LAPORAN PRAKTIKUM
PENGENALAN BASIS DATA

Proyek Merancang, membangun dan mengelola basisdata di MYSQL

Wina Sorta Maria Gea
13323009
DIII Teknologi Komputer

INSTITUT TEKNOLOGI DEL
FAKULTAS VOKASI

Proyek Merancang, Membangun dan Mengelola Basisdata di MYSQL

Tujuan Proyek

Tujuan proyek ini adalah membuat database yang kompatibel dengan MySQL untuk mengelola informasi pekerjaan di bidang Teknologi Komputer. Database ini akan menyederhanakan pengelolaan dan analisis data, serta mempercepat pengambilan keputusan dalam proses penginformasian.

Deskripsi Proyek

Proyek ini bertujuan untuk menciptakan sebuah situs web informasi pekerjaan di bidang Teknologi Komputer yang menyajikan berbagai jenis pekerjaan terkait, panduan karir, dan pengembangan diri. Situs web ini dirancang untuk memberikan kemudahan kepada pengunjung dalam mencari informasi pekerjaan untuk lulusan di bidang teknologi komputer.

Melalui situs web ini, pengunjung dapat dengan mudah menemukan informasi tentang berbagai jenis pekerjaan yang tersedia. Informasi tersebut dapat diakses melalui media promosi seperti pamflet, brosur, iklan, atau langsung di situs web. Pengunjung dapat melihat daftar pekerjaan, deskripsi, persyaratan, dan foto-foto yang mengilustrasikan pekerjaan tersebut.

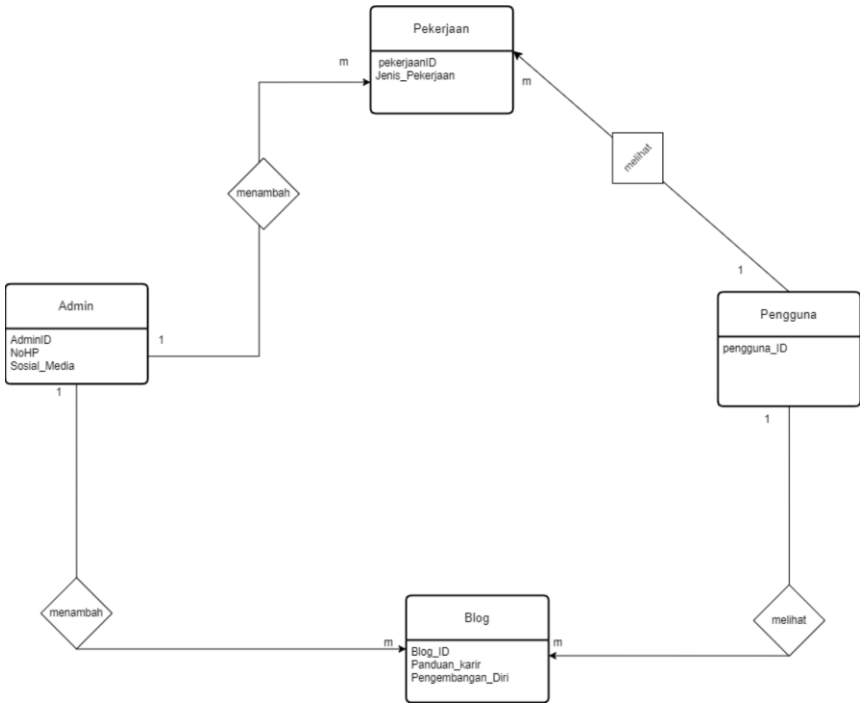
Situs web ini juga menyediakan fitur jenis pekerjaan. Pengunjung dapat melohat informasi mengenai pekerjaan yang cocok dan sesuai bagi lulusan Teknologi komputer web disediakan. Mereka dapat memilih jenis pekerjaan atau layanan yang diminati serta melihat panduan dalam karir dan pengembangan diri yang baik.

Diharapkan dengan adanya situs web ini, pengunjung dapat dengan mudah mencari dan mengetahiu iiniformasi pekerjaan di bagian teknologi komputer. Hal ini akan membantu pengunjung dalam pencarian informasi, sehingga mereka dapat lebih fokus pada persiapan karir di bidang Teknologi Komputer..

Tugas

1. Buat rancangan basisdata (buat ERD dan lakukan translasi ERD menjadi table-table yang sesuai)

ERD



2. Lakukan validasi terhadap rancangan anda dengan konsep normalisasi

Tabel admin

id_Admin	No_Telp	Email_Admin	SosIal_Media_Admin
1	083423525	winagea22@gmail.com	wina_gea
2	0834235258	bastian05@gmail.com	bastian.sitanggang
3	08342355	endrogel122@gmail.com	endrogelsibarani
4	083423525	ramhotsagala@gmail.com	ramhot_sgl

3NF

Tabel Kontak

Id_Admin	NO_Telp
1	083423525
2	0834235258
3	08342355
4	083423525

Tabel Email

Id_Admin	Email_Admin
1	winagea22@gmail.com
2	bastian05@gmail.com
3	endrogel122@gmail.com
4	ramhotsagala@gmail.com

