Tugas-SDB2-03

SISTEM DATABASE II

Disusun Oleh:

140810210015 – Amir Salim

140810210029 – Rifqi Muhadzib Ahdan

140810210041 – Muhammad Fauzan Azhima

PROGRAM STUDI S-1 TEKNIK INFORMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS PADJADJARAN

JATINANGOR

2022

BAGIAN I

1. Aktifkan login HR pada masing-masing server database (login dengan user system melalui enterprise manager console)
2. Buatlah konfigurasi database link yang melibatkan tiga buah komputer server database oracle yang saling terhubung antara satu dengan yang lain. (komputer server bisa disimulasikan menggunakan virtual machine/ Virtual Box). Lihat gambar dibawah (arah tanda panah merupakan arah dari koneksi database link). Komputer server A dapat mengakses data pada komputer server B. Sedangkan komputer server B dapat mengakases data pada komputer server C dan sebaliknya komputer server C juga dapat mengakses komputer server B.
3. Buatlah table Salary pada server C, yang memiliki isi EMPLOYEE_ID, SAL_ID, TOTAL_SALARY, DATE. Kemudian isikan data dummy (data simulasi) pada table tersebut. Lakukan Query pada server A untuk mengambil data Salary yang ada pada server C Dengan menggunakan database link yang terdapat pada Server A, tambahkan data berikut pada table Region yang terdapat pada Server C. (DML dilakukan pada server A)

country_id	country_name	region_id
ID	Indonesia	6
MY	Malaysia	6
SG	Singapore	6
TH	Thailand	6
BD	Brunai	6

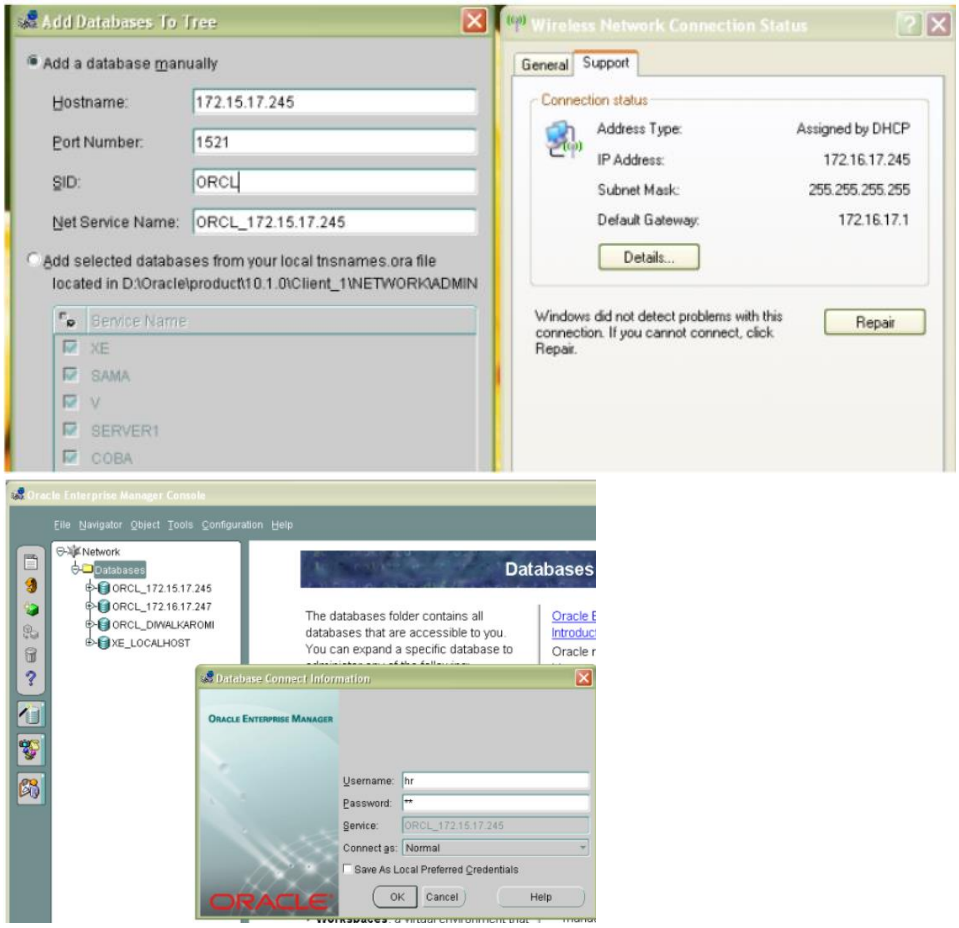
4. . Sesuai dengan konfigurasi sebelumnya, bisakah dari server database C, melakukan query ke server database A, (bisa/tidak)? Jelaskan secara logis alasan Anda (kenapa bisa/tidak) !!!
5. Dari server Database A buatlah view yang digunakan untuk menampilkan seluruh Country yang ada di server B dan berada di wilayah Region South East Asia (ingat!!! Data region South East Asia hanya terdapat di server C).
6. Dari server A buatlah query untuk menampilkan data Countries di server B yang tidak dimiliki oleh server C.
7. Dari server A buatlah satu buah DML yang digunakan untuk menambahkan data Countries di server C, sehingga data tabel Countries yang ada pada server C, sama dengan yang ada pada server B.

Catatan:

Gunakan SQL SYNONYM untuk membuat nama alias pada suatu schema objek database link

