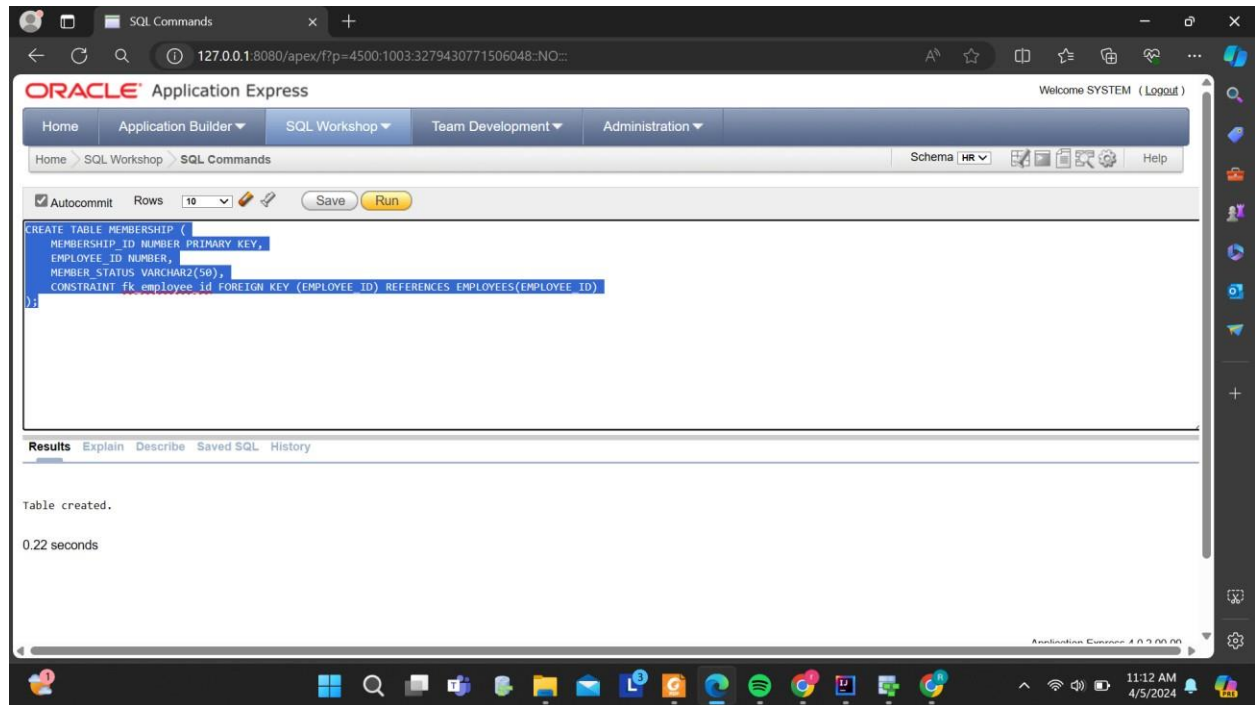
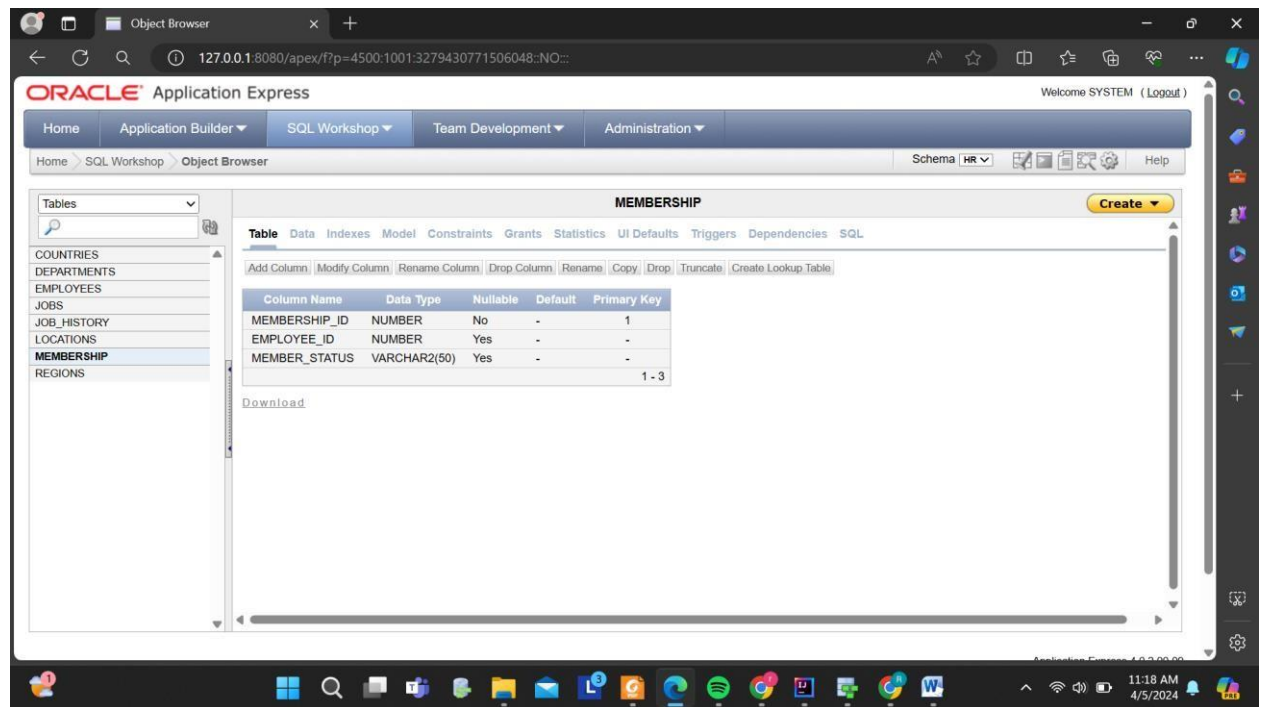