Tabel Sosial_Media_Admin

Id_Admin	Sosial_Media_Admin
1	wina_gea
2	bastian.sitanggang
3	endrogelsibarani
4	ramhot_sgl

Tabel Pengguna

id_Pengguna	Nama_Pengguna	Email_Pengguna
1	Stevi Sianipar	Stevi01@gmail.com
2	Rahel Napitupulu	Rahel23@gmail.com
3	Naomi Butar Butar	Naomi29@gmail.com
4	Dealova Zevanya	Dealova11@gmail.com
5	Romaito Silaen	Romaito17@gmail.com

6	Lely Butar Butar	Lely99@gmail.com
7	Enjelita Sitorus	Enjelita19@gmail.com
8	Tanisha Aritonang	Tanisha16@gmail.com
9	Agita Panjaitan	Agitaa@gmail.com
10	Sofia Sianipar	SofiaS@gmail.com

2NF

Tabel Pengguna

Id_Pengguna	Nama_Pengguna
1	Stevi Sianipar
2	Rahel Napitupulu
3	Naomi Butar Butar
4	Dealova Zevanya
5	Romaito Silaen
6	Lely Butar Butar
7	Enjelita Sitorus
8	Tanisha Aritonang
9	Agita Panjaitan
10	Sofia Sianipar

3. Lakukan pengembangan basisdata sesuai rancangan final (menerapkan Data Definition Language pada SQL)

a. Pembuatan basisdata beserta tabel-tabelnya

b. Enforcing Data Integrity dengan menerapkan jenis-jenis constraint yang sudah anda pelajari. Jawaban a dan b :

```
CREATE DATABASE proyek_pbd
USE proyek_pbd
CREATE TABLE Admin (
   id_Admin INT PRIMARY KEY,
   NO_Telp VARCHAR(25),
   Email_Admin VARCHAR(225),
   Sosial_Media_Admin VARCHAR(225)
);
ALTER TABLE Admin ADD COLUMN Pekerjaan VARCHAR(225);
ALTER TABLE Admin ADD COLUMN Blog VARCHAR(225);
CREATE TABLE Pekerjaan (
   id_Pekerjaan INT PRIMARY KEY,
   Jenis_Pekerjaan VARCHAR(225)
);

ALTER TABLE Pekerjaan ADD COLUMN id_Admin INT;
ALTER TABLE Pekerjaan ADD FOREIGN KEY (id_Admin) REFERENCES
Admin(id_Admin);

CREATE TABLE Blog (
   id_Blog INT PRIMARY KEY,
   Panduan_Karir VARCHAR(225),
   pengembangan_diri VARCHAR(225),
   id_Admin INT,
   FOREIGN KEY (id_Admin) REFERENCES Admin(id_Admin)
);

CREATE TABLE Pengguna (
   id_pengguna INT PRIMARY KEY,
```

Nama_Pengguna VARCHAR(225),

Email_Pengguna VARCHAR(225)

);

ALTER TABLE Pengguna

4. Lakukan aktifitas pengelolaan basisdata (menerapkan Data Manipulation Language pada SQL)

a) Lakukan pengisian data dummy pada basisdata yang anda kembangkan

INSERT INTO Admin(id_Admin, NO_Telp, Email_Admin, Sosial_Media_Admin)

VALUES

(1, '0812345678', 'winagea22@gmail.com', 'wina_gea'),

(2, '0832459032', 'bastiansitanggang05@gmail.com', 'bastian.sitanggang'),

(3, '0893245678', 'endrogel11@gmail.com', 'endrogel_sibaranii');

INSERT INTO Pekerjaan(id_Pekerjaan,Jenis_Pekerjaan)

VALUE

(1,'Network Administrator'),

(2,'Network Engineer'),

(3,'Network Architect'),

(4,'Database Administrator'),

(5,'Computer Systems Analyst'),

(6,'Computer and Information System'),

(7,'Network Technician'),

(8,'Embedded Systems Engineer'),

(9,'Embedded Electronics Engineer'),

(10,'Control Systems Engineer'),

(11,'Embedded Network Engineer'),

(12,'Cybersecurity Embedded Sytem Developer'),

(13,'Cloud Engineer'),

(14,'Cloud Administrator'),

(15,'Security Engineer'),

(16,'Cloud Architect'),

(17,'Data Engineer'),

(18,'Data Scientist'),

(19,'Backup and Recovery'),

(20,'System Administrator'),

(21,'Fronted Developer'),

(22,'Software Developer'),

(23,'Database Managemnet'),

(24,'UI Developer'),

(25,'Back-end-Developer');

INSERT INTO Blog(id_Blog,Panduan_karir,Pengembangan_Diri)

VALUE

(1,'C/V Resume','Cara Pengembangan Diri Untuk Menggali Potensi Diri'),

(2,'Cover Letter','Growth Mindset: Arti, Penerapan dan Manfaatnya dalam Karir'),

(3,'Portofolio & Personal Branding','Refeleksi Diri: Manfaat, Pengertian, Kepentingannnya, dan Caranya'),

(4,'Skkill Interview','Kenali 5 Ciri Ciri Burnout dan Cara Mengatasinya!'),

(5,'Tips Cari Kerja','Mau Sukses di Uaia Muda? Tiru 10 Kebiasaan Ini!');