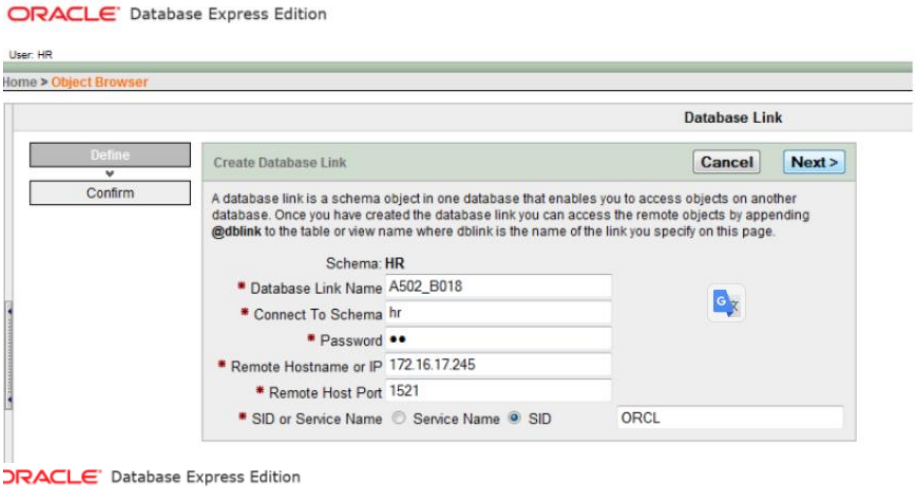
JAWABAN

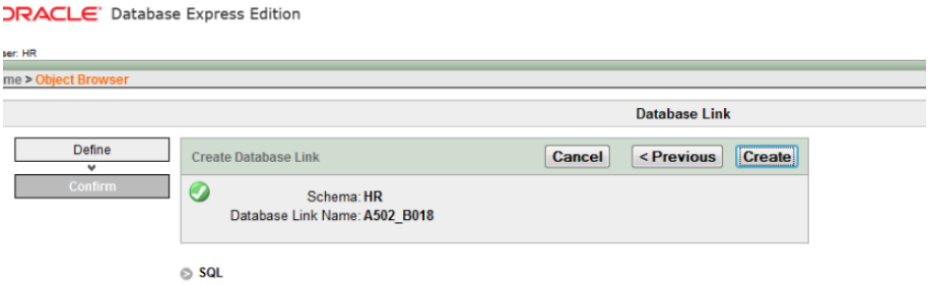
1. Mengaktifkan Login HR

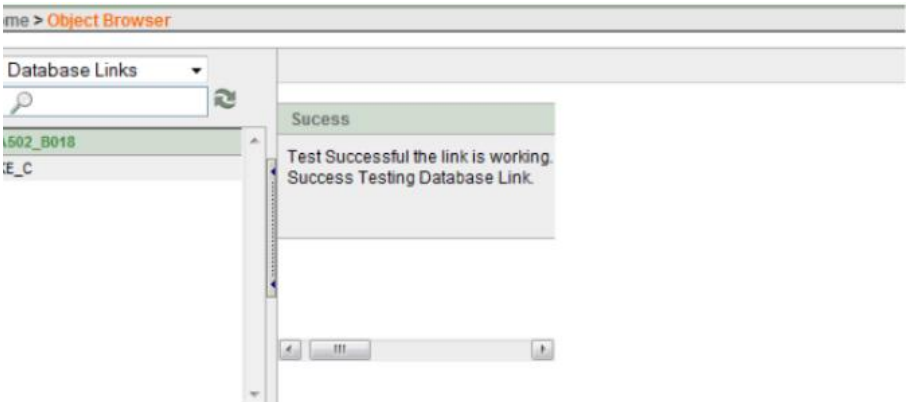


2. Membuat konfigurasi Database Link
   Informasi alamat ip tiap server :
   Server A: 172.16.17.247
   Server B: 172.16.17.245
   Server C: 172.16.17.149