NAMA : MUHAMMAD CAEZA ADRIAWAN EFFENDI
NIM : 202210370311386
TUGAS : CODELAB MOD 4
IMPLEMENTASI DDL PADA SKEMA HR

MEMBUAT TABEL





CREATE TABLE MEMBERSHIP: Ini adalah perintah yang digunakan untuk membuat tabel baru dalam sebuah basis data. Kata kunci CREATE TABLE diikuti oleh nama tabel yang ingin Anda buat, yaitu MEMBERSHIP.

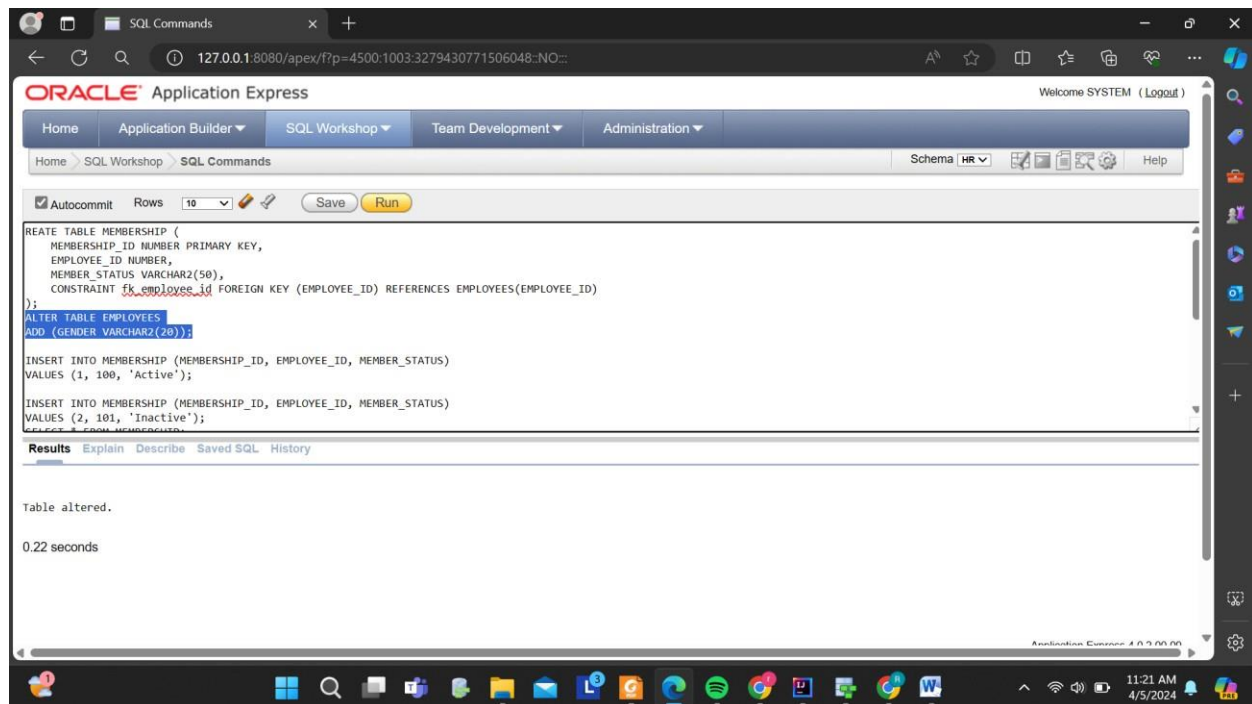
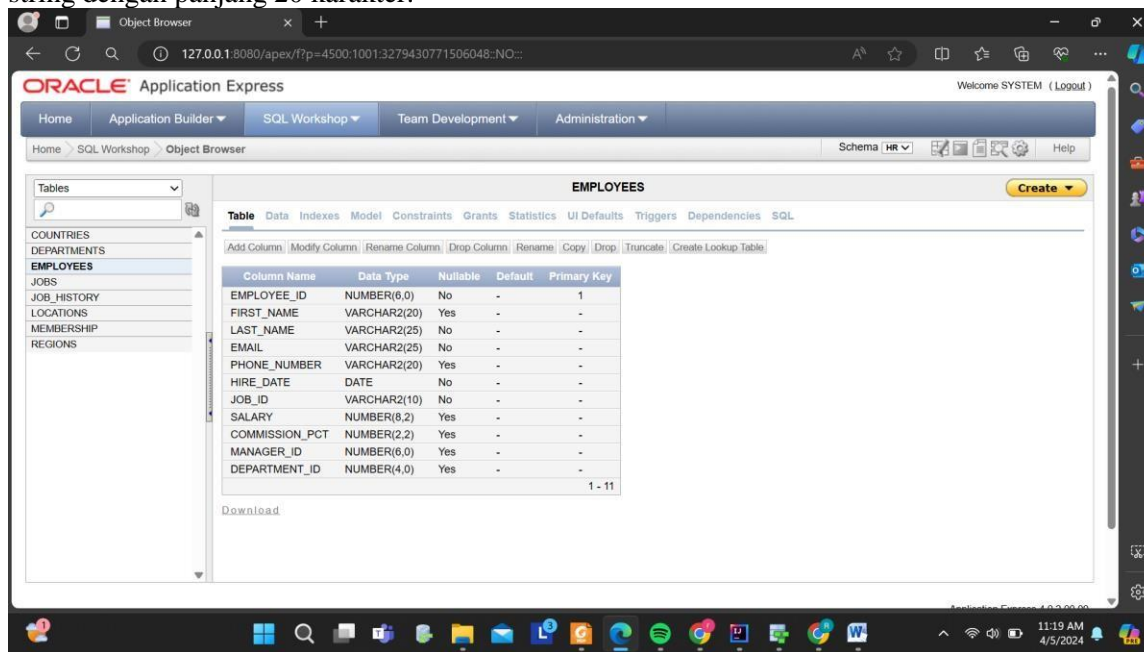
(MEMBERSHIP_ID NUMBER PRIMARY KEY, EMPLOYEE_ID NUMBER, MEMBER_STATUS VARCHAR(50)): Bagian ini mendefinisikan struktur kolom untuk tabel MEMBERSHIP. Setiap kolom didefinisikan dengan nama kolom diikuti oleh tipe data kolom. Di sini, kita mendefinisikan tiga kolom: MEMBERSHIP_ID dengan tipe data NUMBER, yang juga ditandai sebagai kunci utama (PRIMARY KEY). Ini berarti nilai di kolom ini harus unik untuk setiap baris, dan ini adalah kolom yang akan digunakan untuk mengidentifikasi setiap baris secara unik dalam tabel.