INSERT INTO Pengguna(id_Pengguna,Nama_Pengguna,Email_Pengguna)

VALUE

(1,'Stevi Sianipar','Stevi01@gmail.com'),

(2,'Rahel Napitupulu','Rahel23@gmail.com'),

(3,'Naomi Butar Butar','Naomi29@gmail.com'),

(4,'Dealova Zevanya','Dealova11@gmail.com'),

(5,'Romaito Silaen','Romaito17@gmail.com'),

(6,'Lely Butar Butar','Lely99@gmail.com'),

(7,'Enjelita Sitorus','Enjelita19@gmail.com'),

(8,'Tanisha Aritonang','Tanisha16@gmail.com'),

(9,'Agita Panjaitan','Agitaa@gmail.com'),

(10,'Sofia Sianipar','SofiaS@gmail.com'),

(11,'Dwi Nainggolan','DwiNainggolan@gmail.com'),

(12,'Calista Panjaitan','Calistaa@gmail.com'),

(13,'Rizki Manullang','Rizkii@gmail.com'),

(14,'Chesia Silaen','Silaen24@gmail.com'),

(15,'Maylani Sagala','May98@gmail.com'),

(16,'Grace Yosevha','Yosevha43@gmail.com'),

(17,'Grace Nainggolan','GraceNainggolan@gmial.com'),

(18,'Samuel Manik','SamuelManik188@gmail.com'),

(19,'Rezeki','Rezekii96@gmail.com'),

(20,'Panji Aritonang,','PanjiAritonangg76@gmial.com'),

(21,'Ripandy Saragih','Ripandyy@gmial.com'),

(22,'Pahala Tambunan','PahalaTambunaann@gmail.com'),

(23,'Arizona Damanik','ArizonaaD14@gmial.com'),

(24,'Yehezkiel Rumapea','Yehezkielll65@gmail.com'),

(25,'Arga Bona','ArgaBona45@gmail.com'),

(26,'Immanuel Manurung','ImmaanuelManurung@gmail.com'),

(27,'Alfred Manurung','Alfred35@gmaiaal.com'),

(28,'Jeysen Siallagan','JeysennSlgn09@gmial.com'),

(29,'Glen Situmorang','GlenSitumorangg31@gmail.com'),

(30,'Samuel Pangaribuan','SamuellPangrib@gmail.com'),

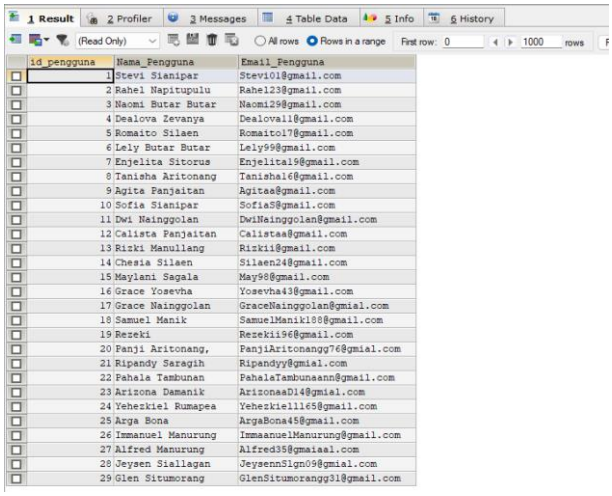
(31,'Gramaldy','Gramaldy221@gmail.com'),

(32,'Handika Pratama','Handikapratama61@gmail.com'),

(33,'Janter Hugo Purba','Janter55@gmail.com');