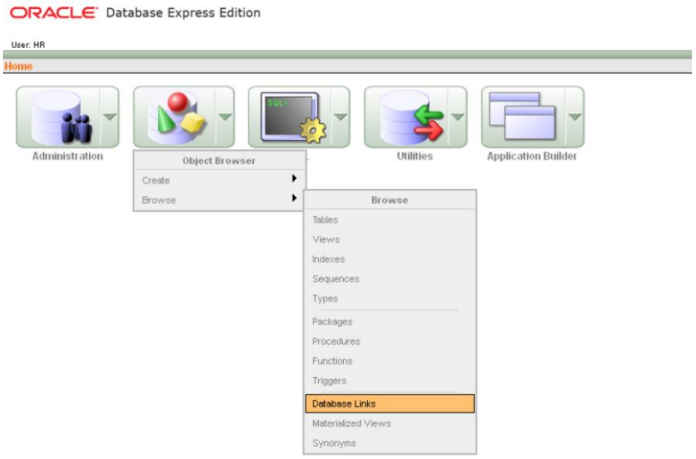
- Db link Server A → Server B

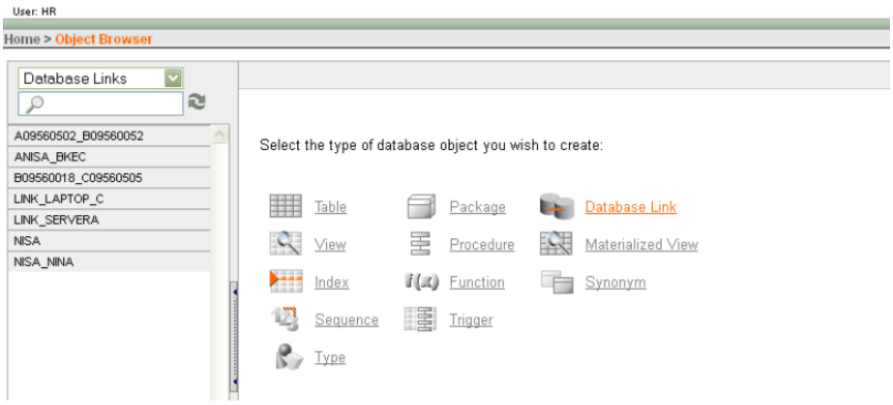


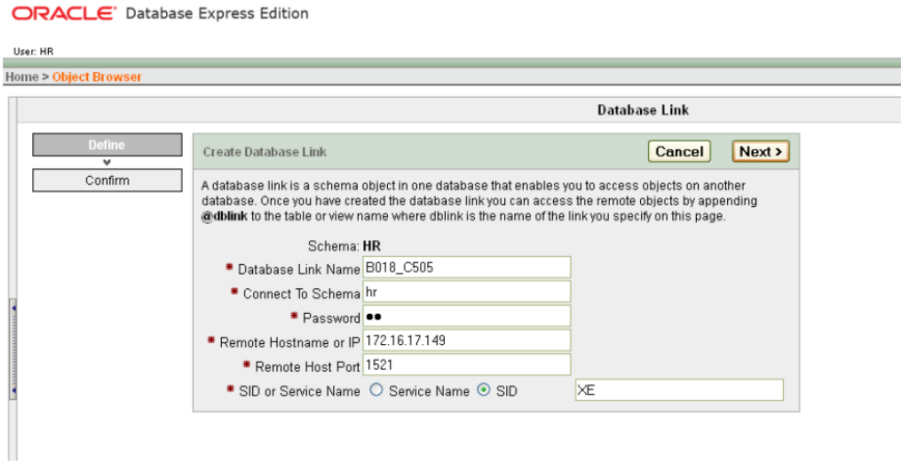


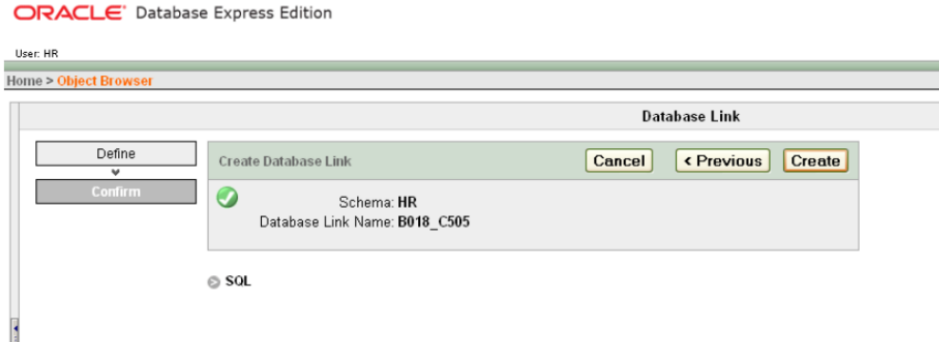


- Db link Server B → Server C

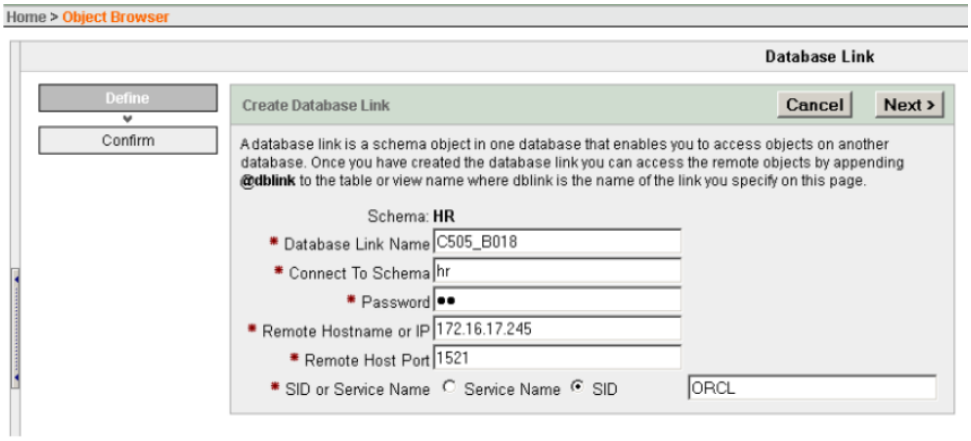


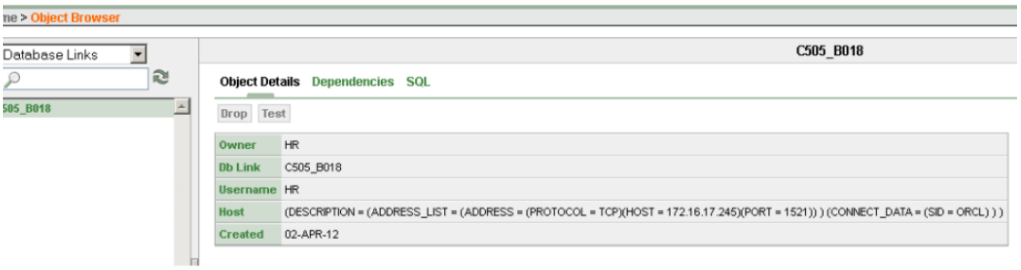


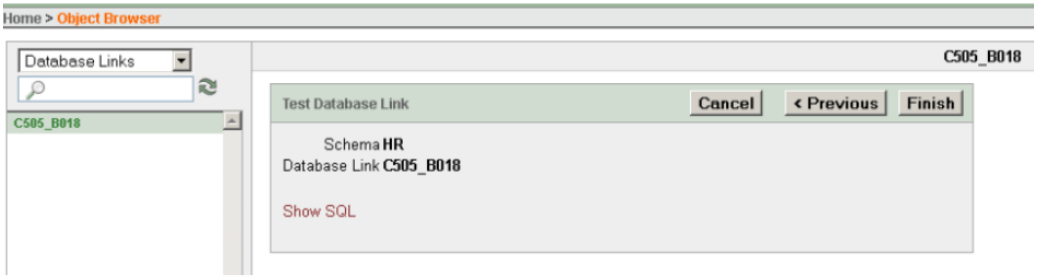




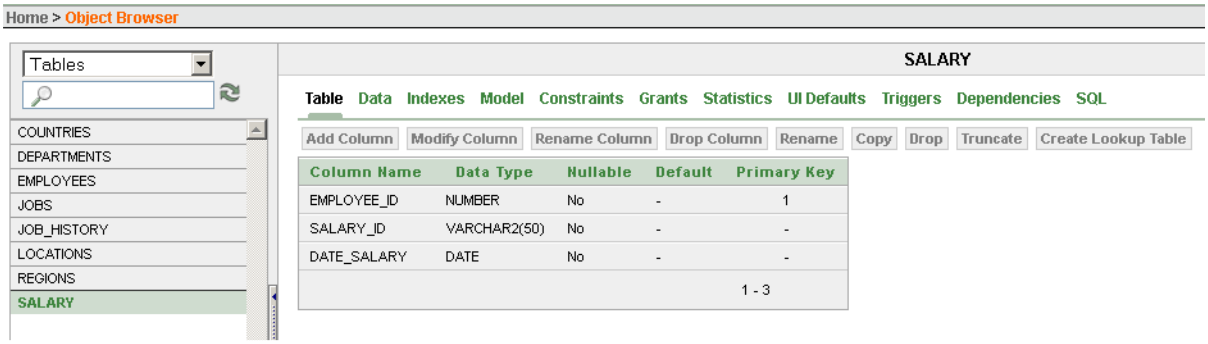
- Db link Server C → Server B

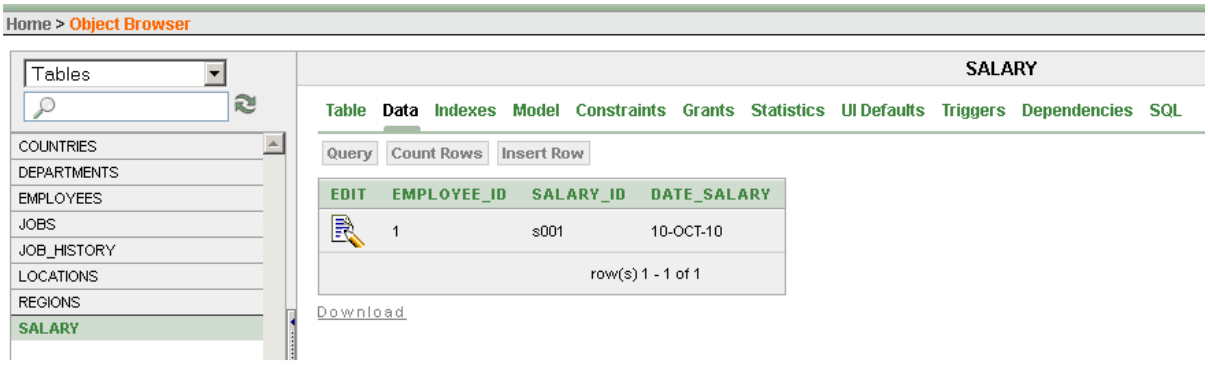




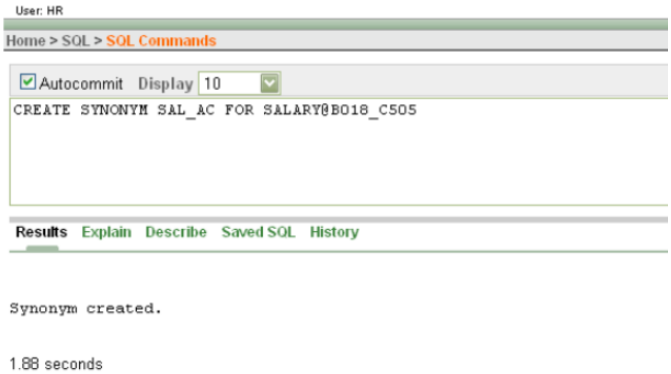


3. Membuat table

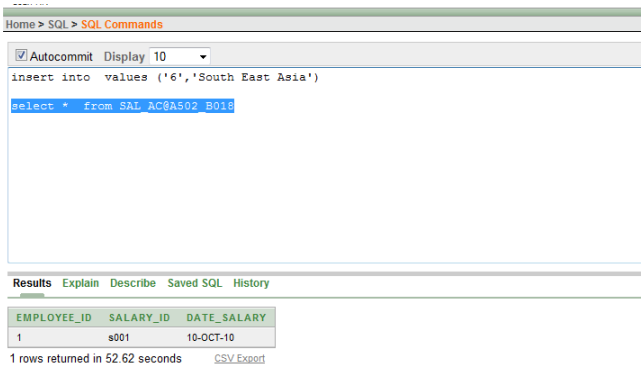




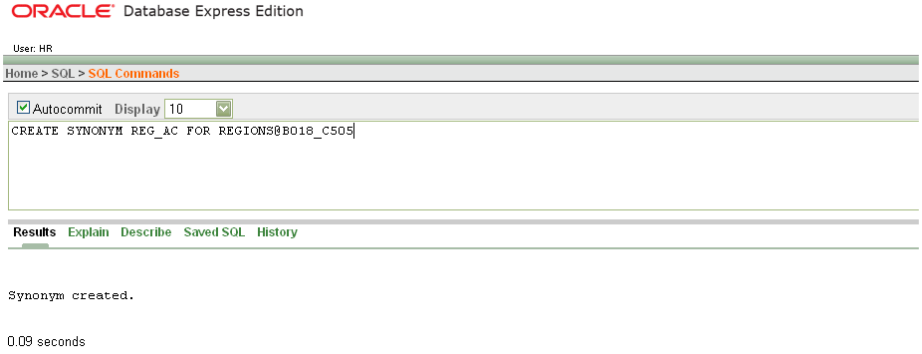
Setelah server C membuat table salary, server B membuat synonym untuk koneksi server A dengan server C karena kedua server tersebut tidak memiliki database link yang menghubungkannya.