EMPLOYEE_ID dengan tipe data NUMBER, yang akan menyimpan ID karyawan yang terkait dengan keanggotaan.

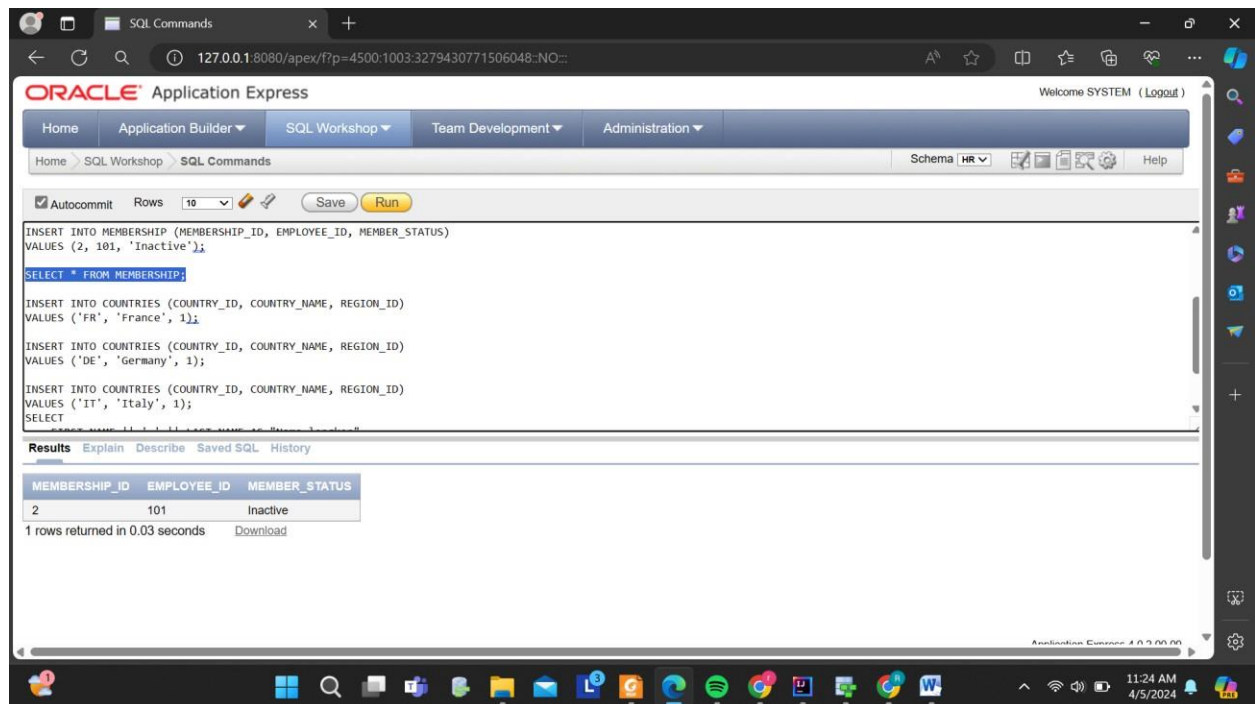