b). Buat 10 query yang menampilkan data dari 1 table

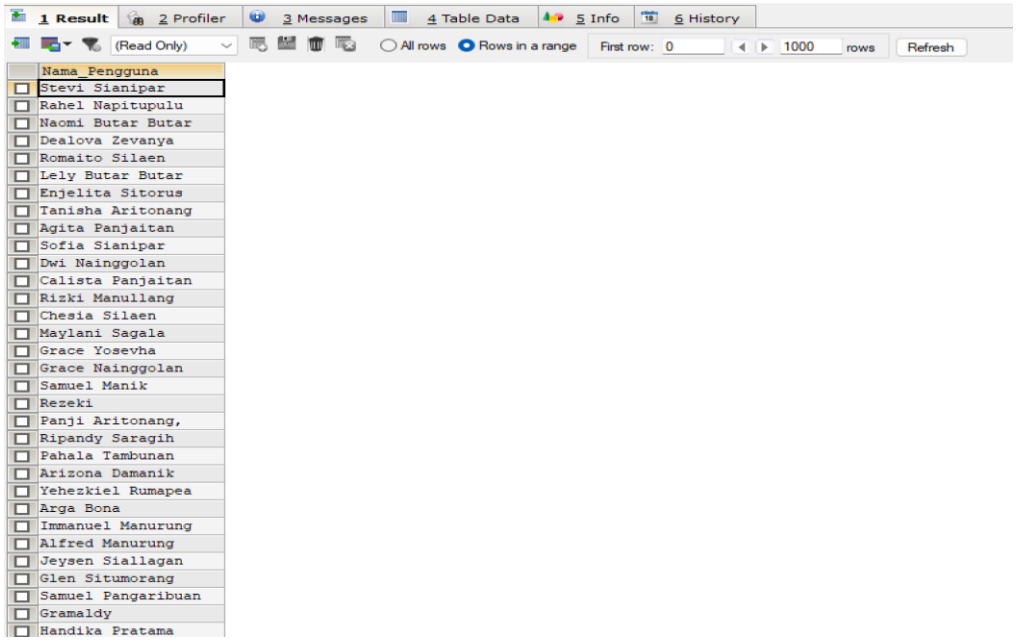
- **SELECT * FROM Pengguna;**



id_pengguna	Nama_Pengguna	Email_Pengguna
1	Stevi Sianipar	Stevi01@gmail.com
2	Rahel Napitupulu	Rahel23@gmail.com
3	Naomi Butar Butar	Naomi29@gmail.com
4	Dealova Zevanya	Dealova11@gmail.com
5	Romaito Silaen	Romaito17@gmail.com
6	Lely Butar Butar	Lely99@gmail.com
7	Enjelita Sitorus	Enjelita19@gmail.com
8	Tanisha Aritonang	Tanisha16@gmail.com
9	Agita Panjaitan	Agitaa@gmail.com
10	Sofia Sianipar	SofiaS@gmail.com
11	Dwi Nainggolan	DwiNainggolan@gmail.com
12	Calista Panjaitan	Calistaa@gmail.com
13	Rizki Manullang	Rizkii@gmail.com
14	Chesia Silaen	Silaen24@gmail.com
15	Maylani Sagala	May98@gmail.com
16	Grace Yosevha	Yosevha43@gmail.com
17	Grace Nainggolan	GraceNainggolan@gmial.com
18	Samuel Manik	SamuelManik188@gmail.com
19	Rezeki	Rezekii96@gmail.com
20	Panji Aritonang,	PanjiAritonangg76@gmial.com
21	Ripandy Saragih	Ripandyy@gmial.com
22	Pahala Tambunan	PahalaTambunaann@gmail.com
23	Arizona Damanik	ArizonaaD14@gmial.com
24	Yehezkiel Rumapea	Yehezkiell165@gmail.com
25	Arga Bona	ArgaBona45@gmail.com
26	Immanuel Manurung	ImmaanuelManurung@gmail.com
27	Alfred Manurung	Alfred35@gmaiaal.com
28	Jeysen Siallagan	JeysennSlgn09@gmial.com
29	Glen Situmorang	GlenSitumorangg31@gmail.com

Syntax SELECT * FROM Pengguna adalah sebuah perintah SQL yang digunakan untuk mengambil atau menampilkan seluruh data yang tersimpan dalam tabel "Pengguna".

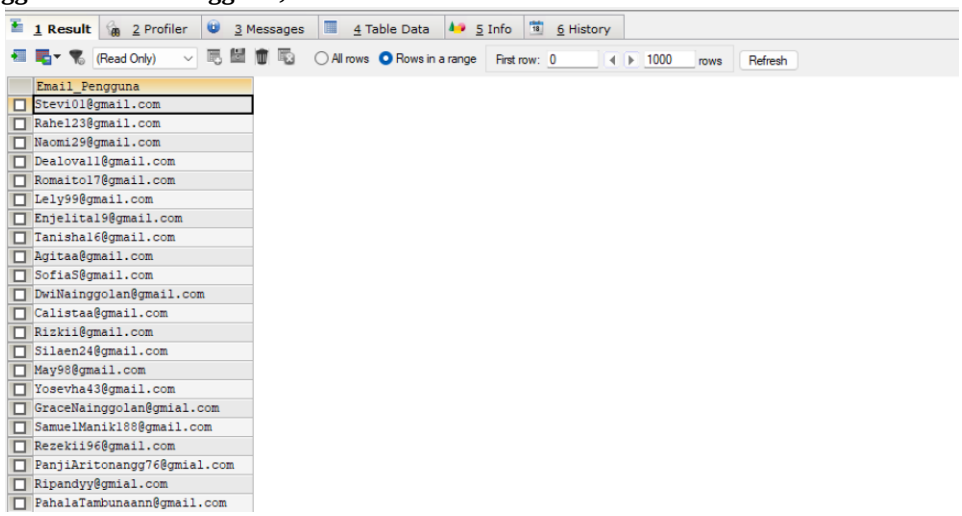
- **SELECT Nama_Pengguna FROM Pengguna;**



Nama_Pengguna
Stevi Sianipar
Rahel Napitupulu
Naomi Butar Butar
Dealova Zevanya
Romaito Silaen
Lely Butar Butar
Enjelita Sitorus
Tanisha Aritonang
Agita Panjaitan
Sofia Sianipar
Dwi Nainggolan
Calista Panjaitan
Rizki Manullang
Chesia Silaen
Maylani Sagala
Grace Yosevha
Grace Nainggolan
Samuel Manik
Rezeki
Panji Aritonang,
Ripandy Saragih
Pahala Tambunan
Arizona Damanik
Yehezkiel Rumapea
Arga Bona
Immanuel Manurung
Alfred Manurung
Jeysen Siallagan
Glen Situmorang
Samuel Pangaribuan
Gramaldy
Handika Pratama

Syntax SELECT Nama_Pengguna FROM Pengguna; adalah perintah SQL yang digunakan untuk mengambil atau menampilkan nilai dari kolom "Nama_Pengguna" dari tabel "Pengguna". Jadi, query ini akan mengambil nilai dari kolom "Nama_Pengguna" dari setiap baris dalam tabel "Pengguna" dan menampilkannya sebagai hasil.