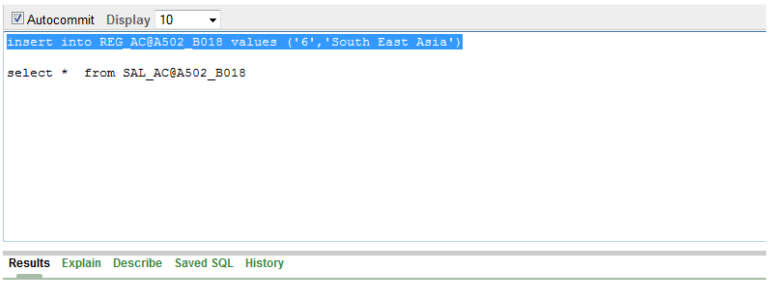
Server A menampilkan Tabel SALARY di dengan synonym table salary



Synonym table regions oleh server B

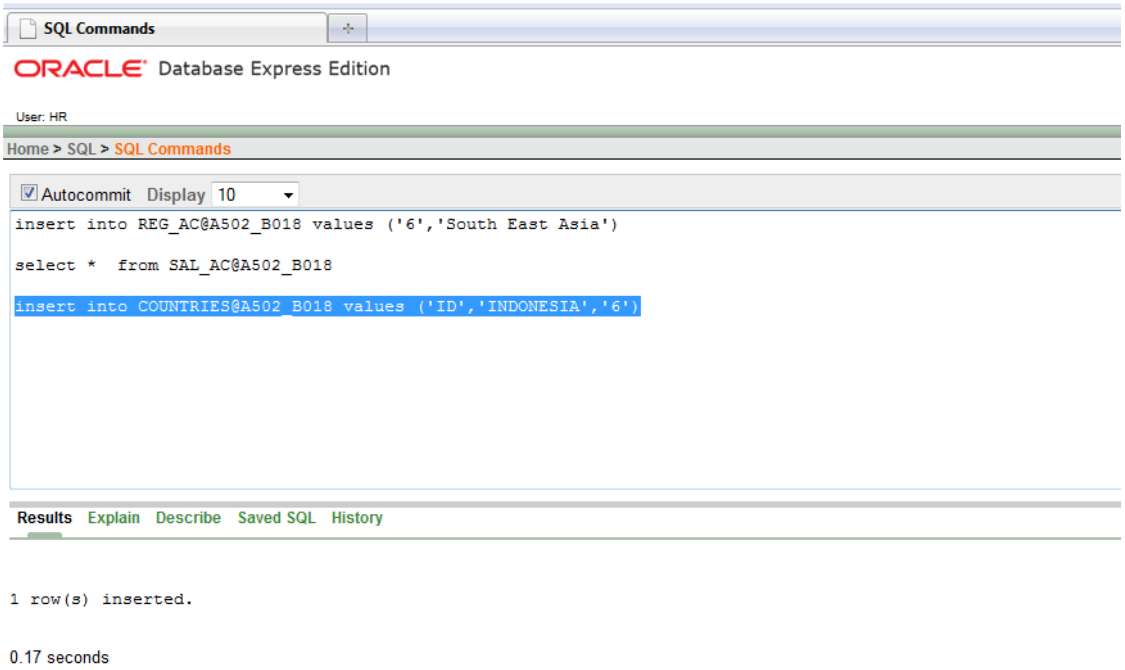


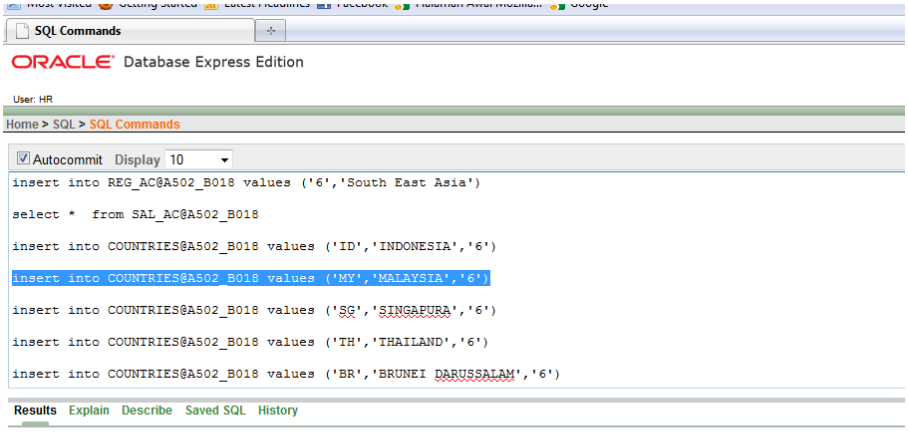
Menginsertkan Data “6, South East Asia” Pada Tabel REGION

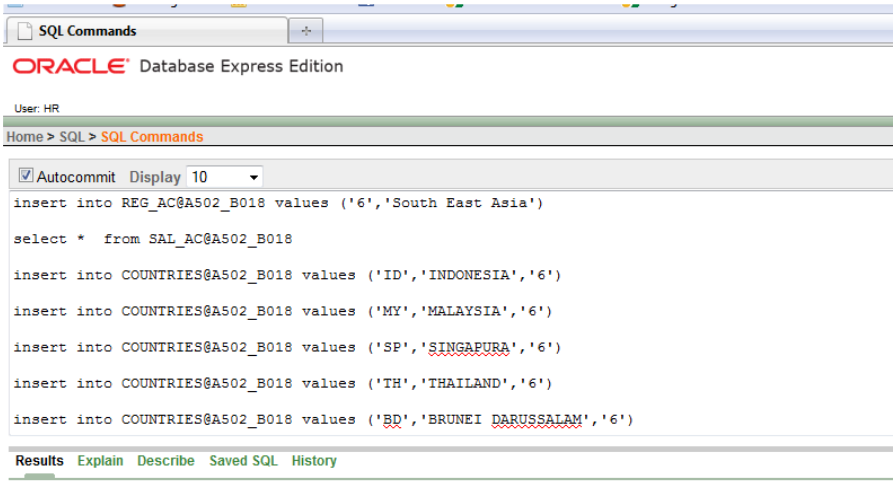


Pengecekan table regions oleh server C

Menambahakan data di Tabel COUNTRIES yang ada di server B melalui server A







Pengecekan data country yang telah dimasukkan oleh server A ke server C pada table countries oleh server B

4. Server database C tidak bisa melakukan query terhadap server database A. Karena komputer server C tidak terhubung secara langsung dengan komputer server A. Walaupun komputer server C dapat mengakses komputer server B yang terhubung dengan komputer server A, akan tetapi komputer server B tidak memiliki akses dengan komputer server A. ini dikarenakan komputer server A hanya memiliki koneksi satu arah terhadap komputer server B.

5. Dari server Database A buatlah view yang digunakan untuk menampilkan seluruh Country yang ada di server B dan berada di wilayah Region South East Asia (ingat!!! Data region South East Asia hanya terdapat di server C).

   Query :

```
CREATE VIEW negara AS
SELECT *
FROM COUNTRIES@A502_B018 a
JOIN REG_AC@A502_B018 b
ON b.REGION_ID = a.Region_ID
WHERE REGION_NAME = 'South East Asia';

SELECT * FROM negara;
```

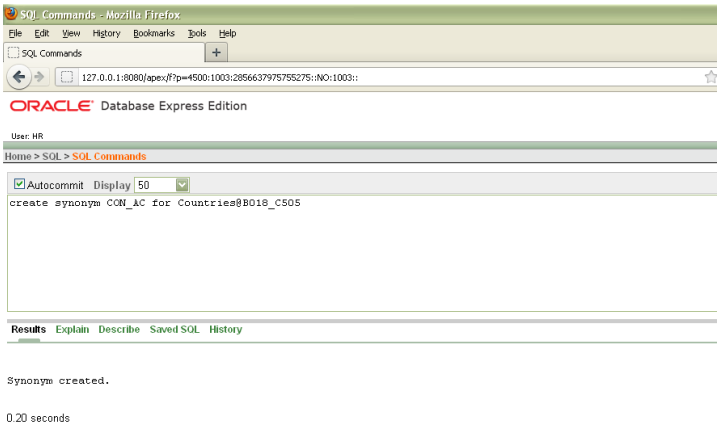
   Hasil :