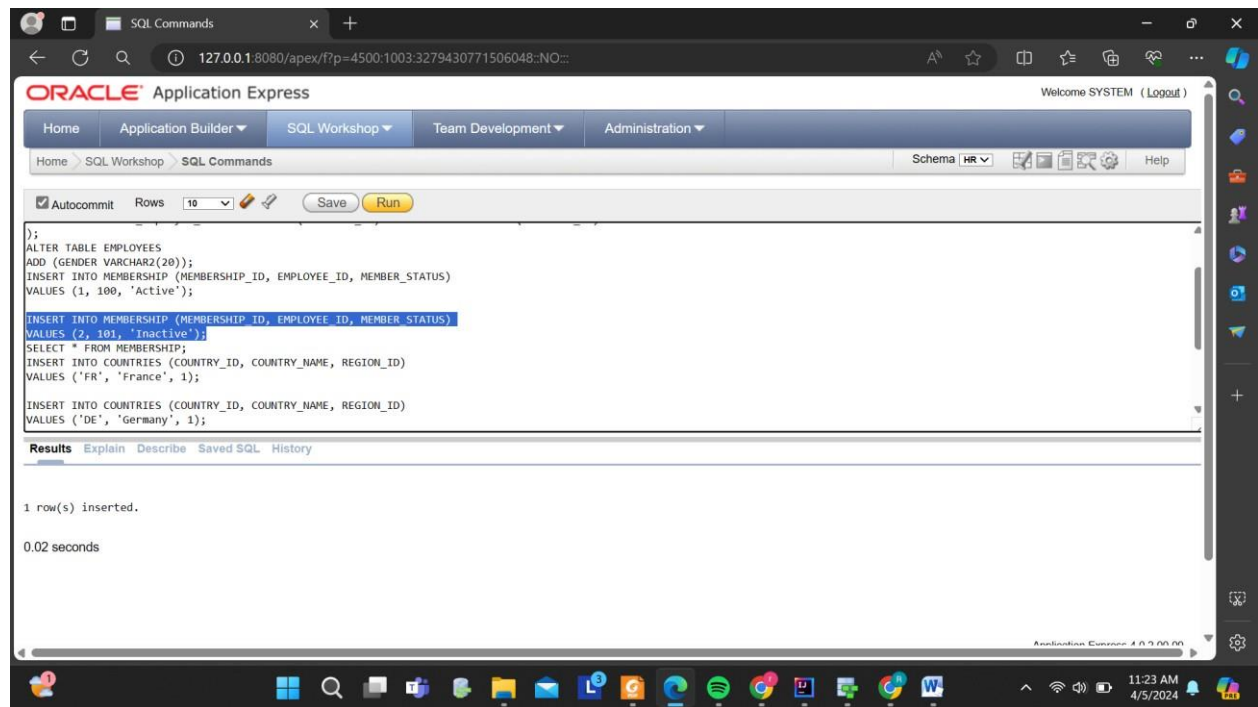
MEMBER_STATUS dengan tipe data VARCHAR(50), yang akan menyimpan status keanggotaan dengan tipe data string dan panjang maksimum 50 karakter.