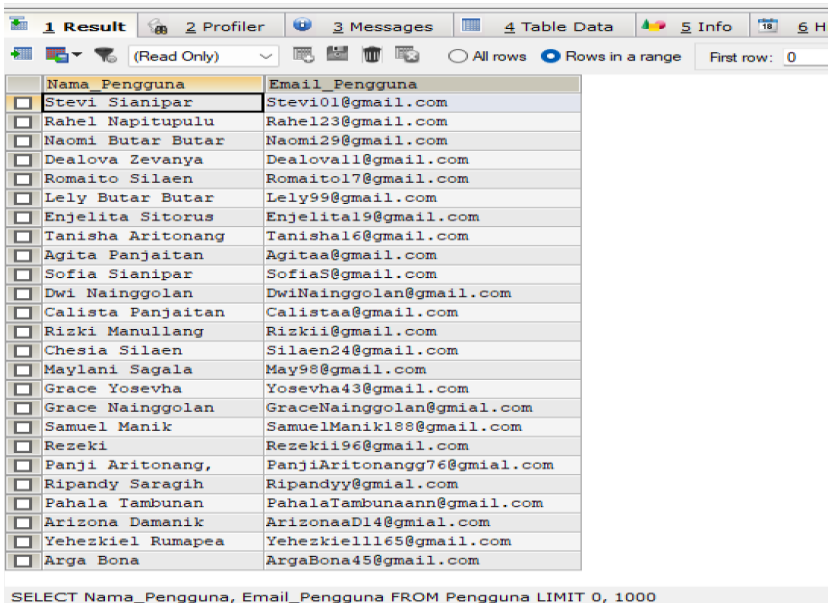
- **SELECT Email_Pengguna FROM Pengguna;**



Email_Pengguna
Stevi01@gmail.com
Rahel23@gmail.com
Naomi29@gmail.com
Dealova11@gmail.com
Romaito17@gmail.com
Lely99@gmail.com
Enjelita19@gmail.com
Tanisha16@gmail.com
Agitaa@gmail.com
SofiaS@gmail.com
DwiNainggolan@gmail.com
Calistaa@gmail.com
Rizkii@gmail.com
Silaen24@gmail.com
May98@gmail.com
Yosevha43@gmail.com
GraceNainggolan@gmial.com
SamuelManik188@gmail.com
Rezekii96@gmail.com
PanjiAritonangg76@gmial.com
Ripandyy@gmial.com
PahalaTambunaann@gmail.com

Query `SELECT Email_Pengguna FROM Pengguna;` digunakan untuk mengambil atau menampilkan nilai dari kolom "Email_Pengguna" dari tabel "Pengguna".query ini untuk mendapatkan daftar email pengguna yang terdaftar dalam tabel "Pengguna". Dengan menjalankan query ini, Anda dapat mengakses dan menggunakan informasi email pengguna untuk berbagai keperluan, seperti mengirimkan notifikasi, verifikasi, atau melakukan analisis data terkait pengguna.

- **SELECT Nama_Pengguna, Email_Pengguna FROM Pengguna;**



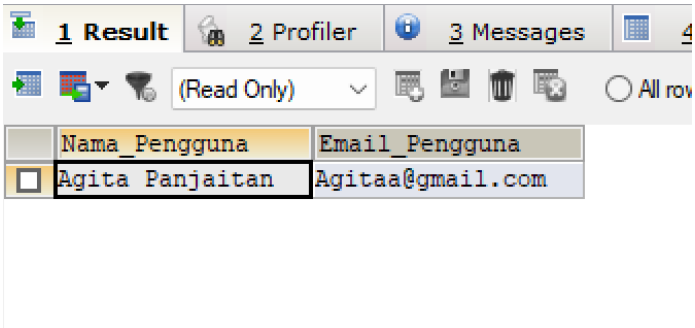
Nama_Pengguna	Email_Pengguna
Stevi Sianipar	Stevi01@gmail.com
Rahel Napitupulu	Rahel23@gmail.com
Naomi Butar Butar	Naomi29@gmail.com
Dealova Zevanya	Dealova11@gmail.com
Romaito Silaen	Romaito17@gmail.com
Lely Butar Butar	Lely99@gmail.com
Enjelita Sitorus	Enjelita19@gmail.com
Tanisha Aritonang	Tanisha16@gmail.com
Agita Panjaitan	Agitaa@gmail.com
Sofia Sianipar	SofiaS@gmail.com
Dwi Nainggolan	DwiNainggolan@gmail.com
Calista Panjaitan	Calistaa@gmail.com
Rizki Manullang	Rizkii@gmail.com
Chesia Silaen	Silaen24@gmail.com
Maylani Sagala	May98@gmail.com
Grace Yosevha	Yosevha43@gmail.com
Grace Nainggolan	GraceNainggolan@gmial.com
Samuel Manik	SamuelManik188@gmail.com
Rezeki	Rezekii96@gmail.com
Panji Aritonang,	PanjiAritonangg76@gmial.com
Ripandy Saragih	Ripandyy@gmial.com
Pahala Tambunan	PahalaTambunaann@gmail.com
Arizona Damanik	ArizonaaD14@gmial.com
Yehezkiel Rumapea	Yehezkiell165@gmail.com
Arga Bona	ArgaBona45@gmail.com