Results Explain Describe Saved SQL History

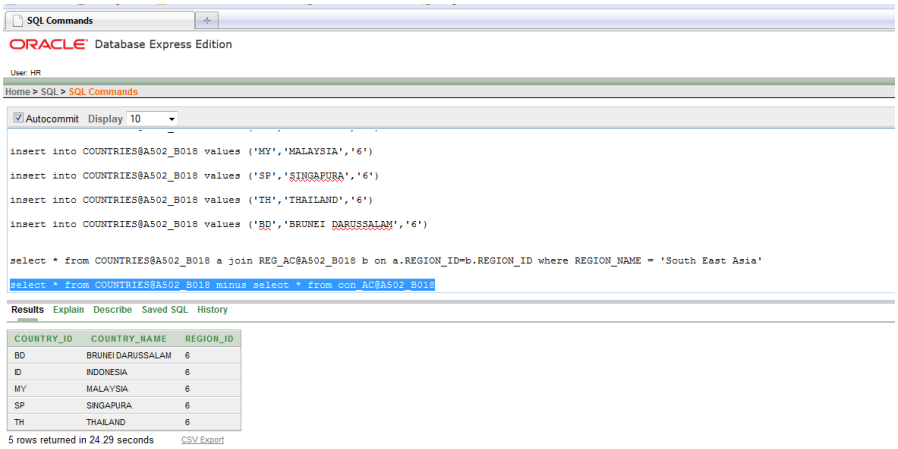
COUNTRY_ID	COUNTRY_NAME	REGION_ID	REGION_ID	REGION_NAME
BD	BRUNEI DARUSSALAM	6	6	South East Asia
ID	INDONESIA	6	6	South East Asia
MY	MALAYSIA	6	6	South East Asia
SP	SINGAPURA	6	6	South East Asia
TH	THAILAND	6	6	South East Asia

5 rows returned in 3.04 seconds CSV Export

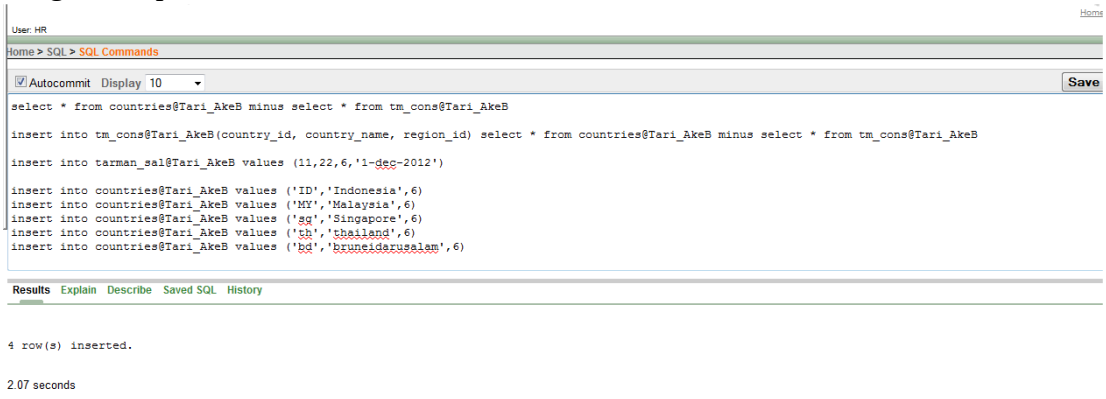
6. Dari server A buatlah query untuk menampilkan data Countries di server B yang tidak dimiliki oleh server C.



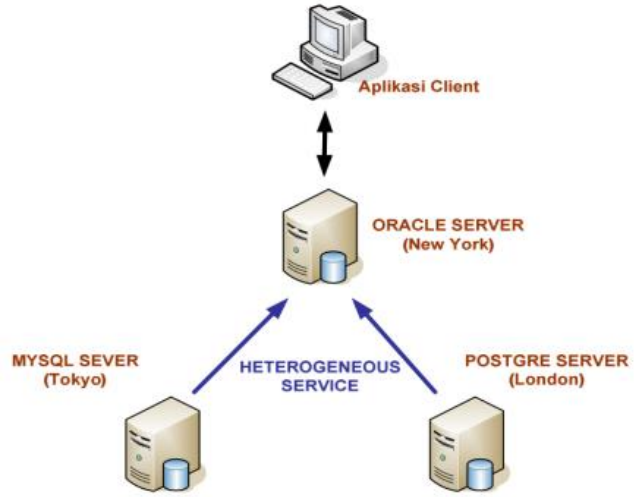
Hasil dari country yang dimiliki oleh server A tetapi tidak dimiliki oleh server B menggunakan query minus



7. Dari server A buatlah satu buah DML yang digunakan untuk menambahkan data Countries di server C, sehingga data tabel Countries yang ada pada server C, sama dengan ada pada server B

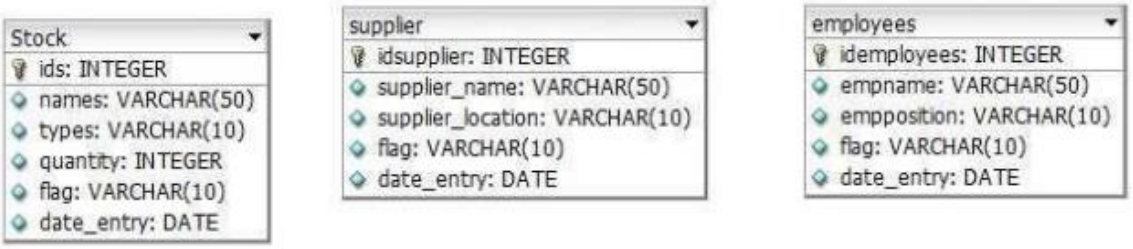


BAGIAN II

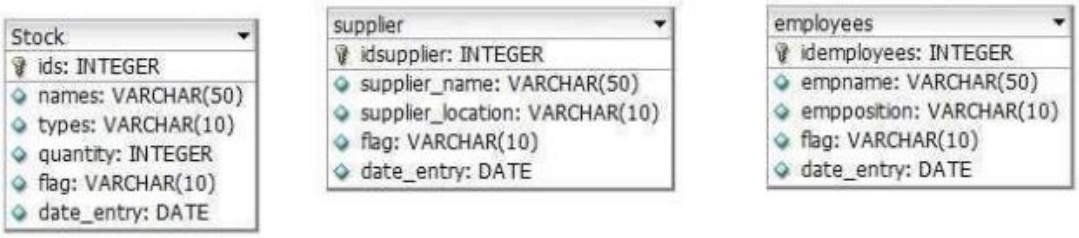


Sesuai dengan scenario gambar diatas :

1. Pada Oracle Server buatlah skema (user) baru dengan nama STORE. Skema ini menyimpan data stock, supplier, dan employees



2. Pada serverr MYSQL (Tokyo) dan Postgre (London), buatlah database baru yang berisi table- tabel berikut ini.



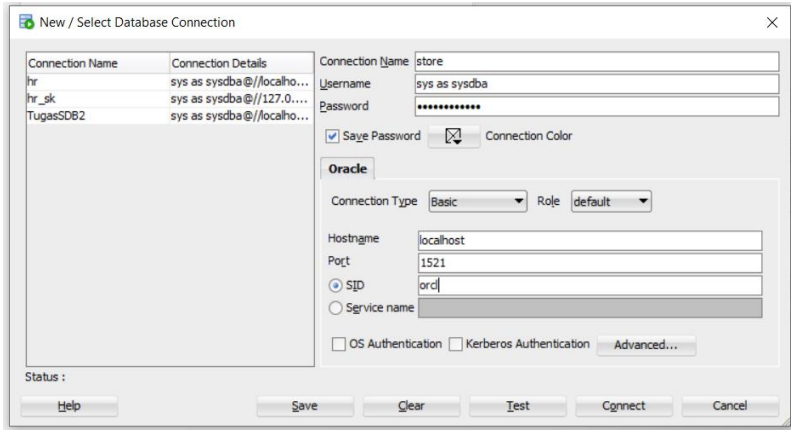
3. Isikan data dummies (simulasi) yang ada pada masing-masing table yang sudah dibuat