CONSTRAINT FK_EMPLOYEE_ID FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEES (EMPLOYEE_ID): Ini adalah definisi dari kunci asing (FOREIGN KEY) dalam tabel. Ini menetapkan batasan integritas referensial antara kolom EMPLOYEE_ID dalam tabel MEMBERSHIP dan kolom EMPLOYEE_ID dalam tabel EMPLOYEES. Ini memastikan bahwa setiap nilai dalam kolom EMPLOYEE_ID di tabel MEMBERSHIP harus ada di kolom EMPLOYEE_ID di tabel EMPLOYEES.

Ubah struktur tabel EMPLOYEES dengan menambahkan kolom baru bernama GENDER yang bertipe data string dengan panjang 20 karakter.



IMPLEMENTASI DML PADA SKEMA HR
MENAMBAHKAN DATA PADA TABEL MEMBERSHIP

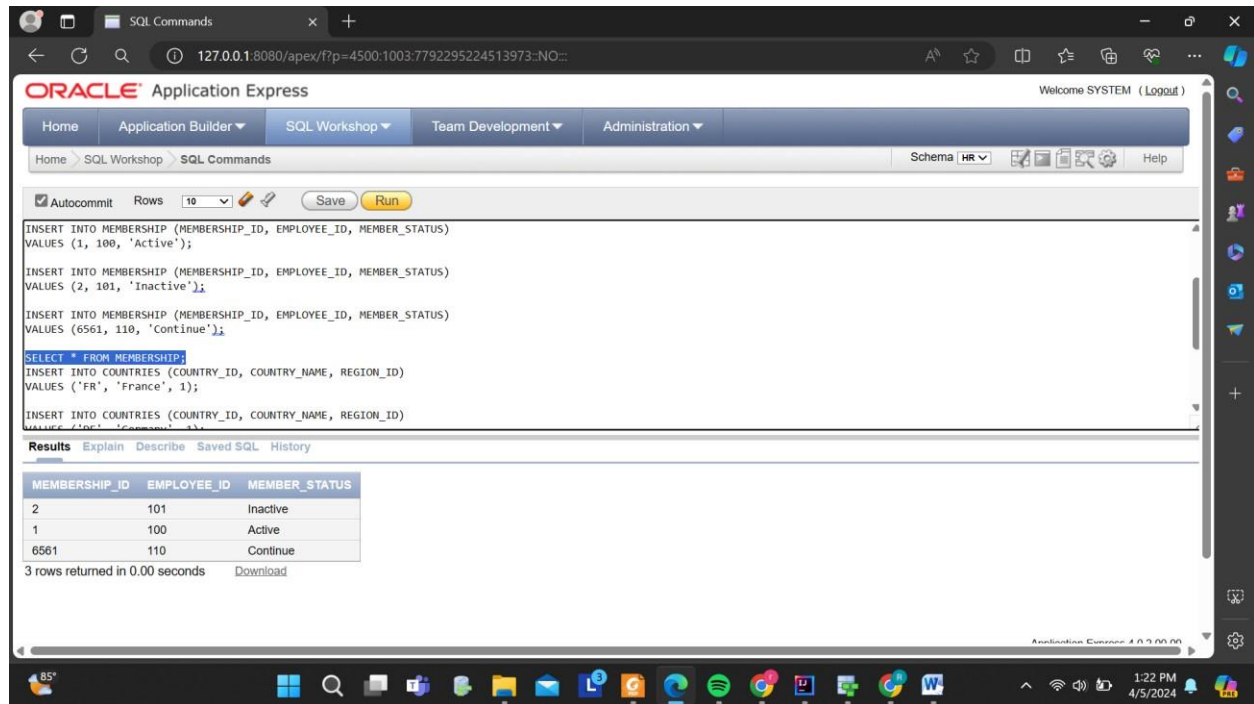


INSERT INTO MEMBERSHIP: Ini adalah perintah untuk memasukkan data kedalam tabel MEMBERSHIP.

(MEMBERSHIP_ID, EMPLOYEE_ID, MEMBER_STATUS): Bagian ini menentukan kolom mana yang akan diisi dengan nilai yang diberikan dalam pernyataan VALUES.

VALUES (2, 101, 'INACTIVE')': Bagian ini menyediakan nilai yang akan dimasukkan ke dalam kolom yang telah ditentukan sebelumnya. Nilai yang diberikan adalah:
2 untuk kolom MEMBERSHIP_ID. 101 untuk kolom EMPLOYEE_ID.
'INACTIVE' untuk kolom MEMBER_STATUS