SELECT Nama_Pengguna, Email_Pengguna FROM Pengguna LIMIT 0, 1000

Query `SELECT Nama_Pengguna, Email_Pengguna FROM Pengguna;` digunakan untuk menampilkan dua kolom dari tabel "Pengguna", yaitu "Nama_Pengguna" dan "Email_Pengguna".

query ini untuk mendapatkan daftar lengkap nama dan email pengguna yang terdaftar dalam tabel "Pengguna". Dengan menjalankan query ini, Anda dapat mengakses informasi lengkap tentang pengguna, yang mungkin dibutuhkan untuk berbagai tujuan seperti pengiriman email, verifikasi identitas, atau analisis data. Ini memungkinkan Anda untuk mengakses informasi yang lebih komprehensif tentang pengguna dalam satu langkah query.

- **SELECT Nama_Pengguna, Email_Pengguna FROM Pengguna WHERE id_pengguna = 9;;**



Syntax ini digunakan untuk memilih kolom "Nama_Pengguna" dan "Email_Pengguna" dari tabel "Pengguna" dimana nilai kolom "id_pengguna" sama dengan 9.syntax ini untuk mendapatkan informasi spesifik tentang pengguna dengan ID tertentu (dalam kasus ini, ID pengguna 9). Dengan menggunakan klausa WHERE, kita dapat memfilter baris yang dikembalikan oleh query hanya untuk pengguna yang memenuhi kriteria tertentu, dalam hal ini, hanya pengguna dengan ID 9. Ini memungkinkan kita untuk mendapatkan informasi yang lebih spesifik tentang pengguna yang dibutuhkan tanpa harus melihat semua entri dalam tabel Pengguna.

- **SELECT * FROM Pengguna ORDER BY id_pengguna ASC;**

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows

id_pengguna	Nama_Pengguna	Email_Pengguna
1	Stevi Sianipar	Stevi01@gmail.com
2	Rahel Napitupulu	Rahel23@gmail.com
3	Naomi Butar Butar	Naomi29@gmail.com
4	Dealova Zevanya	Dealova11@gmail.com
5	Romaito Silaen	Romaito17@gmail.com
6	Lely Butar Butar	Lely99@gmail.com
7	Enjelita Sitorus	Enjelita19@gmail.com
8	Tanisha Aritonang	Tanisha16@gmail.com
9	Agita Panjaitan	Agitaa@gmail.com
10	Sofia Sianipar	SofiaS@gmail.com
11	Dwi Nainggolan	DwiNainggolan@gmail.com
12	Calista Panjaitan	Calistaa@gmail.com
13	Rizki Manullang	Rizkii@gmail.com
14	Chesia Silaen	Silaen24@gmail.com
15	Maylani Sagala	May98@gmail.com
16	Grace Yosevha	Yosevha43@gmail.com
17	Grace Nainggolan	GraceNainggolan@gmial.com
18	Samuel Manik	SamuelManik188@gmail.com
19	Rezeki	Rezekii96@gmail.com
20	Panji Aritonang,	PanjiAritonangg76@gmial.com
21	Ripandy Saragih	Ripandyy@gmial.com
22	Pahala Tambunan	PahalaTambunaann@gmail.com
23	Arizona Damanik	ArizonaaD14@gmial.com
24	Yehezkiel Rumapea	Yehezkiell165@gmail.com
25	Arga Bona	ArgaBona45@gmail.com

SELECT * FROM Pengguna ORDER BY id_pengguna ASC LIMIT 0, 1000

Syntax ini digunakan untuk mengambil seluruh baris dari tabel "Pengguna" dan mengurutkannya berdasarkan nilai kolom "id_pengguna" secara berurutan dari yang terkecil (ascending). query ini untuk mendapatkan daftar pengguna yang diurutkan berdasarkan ID pengguna secara teratur dari yang terkecil ke yang terbesar. Dengan demikian, kita dapat dengan mudah melihat dan menganalisis data pengguna dalam urutan yang terstruktur, memudahkan dalam melihat pola data atau menemukan informasi tertentu yang mungkin dicari berdasarkan urutan ID pengguna.

- **SELECT * FROM Pengguna LIMIT 8;**

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info

(Read Only) All rows Rows in a range First row

id_pengguna	Nama_Pengguna	Email_Pengguna
1	Stevi Sianipar	Stevi01@gmail.com
2	Rahel Napitupulu	Rahel23@gmail.com
3	Naomi Butar Butar	Naomi29@gmail.com
4	Dealova Zevanya	Dealova11@gmail.com
5	Romaito Silaen	Romaito17@gmail.com
6	Lely Butar Butar	Lely99@gmail.com
7	Enjelita Sitorus	Enjelita19@gmail.com
8	Tanisha Aritonang	Tanisha16@gmail.com

```
SELECT * FROM Pengguna LIMIT 8
```

Syntax ini digunakan untuk mengambil seluruh baris dari tabel "Pengguna" dan mengurutkannya berdasarkan nilai kolom "id_pengguna" secara berurutan dari yang terkecil (ascending). query ini untuk mendapatkan daftar pengguna yang diurutkan berdasarkan ID pengguna secara teratur dari yang terkecil ke yang terbesar. Dengan demikian, kita dapat dengan mudah melihat dan menganalisis data pengguna dalam urutan yang terstruktur, memudahkan dalam melihat pola data atau menemukan informasi tertentu yang mungkin dicari berdasarkan urutan ID pengguna..