4. Buatlah koneksi dengan menggunakan Heterogeneous Service antara database Oracle dengan database MySQL

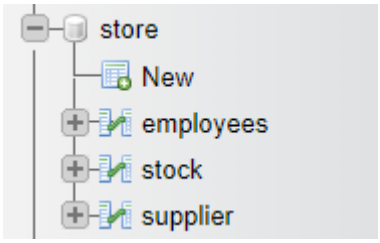
JAWABAN

1. Membuat schema baru pada ORACLE server yaitu schema “store”

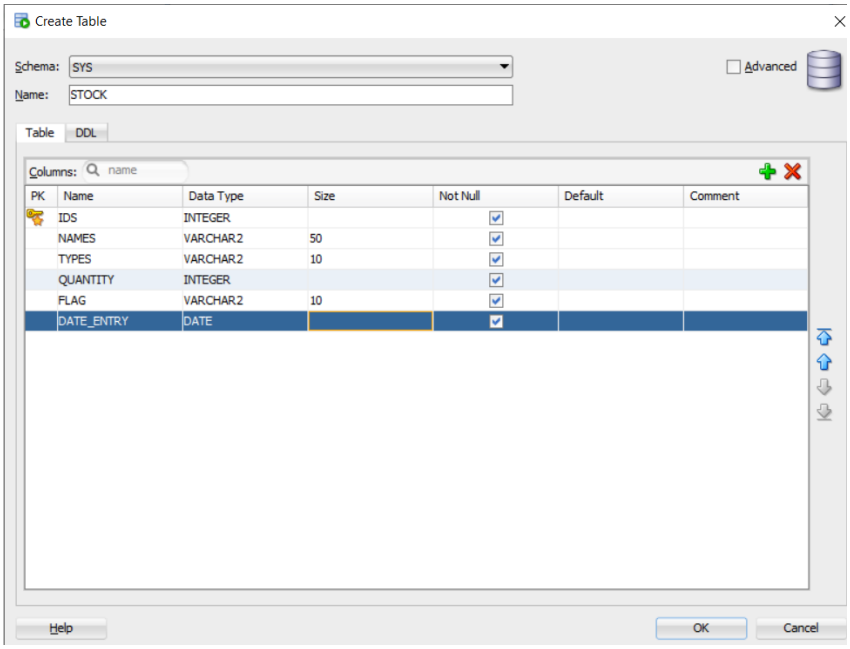
- Membuat Schema baru (store)



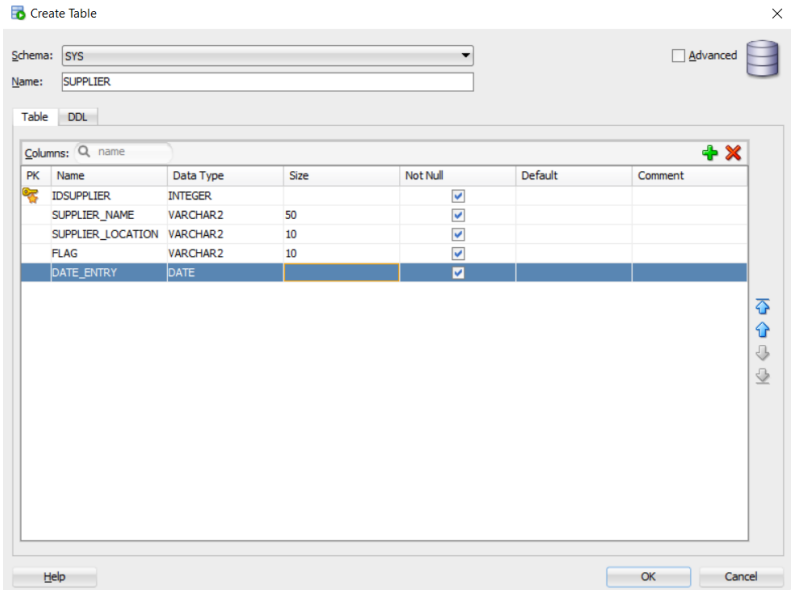
Membuat table



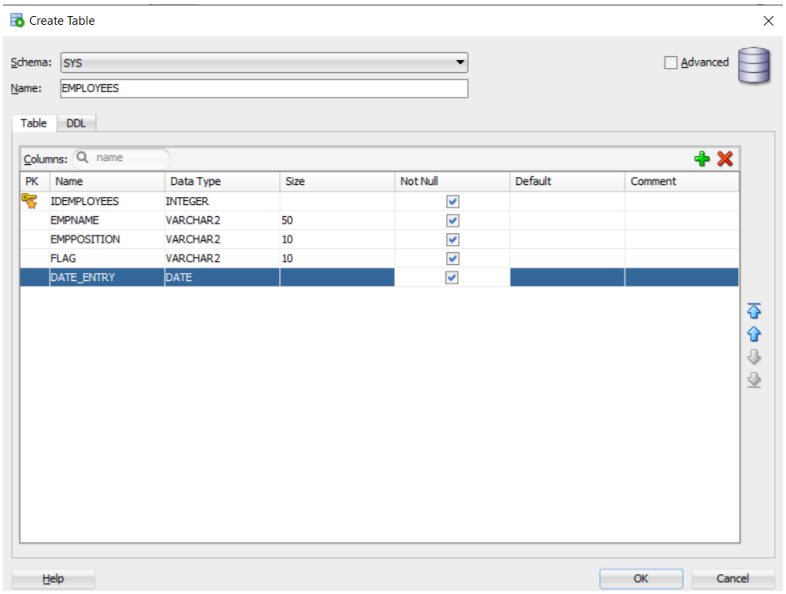
- Membuat table stock



- Membuat table supplier

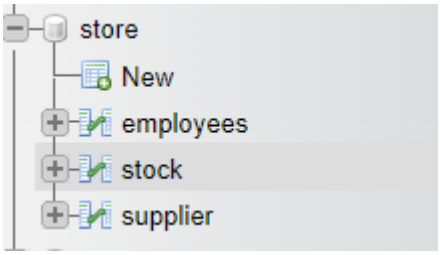


- Membuat table employees



2. Membuat database dan table di mysql dan PostgreSQL

➢ MYSQL



- Table stock

Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idx	int(11)			No	None			Change Drop More
2	names	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	types	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
4	quantity	int(11)			No	None			Change Drop More
5	flag	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
6	date_entry	date			No	None			Change Drop More

- Table supplier

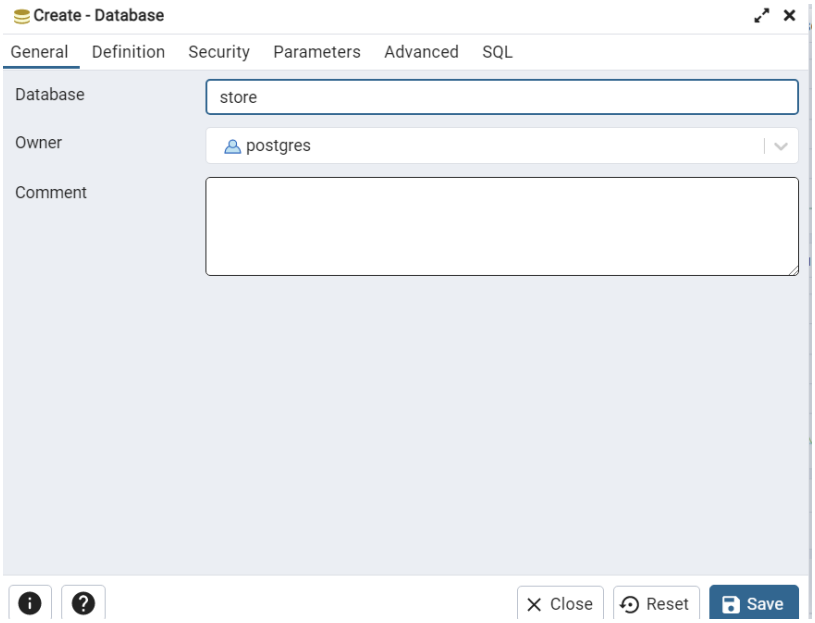
Table structure | Relation view

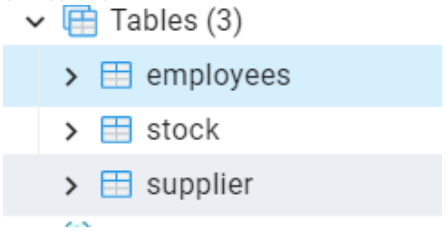
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idspplier	int(11)			No	None			Change Drop More
2	supplier_name	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	supplier_location	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
4	flag	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
5	date_entry	date			No	None			Change Drop More

