.126, Purbalingga, Purbalingga Lor, Kec. Purbalingga, Kabupaten Purbalingga','0281894863');
8 execute add_restoran('Waroeng Djoglo Purbalingga','Jalan Kanoman No.3-5, Bancar, Kec. Purbalingga, Kabupaten Purbalingga',
9 '0281896540');
```

```
SQL> select * from restoran;
```

ID_RESTORA	NAMA_RESTORAN	ALAMAT_RESTORAN	NO_TEL_RESTORAN
RSTRN1	Soto Bancar Bu Misdar	Jl. Letjen S Parman No.14, Bancar, Kec. Purbalingga, Kabupaten Purbalingga	6210223144
RSTRN2	RM Wapo Kebon Kelapa	Jalan Ketuhu, Purbalingga Lor, Kecamatan Purbalingga, Wirasana, Kec. Purbalingga, Kabupaten Purbalin gga, Jawa Tengah	081903024327
RSTRN3	Ayam Penyet Surabaya	Jl. Jend. Sudirman No.126, Purbalingga, Purbalingga Lor, Kec. Purbalingga, Kabupaten Purbalingga	0281894863
RSTRN4	Waroeng Djoglo Purbalingga	Jalan Kanoman No.3-5, Bancar, Kec. Purbalingga, Kabupaten Purbalingga	0281896540

28. Menginsertkan data makanan ke tabel makanan

```
1 execute add_makanan('Soto Ayam',15000,'RSTRN1');
2 execute add_makanan('Soto Sapi',25000,'RSTRN1');
3 execute add_makanan('Soto Campur',20000,'RSTRN1');
4 execute add_makanan('Ikan Gurame',24000,'RSTRN2');
5 execute add_makanan('Ikan Bali',30000,'RSTRN2');
6 execute add_makanan('Nasi Goreng',10000,'RSTRN2');
7 execute add_makanan('Nasi Uduk',15000,'RSTRN2');
8 execute add_makanan('Nasi Pecel',20000,'RSTRN2');
9 execute add_makanan('Ayam Goreng',13000,'RSTRN2');
10 execute add_makanan('Ayam Bakar',15000,'RSTRN2');
11 execute add_makanan('Ayam Bakar',20000,'RSTRN3');
12 execute add_makanan('Ayam Penyet',25000,'RSTRN3');
13 execute add_makanan('Lele Penyet',23000,'RSTRN3');
14 execute add_makanan('Nila Penyet',25000,'RSTRN3');
15 execute add_makanan('Gurame Penyet Bakar',35000,'RSTRN3');
16 execute add_makanan('Bebek Penyet',31000,'RSTRN3');
17 execute add_makanan('Ayam Goreng Kampung',20000,'RSTRN4');
18 execute add_makanan('Ayam Goreng Broiler',15000,'RSTRN4');
19 execute add_makanan('Bandeng Presto Goreng',25000,'RSTRN4');
20 execute add_makanan('Ikan Uceng',11000,'RSTRN4');
21 execute add_makanan('Pete Goreng',13000,'RSTRN4');
```

```
SQL> select * from makanan;
```

ID_MAKANAN	ID_RESTORA	NAMA_MAKANAN	HARGA_MAKANAN	RATING_MAKANAN
MKN1	RSTRN1	Soto Ayam	15000	4
MKN2	RSTRN1	Soto Sapi	25000	5
MKN3	RSTRN1	Soto Campur	20000	4,5
MKN4	RSTRN2	Ikan Gurame	24000	4
MKN5	RSTRN2	Ikan Bali	30000	3
MKN6	RSTRN2	Nasi Goreng	10000	3
MKN7	RSTRN2	Nasi Uduk	15000	5
MKN8	RSTRN2	Nasi Pecel	20000	5
MKN9	RSTRN2	Ayam Goreng	13000	2,5
MKN10	RSTRN2	Ayam Bakar	15000	4,5
MKN11	RSTRN3	Ayam Bakar	20000	3

ID_MAKANAN	ID_RESTORA	NAMA_MAKANAN	HARGA_MAKANAN	RATING_MAKANAN
MKN12	RSTRN3	Ayam Penyet	25000	
MKN13	RSTRN3	Lele Penyet	23000	
MKN14	RSTRN3	Nila Penyet	25000	
MKN15	RSTRN3	Gurame Penyet Bakar	35000	
MKN16	RSTRN3	Bebek Penyet	31000	
MKN17	RSTRN4	Ayam Goreng Kampung	20000	
MKN18	RSTRN4	Ayam Goreng Broiler	15000	
MKN19	RSTRN4	Bandeng Presto Goreng	25000	
MKN20	RSTRN4	Ikan Uceng	11000	
MKN21	RSTRN4	Pete Goreng	13000	