- **SELECT * FROM Pengguna WHERE Nama_Pengguna LIKE 'A%';**

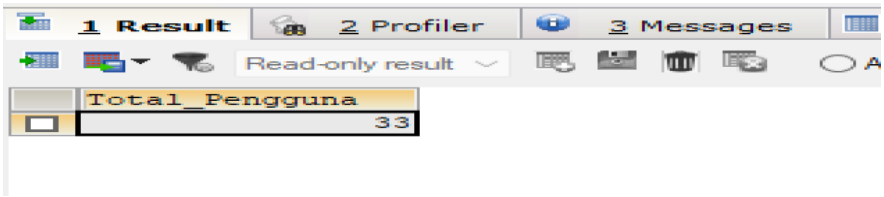
1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 His

(Read Only) All rows Rows in a range First row: 0

id_pengguna	Nama_Pengguna	Email_Pengguna
9	Agita Panjaitan	Agitaa@gmail.com
23	Arizona Damanik	ArizonaaD14@gmial.com
25	Arga Bona	ArgaBona45@gmail.com
27	Alfred Manurung	Alfred35@gmaiaal.com

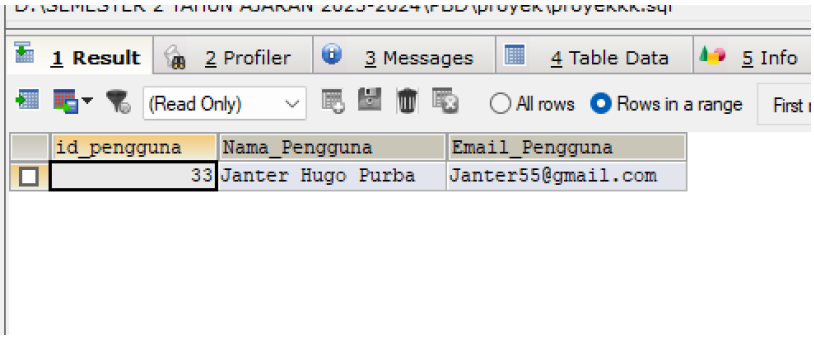
Syntax ini digunakan untuk mengambil semua baris dari tabel "Pengguna" di mana nilai kolom "Nama_Pengguna" dimulai dengan huruf 'A'. Syntax ini untuk memfilter data pengguna berdasarkan awalan nama pengguna. Dengan menggunakan "LIKE 'A%'", query ini akan mengembalikan semua baris di mana nilai kolom "Nama_Pengguna" dimulai dengan huruf 'A', sehingga memungkinkan pengguna untuk mendapatkan daftar pengguna yang memiliki awalan nama tertentu untuk analisis lebih lanjut atau keperluan lainnya.

- **SELECT COUNT(*) AS Total_Pengguna FROM Pengguna;**



Syntax ini digunakan untuk menghitung jumlah total baris (rekord) yang ada dalam tabel "Pengguna". Kemudian, hasil perhitungan tersebut akan ditampilkan sebagai kolom baru dengan nama "Total_Pengguna". syntax ini untuk memberikan informasi tentang jumlah total pengguna yang terdaftar dalam sistem. Hal ini dapat membantu dalam pemantauan dan analisis data, serta memberikan gambaran tentang seberapa besar basis pengguna yang dimiliki oleh sistem.

- **SELECT COUNT(*) AS Total_Pengguna FROM Pengguna;**



Syntax ini digunakan untuk mengambil semua data dari tabel "Pengguna" di mana nilai ID pengguna (id_pengguna) sama dengan nilai maksimum (MAX) dari ID pengguna di tabel "Pengguna". Dengan kata lain, ini mengambil baris dari tabel "Pengguna" yang memiliki ID pengguna tertinggi. syntax ini untuk mendapatkan informasi tentang pengguna terakhir yang ditambahkan ke dalam tabel. Ini bisa berguna dalam situasi di mana Anda ingin melihat atau melakukan operasi terhadap data pengguna terbaru yang dimasukkan ke dalam sistem. Misalnya, untuk melakukan tindakan tertentu atau memverifikasi informasi pengguna terbaru.

c) Buat 10 query untuk menampilkan data dari beberapa tabel (harus mencakup contoh inner JOIN, left/right JOIN)

```
SELECT Admin.*, Pekerjaan.*
FROM Admin
LEFT JOIN Pekerjaan ON Admin.id_Admin = Pekerjaan.id_Admin;
```