- Table employees

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idemployees	int(11)			No	None			Change Drop More
2	empname	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	empposition	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
4	flag	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
5	date_entry	date			No	None			Change Drop More

- POSTGRESQL





  - Table stock

Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
✎	🗑	ids	integer			○	●	
✎	🗑	names	character varying	50		●	○	
✎	🗑	types	character varying	10		●	○	
✎	🗑	quantity	integer			●	○	
✎	🗑	flag	character varying	10		●	○	
✎	🗑	date_entry	date			●	○	

  - Table supplier

Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
✎	🗑	idsupplier	integer			●	●	
✎	🗑	supplier_name	character varying	50		●	○	
✎	🗑	supplier_location	character varying	10		●	○	
✎	🗑	flag	character varying	10		●	○	
✎	🗑	date_entry	date			●	○	

- Table employees

Inherited from table(s) | Select to inherit from...

Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
✎	🗑	idemployees	integer			●	●	
✎	🗑	empname	character varying	50		●	○	
✎	🗑	empposition	character varying	10		●	○	
✎	🗑	flag	character varying	10		●	○	
✎	🗑	date_entry	date			●	○	

3. Mengisi data dummies pada masing2 table di masing2 database per jenis Bahasa sql

➢ ORACLE

- Table Stock

```
INSERT INTO stock
VALUES (101, 'Barang A', 'Medium', 3, 'Ind', to_date('2022-11-02', 'yyyy/mm/dd'));
INSERT INTO stock
VALUES ('102', 'Barang B', 'Best', '4', 'Tky', to_date('2019-11-06', 'yyyy/mm/dd'));
```

	IDS	NAMES	TYPES	QUANTITY	FLAG	DATE_ENTRY
1	101	Barang A	Medium	3	Ind	02-11-2022
2	102	Barang B	Best	4	Tky	06-11-2019

- Table Supplier

sheet | Query Builder

```
INSERT INTO supplier VALUES (201, 'Amir', 'Indonesia', 'Ind', to_date('2022-11-01', 'yyyy/mm/dd'));
INSERT INTO supplier VALUES (202, 'Rifqi', 'Tokyo', 'Tky', to_date('2022-11-02', 'yyyy/mm/dd'));
```

	IDSUPPLIER	SUPPLIER_NAME	SUPPLIER_LOCATION	FLAG	DATE_ENTRY
1	201	Amir	Indonesia	Ind	01-11-2022
2	202	Rifqi	Tokyo	Tky	02-11-2022

- Table Employees

```
INSERT INTO employees
VALUES (301, 'Indo Charity', 'Indonesia', 'Ind', to_date('2020-11-11', 'yyyy/mm/dd'));
INSERT INTO employees
VALUES(302, 'Tokyo Charity', 'Tokyo', 'Tky', to_date('2020-11-20', 'yyyy/mm/dd'));
```

	IDEMPLOYEES	EMPNAME	EMPPOSITION	FLAG	DATE_ENTRY
1	301	Indo Charity	Indonesia	Ind	11-11-2020
2	302	Tokyo Charity	Tokyo	Tky	20-11-2020

➢ MYSQL

- Table Stock

	idx	names	types	quantity	flag	date_entry
☐ Edit Copy Delete	101	Barang A	Medium	3	Ind	2022-11-02
☐ Edit Copy Delete	102	Barang B	Best	4	Tky	2019-11-06

- Table Supplier

	idspplier	supplier_name	supplier_location	flag	date_entry
Edit Copy Delete	201	Amir	Indonesia	Ind	2022-11-01
Edit Copy Delete	202	Rifqi	Tokyo	Tky	2022-11-02

- Table Employees

	idemployees	empname	empposition	flag	date_entry
Edit Copy Delete	301	Indo Charity	Indonesia	Ind	2020-11-11
Edit Copy Delete	302	Tokyo Charity	Tokyo	Tky	2020-11-20

➢ POSTGRESQL

- Table Stock

```
INSERT INTO public.stock(
    ids, names, types, quantity, flag, date_entry)
    VALUES (101, 'Barang A', 'Medium', 3, 'Ind', '2022-11-02'),
    (102, 'Barang B', 'Best', 4, 'Tky', '2019-11-06')
```

	ids [PK] integer	names character varying (50)	types character varying (10)	quantity integer	flag character varying (10)	date_entry date
1	101	Barang A	Medium	3	Ind	2022-11-02
2	102	Barang B	Best	4	Tky	2019-11-06

- Table Supplier

```
INSERT INTO public.supplier(
    idsupplier, supplier_name, supplier_location, flag, date_entry)
    VALUES (201, 'Amir', 'Indonesia', 'Ind', '2022-11-01'),
    (202, 'Rifqi', 'Tokyo', 'Tky', '2022-11-02');
```

	idsupplier [PK] integer	supplier_name character varying (50)	supplier_location character varying (10)	flag character varying (10)	date_entry date
1	201	Amir	Indonesia	Ind	2022-11-01
2	202	Rifqi	Tokyo	Tky	2022-11-02

- Table Employees

```
INSERT INTO public.employees(
    idemployees, empname, empposition, flag, date_entry)
    VALUES (301, 'Indo Charity', 'Indonesia', 'Ind', '2020-11-11'),
    (302, 'Tokyo Charity', 'Tokyo', 'Tky', '2020-11-20');
```

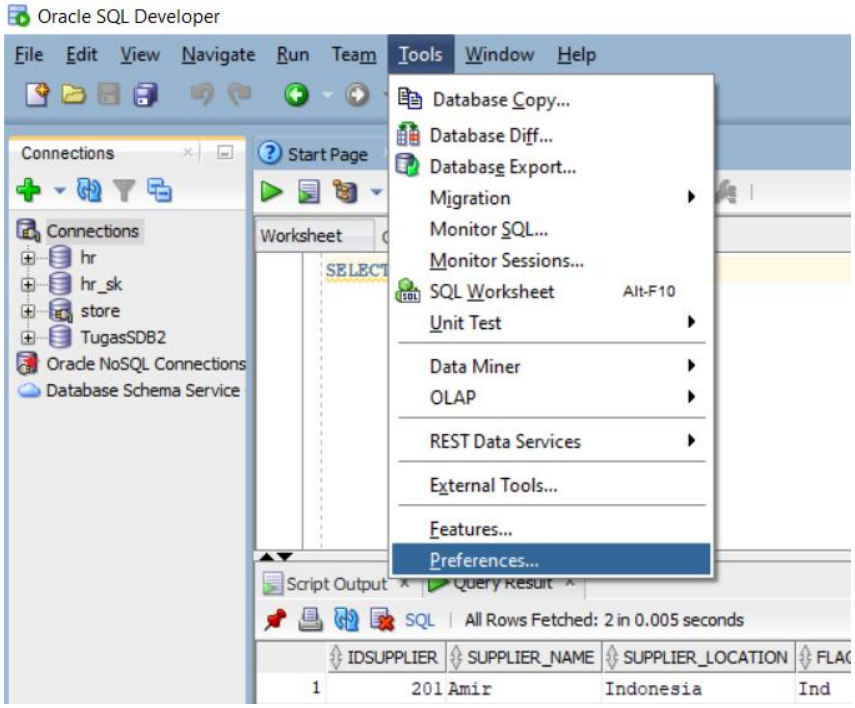
	idemployees [PK] integer	empname character varying (50)	empposition character varying (10)	flag character varying (10)	date_entry date
1	301	Indo Charity	Indonesia	Ind	2020-11-11
2	302	Tokyo Charity	Tokyo	Tky	2020-11-20