Tampilkan Data yang baru saja anda tambahkan pada kolom MEMBERSHIP



The screenshot shows the Oracle Application Express SQL Workshop interface. The SQL Commands window contains the following SQL statements:

```
INSERT INTO MEMBERSHIP (MEMBERSHIP_ID, EMPLOYEE_ID, MEMBER_STATUS)
VALUES (1, 100, 'Active');

INSERT INTO MEMBERSHIP (MEMBERSHIP_ID, EMPLOYEE_ID, MEMBER_STATUS)
VALUES (2, 101, 'Inactive');

INSERT INTO MEMBERSHIP (MEMBERSHIP_ID, EMPLOYEE_ID, MEMBER_STATUS)
VALUES (6561, 110, 'Continue');

SELECT * FROM MEMBERSHIP;

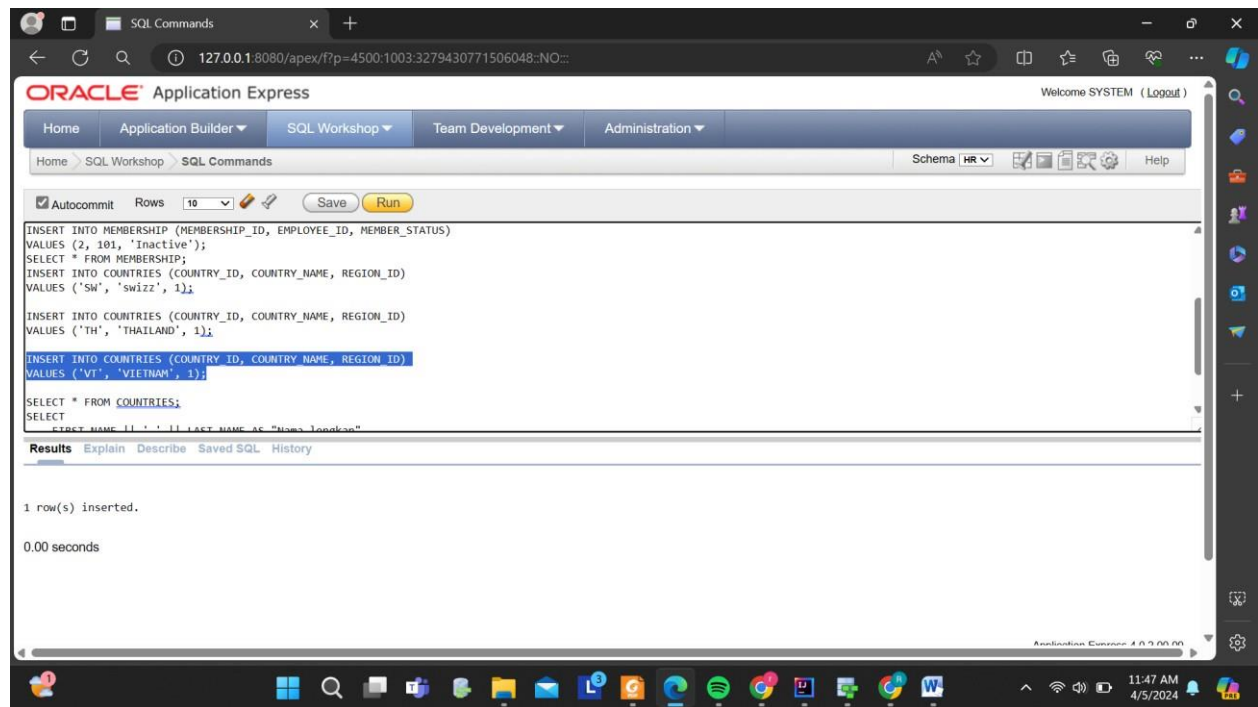
INSERT INTO COUNTRIES (COUNTRY_ID, COUNTRY_NAME, REGION_ID)
VALUES ('FR', 'France', 1);

INSERT INTO COUNTRIES (COUNTRY_ID, COUNTRY_NAME, REGION_ID)
VALUES ('FR', 'France', 1);
```

The Results window displays the output of the SELECT statement, showing 3 rows returned in 0.00 seconds:

MEMBERSHIP_ID	EMPLOYEE_ID	MEMBER_STATUS
2	101	Inactive
1	100	Active
6561	110	Continue

Tambahkan data negara pada tabel COUNTRIES sebanyak 3 data sesuaikan dengan region negaranya.