```
21 rows selected.
```

29. Menginsertkan data pesanan ke tabel pesanan

```
1 execute add_pesanan('PBLI1','DRV1');
2 execute add_pesanan('PBLI2','DRV2');
3 execute add_pesanan('PBLI3','DRV3');
4 execute add_pesanan('PBLI4','DRV4');
5 execute add_pesanan('PBLI5','DRV5');
6 execute add_pesanan('PBLI6','DRV6');
7 execute add_pesanan('PBLI7','DRV7');
8 execute add_pesanan('PBLI8','DRV8');
9 execute add_pesanan('PBLI9','DRV9');
10 execute add_pesanan('PBLI10','DRV10');
```

```
SQL> select * from pesanan;
```

ID_PESANAN	ID PEMBELI	ID_DR	TANGGAL_PEMESANAN	TOTAL_HARGA
PSN1	PBLI1	DRV1	05-06-2022 09.24.41,871000	60000
PSN2	PBLI2	DRV2	05-06-2022 09.24.41,874000	108000
PSN3	PBLI3	DRV3	05-06-2022 09.24.41,880000	30000
PSN4	PBLI4	DRV4	05-06-2022 09.24.41,881000	35000
PSN5	PBLI5	DRV5	05-06-2022 09.24.41,896000	28000
PSN6	PBLI6	DRV6	05-06-2022 09.24.41,911000	
PSN7	PBLI7	DRV7	05-06-2022 09.24.41,927000	
PSN8	PBLI8	DRV8	05-06-2022 09.24.41,928000	
PSN9	PBLI9	DRV9	05-06-2022 09.24.41,942000	
PSN10	PBLI10	DRV10	05-06-2022 09.24.43,186000	

```
10 rows selected.
```


30. Menginsertkan data detail pesanan ke tabel detail_pesanan

```
1 execute add_detail_pesanan('PSN1','MKN1',1);
2 execute add_detail_pesanan('PSN1','MKN2',1);
3 execute add_detail_pesanan('PSN1','MKN3',1);
4 execute add_detail_pesanan('PSN2','MKN4',2);
5 execute add_detail_pesanan('PSN2','MKN5',2);
6 execute add_detail_pesanan('PSN3','MKN6',3);
7 execute add_detail_pesanan('PSN4','MKN7',1);
8 execute add_detail_pesanan('PSN4','MKN8',1);
9 execute add_detail_pesanan('PSN5','MKN9',1);
10 execute add_detail_pesanan('PSN5','MKN10',1);
```

```
SQL> select * from detail_pesanan;
```

ID_PESANAN	ID_MAKANAN	JUMLAH_MAKANAN
PSN1	MKN1	1
PSN1	MKN2	1
PSN1	MKN3	1
PSN2	MKN4	2
PSN2	MKN5	2
PSN3	MKN6	3
PSN4	MKN7	1
PSN4	MKN8	1
PSN5	MKN9	1
PSN5	MKN10	1

```
10 rows selected.
```

31. Menginsertkan data rating setiap driver ke tabel rating_driver

```
1 execute add_rating_driver('PBLI1', 'DRV1', 5);
2 execute add_rating_driver('PBLI1', 'DRV2', 5);
3 execute add_rating_driver('PBLI1', 'DRV3', 4);
4 execute add_rating_driver('PBLI1', 'DRV4', 5);
5 execute add_rating_driver('PBLI1', 'DRV5', 5);
6 execute add_rating_driver('PBLI1', 'DRV6', 2);
7 execute add_rating_driver('PBLI1', 'DRV7', 3);
8 execute add_rating_driver('PBLI1', 'DRV8', 5);
9 execute add_rating_driver('PBLI1', 'DRV9', 4);
10 execute add_rating_driver('PBLI1', 'DRV10', 5);
11 execute add_rating_driver('PBLI2', 'DRV1', 5);
12 execute add_rating_driver('PBLI2', 'DRV2', 5);
13 execute add_rating_driver('PBLI2', 'DRV3', 5);
14 execute add_rating_driver('PBLI2', 'DRV4', 5);
15 execute add_rating_driver('PBLI2', 'DRV5', 5);
16 execute add_rating_driver('PBLI2', 'DRV6', 5);
17 execute add_rating_driver('PBLI2', 'DRV7', 4);
18 execute add_rating_driver('PBLI2', 'DRV8', 1);
19 execute add_rating_driver('PBLI2', 'DRV9', 3);
20 execute add_rating_driver('PBLI2', 'DRV10', 3);
```

```
SQL> select * from rating_driver;
```

ID_PEMBELI	ID_DR	RATING_DRIVER
PBLI1	DRV1	5
PBLI1	DRV2	5
PBLI1	DRV3	4
PBLI1	DRV4	5
PBLI1	DRV5	5
PBLI1	DRV6	2
PBLI1	DRV7	3
PBLI1	DRV8	5
PBLI1	DRV9	4
PBLI1	DRV10	5
PBLI2	DRV1	5

ID_PEMBELI	ID_DR	RATING_DRIVER
PBLI2	DRV2	5
PBLI2	DRV3	5
PBLI2	DRV4	5
PBLI2	DRV5	5
PBLI2	DRV6	5
PBLI2	DRV7	4
PBLI2	DRV8	1
PBLI2	DRV9	3
PBLI2	DRV10	3