4. Buat koneksi oracle dan mysql

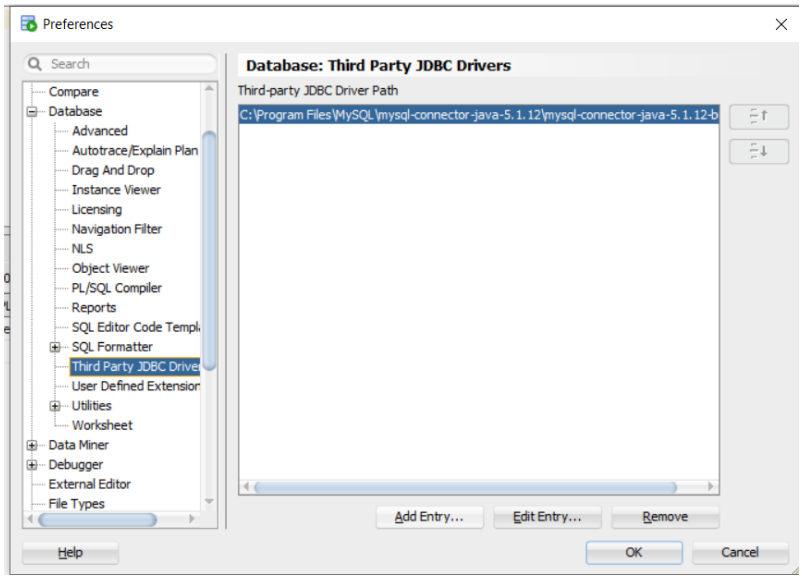
➢ Download software mysql connector. Di sini kami menggunakan mysql-connector-java
http://download.nust.na/pub6/mysql/downloads/connector/j/index.html

docs	06/11/2022 19:38	File folder	
src	06/11/2022 19:38	File folder	
build	18/02/2010 6:16	XML Document	44 KB
CHANGES	18/02/2010 6:16	File	181 KB
COPYING	18/02/2010 6:16	File	19 KB
EXCEPTIONS-CONNECTOR-J	18/02/2010 6:16	File	6 KB
mysql-connector-java-5.1.12-bin	18/02/2010 6:16	Executable Jar File	716 KB
README	18/02/2010 6:16	File	97 KB
README	18/02/2010 6:16	Text Document	100 KB

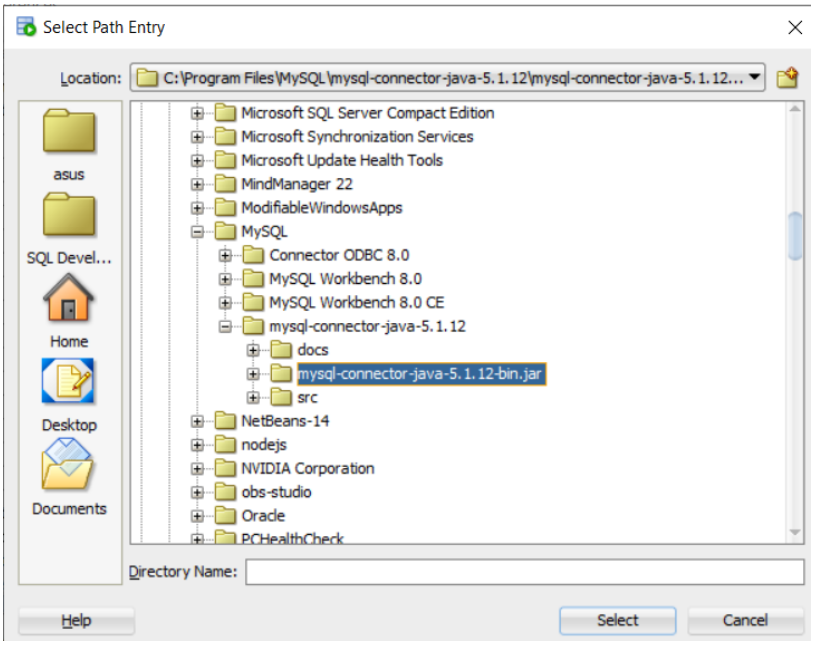
- Buka tools > Preferences di sql developer



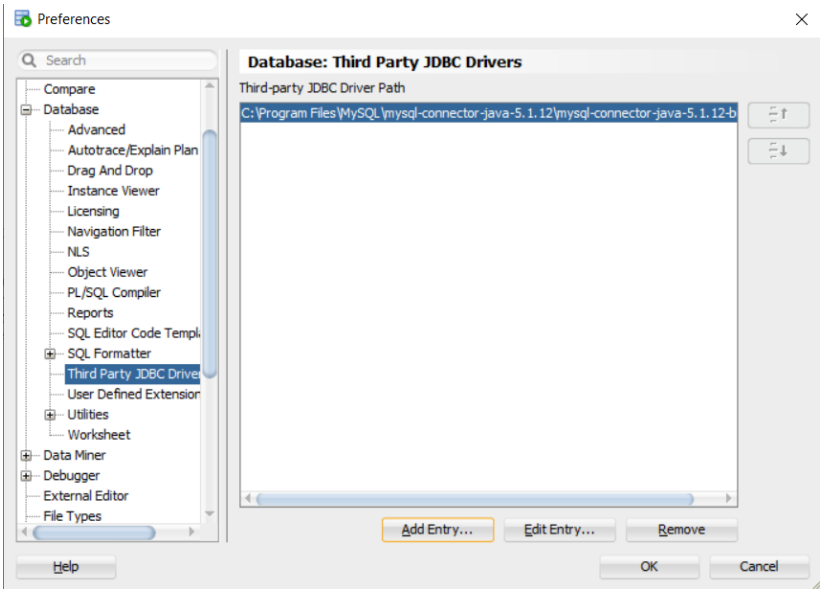
- Klik third party JDBC > add entry (Konektor dengan sql lain)



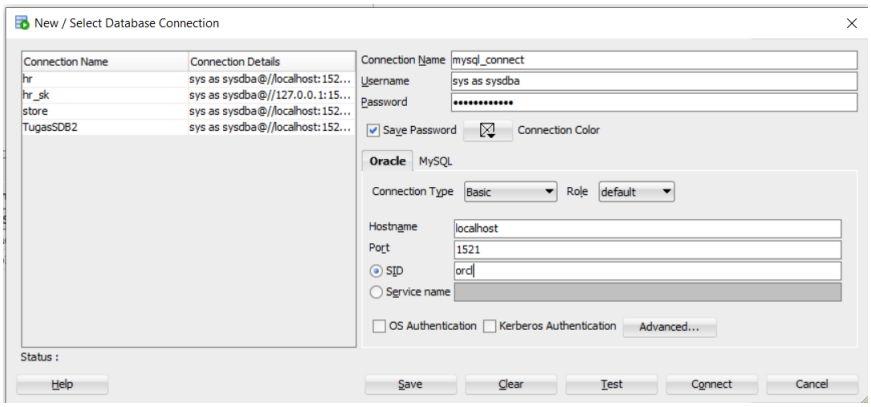
- Select software tempat konektor (mysql-connector-java.bin.jar) berada



- Klik ok



- Buat koneksi oracle dengan mysql



Koneksi oracle dengan mysql berhasil dibuat!