NG	Nigeria	4
NL	Netherlands	1
SG	Singapore	3
SW	swizz	1
TH	THAILAND	1
UK	United Kingdom	1
US	United States of America	2
VT	VIETNAM	1
ZM	Zambia	4
ZW	Zimbabwe	4

28 rows returned in 0.00 seconds [Download](#)

HASIL dari penambahan 3 negara.

Tampilkan Nama lengkap, email, nomor handphone, dan gaji berikan alias pada masing - masing kolom agar terlihat rapi (hint: **AS**)

SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3279430771506048::NO...

```

SELECT * FROM COUNTRIES;
SELECT
  FIRST_NAME || ' ' || LAST_NAME AS "Nama lengkap",
  EMAIL AS "Email",
  PHONE_NUMBER AS "Nomor handphone",
  SALARY AS "Gaji"
FROM EMPLOYEES;

```

Results Explain Describe Saved SQL History

Nama lengkap	Email	Nomor handphone	Gaji
Steven King	SKING	515.123.4567	24000
Neena Kochhar	NKOCHHAR	515.123.4568	17000
Lex De Haan	LDEHAAN	515.123.4569	17000
Alexander Hunold	AHUNOLD	590.423.4567	9000
Bruce Ernst	BERNST	590.423.4568	6000
David Austin	DAUSTIN	590.423.4569	4800
Valli Pataballa	VPATABAL	590.423.4560	4800
Diana Lorentz	DLORENTZ	590.423.5567	4200
Nancy Greenberg	NGREENBE	515.124.4569	12008
Daniel Faviet	DFAVIET	515.124.4169	9000
John Chen	JCHEN	515.124.4269	8200
Ismail Sciarra	ISCIARRA	515.124.4369	7700
Jose Manuel Urman	JMURMAN	515.124.4469	7800
Luis Popp	LPOPP	515.124.4567	6900
Den Raphaely	DRAPHEAL	515.127.4561	11000
Alexander Khoo	AKHOO	515.127.4562	3100
Shelli Baida	SBAIDA	515.127.4563	2900
Sigal Tobias	STOBIAS	515.127.4564	2800
Guy Himuro	GHIMURO	515.127.4565	2600

SELECT: Ini adalah kata kunci yang digunakan untuk memilih kolom atau ekspresi yang ingin ditampilkan dalam hasil kueri.

FIRST_NAME || ' ' || LAST_NAME AS Nama_Lengkap: Ini adalah ekspresi yang menggabungkan nilai kolom FIRST_NAME dan LAST_NAME dari setiap baris dalam tabel EMPLOYEES. Operator || digunakan untuk menggabungkan nilai dalam SQL, dan tanda kutip tunggal ' ' digunakan untuk menyisipkan spasi di antara nama pertama dan nama terakhir. Frasa AS Nama_Lengkap memberikan alias untuk hasil penggabungan, sehingga hasilnya akan ditampilkan dengan nama kolom yang lebih deskriptif.

EMAIL AS Email: Ini adalah contoh penggunaan alias di mana kolom EMAIL dari tabel EMPLOYEES diubah namanya menjadi Email. Penggunaan alias membantu untuk memberikan nama yang lebih deskriptif atau lebih mudah dimengerti pada hasil kueri.

PHONE_NUMBER AS Nomor Handphone: Ini adalah contoh lain penggunaan alias di mana kolom PHONE_NUMBER dari tabel EMPLOYEES diubah namanya menjadi Nomor_Handphone. Sama seperti sebelumnya, penggunaan alias membantu meningkatkan kejelasan hasil kueri.

SALARY AS Gaji: Ini adalah contoh lain penggunaan alias di mana kolom SALARY dari tabel EMPLOYEES diubah namanya menjadi Gaji. Penggunaan alias juga dapat mempermudah dalam pemrosesan hasil kueri.