```
20 rows selected.
```

32. Menginsertkan data rating setiap makanan ke tabel rating_makanan

```
1 execute add_rating_makanan('PBLI1', 'MKN1', 5);
2 execute add_rating_makanan('PBLI1', 'MKN2', 5);
3 execute add_rating_makanan('PBLI1', 'MKN3', 4);
4 execute add_rating_makanan('PBLI1', 'MKN4', 5);
5 execute add_rating_makanan('PBLI1', 'MKN5', 1);
6 execute add_rating_makanan('PBLI1', 'MKN6', 2);
7 execute add_rating_makanan('PBLI1', 'MKN7', 5);
8 execute add_rating_makanan('PBLI1', 'MKN8', 5);
9 execute add_rating_makanan('PBLI1', 'MKN9', 3);
10 execute add_rating_makanan('PBLI1', 'MKN10', 5);
11 execute add_rating_makanan('PBLI2', 'MKN1', 5);
12 execute add_rating_makanan('PBLI2', 'MKN2', 5);
13 execute add_rating_makanan('PBLI2', 'MKN3', 5);
14 execute add_rating_makanan('PBLI2', 'MKN4', 3);
15 execute add_rating_makanan('PBLI2', 'MKN5', 5);
16 execute add_rating_makanan('PBLI2', 'MKN6', 4);
17 execute add_rating_makanan('PBLI2', 'MKN7', 5);
18 execute add_rating_makanan('PBLI2', 'MKN8', 5);
19 execute add_rating_makanan('PBLI2', 'MKN9', 2);
20 execute add_rating_makanan('PBLI2', 'MKN10', 4);
21 execute add_rating_makanan('PBLI2', 'MKN11', 3);
```

```
SQL> select * from rating_makanan;
```

ID_PEMBELI	ID_MAKANAN	RATING_MAKANAN
PBLI1	MKN1	3
PBLI1	MKN2	5
PBLI1	MKN3	4
PBLI1	MKN4	5
PBLI1	MKN5	1
PBLI1	MKN6	2
PBLI1	MKN7	5
PBLI1	MKN8	5
PBLI1	MKN9	3
PBLI1	MKN10	5
PBLI2	MKN1	5

ID_PEMBELI	ID_MAKANAN	RATING_MAKANAN
PBLI2	MKN2	5
PBLI2	MKN3	5
PBLI2	MKN4	3
PBLI2	MKN5	5
PBLI2	MKN6	4
PBLI2	MKN7	5
PBLI2	MKN8	5
PBLI2	MKN9	2
PBLI2	MKN10	4
PBLI2	MKN11	3

```
21 rows selected.
```

33. Melihat info pesanan dengan view info_pesanan

```
1 create or replace view info_pesanan as
2 select id_pesanan, nama_pembeli, nama_driver, alamat_pembeli, tanggal_pemesanan, total_harga
3      from pesanan join pembeli on pesanan.id_pembeli = pembeli.id_pembeli
4      join driver on pesanan.id_driver = driver.id_driver;
```

```
SQL> select * from info_pesanan;
```

ID_PESANAN	NAMA_PEMBELI	NAMA_DRIVER	ALAMAT_PEMBELI	TANGGAL_PEMESANAN	TOTAL_HARGA
PSN1	Nate Creer	Fred Siggins	9 Sheridan Hill	05-06-2022 09.24.41,871000	60000
PSN2	Layne Claybourn	Tuesday Bernette	5132 Killdeer Center	05-06-2022 09.24.41,874000	108000
PSN3	Sim Lyford	Cati Malcher	51992 Fisk Parkway	05-06-2022 09.24.41,880000	30000
PSN4	Ketty McNae	Natka Scoterbosh	48 Trailsway Road	05-06-2022 09.24.41,881000	35000
PSN5	Innis Warke	Jens Sepey	73331 Valley Edge Terrace	05-06-2022 09.24.41,896000	28000
PSN6	Nert Downton	Nathalie Scyner	57 Fallview Lane	05-06-2022 09.24.41,911000	
PSN7	Ruprecht Beany	Nixie Dast	2 Beilfuss Crossing	05-06-2022 09.24.41,927000	
PSN8	Bailie Storey	Margarethe Killelea	45 Corben Hill	05-06-2022 09.24.41,928000	
PSN9	Nicholas Phelan	Gates O'Shiel	32 Sachs Park	05-06-2022 09.24.41,942000	
PSN10	Angie Winnard	Edie Keegan	6 Shelley Court	05-06-2022 09.24.43,186000	

```
10 rows selected.
```