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog	id_Pekerjaan	Jenis_Pekerjaan	id_Admin
1	0812345678	winagea22@gmail.com	wina_gea	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
2	0832459032	bastiansitanggang05@gmail.com	bastian.sitanggang	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
3	0893245678	endrogel11@gmail.com	endrogel_sibaranii	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Jadi, hasil dari syntax ini akan menghasilkan gabungan antara data admin dan pekerjaan, di mana setiap baris admin akan memiliki semua informasi yang sesuai dengan pekerjaan yang terkait dengan admin tersebut, dan jika tidak ada pekerjaan yang terkait, nilai-nilai pekerjaan akan menjadi NULL.

```
SELECT Admin.*, Pekerjaan.*, Pengguna.*
FROM Admin
LEFT JOIN Pekerjaan ON Admin.id_Admin = Pekerjaan.id_Admin
LEFT JOIN Pengguna ON Admin.id_Admin = Pengguna.id_pengguna;
```

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog	id_Pekerjaan	Jenis_Pekerjaan	id_Admin	id_pengguna	Nama_Pe
1	0812345678	winagea22@gmail.com	wina_gea	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1	Stevi Si
2	0832459032	bastiansitanggang05@gmail.com	bastian.sitanggang	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	2	Rahel Na
3	0893245678	endrogel11@gmail.com	endrogel_sibaranii	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	3	Naomi Bu

Jadi, hasil dari syntax ini akan menghasilkan gabungan antara data admin, pekerjaan, dan pengguna. Setiap baris admin akan memiliki semua informasi yang sesuai dengan pekerjaan dan pengguna yang terkait dengan admin tersebut, dan jika tidak ada pekerjaan atau pengguna yang terkait, nilai-nilai pekerjaan atau pengguna akan menjadi NULL.

```
SELECT Pekerjaan.*, Admin.*
FROM Pekerjaan
LEFT JOIN Admin ON Pekerjaan.id_Admin = Admin.id_Admin;
```

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_Pekerjaan	Jenis_Pekerjaan	id_Admin	id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog
1	Network Administrator	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
2	Network Engineer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
3	Network Architect	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
4	Database Administrator	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
5	Computer Systems Analyst	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
6	Computer and Information System	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
7	Network Technician	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
8	Embedded Systems Engineer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
9	Embedded Electronics Engineer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
10	Control Systems Engineer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
11	Embedded Network Engineer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
12	Cybersecurity Embedded Sytem Developer	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Jadi, hasil dari perintah ini akan menghasilkan gabungan antara data pekerjaan dan informasi admin yang terkait dengan setiap pekerjaan. Setiap baris pekerjaan akan memiliki semua informasi yang sesuai dengan admin yang terkait, dan jika tidak ada admin yang terkait, nilai-nilai admin akan menjadi NULL.

SELECT Pekerjaan.*, Pengguna.*

FROM Pekerjaan

LEFT JOIN Pengguna ON Pekerjaan.id_Pekerjaan = Pengguna.id_pengguna;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_Pekerjaan	Jenis_Pekerjaan	id_Admin	id_pengguna	Nama_Pengguna	Email_Pengguna
1	Network Administrator	(NULL)	1	Stevi Sianipar	Stevi01@gmail.com
2	Network Engineer	(NULL)	2	Rahel Napitupulu	Rahel23@gmail.com
3	Network Architect	(NULL)	3	Naomi Butar Butar	Naomi29@gmail.com
4	Database Administrator	(NULL)	4	Dealova Zevanya	Dealova11@gmail.com
5	Computer Systems Analyst	(NULL)	5	Romaito Silaen	Romaito17@gmail.com
6	Computer and Information System	(NULL)	6	Lely Butar Butar	Lely99@gmail.com
7	Network Technician	(NULL)	7	Enjelita Sitorus	Enjelita19@gmail.com
8	Embedded Systems Engineer	(NULL)	8	Tanisha Aritonang	Tanisha16@gmail.com
9	Embedded Electronics Engineer	(NULL)	9	Agita Panjaitan	Agitaa@gmail.com
10	Control Systems Engineer	(NULL)	10	Sofia Sianipar	SofiaS@gmail.com
11	Embedded Network Engineer	(NULL)	11	Dwi Nainggolan	DwiNainggolan@gmail.com
12	Cybersecurity Embedded Sytem Developer	(NULL)	12	Calista Panjaitan	Calistaa@gmail.com
13	Cloud Engineer	(NULL)	13	Rizki Manullang	Rizkii@gmail.com

Jadi, hasil dari perintah ini akan menghasilkan gabungan antara data pekerjaan dan data pengguna yang terkait dengan setiap pekerjaan. Setiap baris pekerjaan akan memiliki semua informasi yang sesuai dengan pengguna yang terkait, dan jika tidak ada pengguna yang terkait, nilai-nilai pengguna akan menjadi NULL.

SELECT Pengguna.*, Admin.*

FROM Admin

RIGHT JOIN Pengguna ON Admin.id_Admin = Pengguna.id_pengguna;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_pengguna	Nama_Pengguna	Email_Pengguna	id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog
1	Stevi Sianipar	Stevi01@gmail.com	1	0812345678	winagea22@gmail.com	wina_gea	(NULL)	(NULL)
2	Rahel Napitupulu	Rahel23@gmail.com	2	0832459032	bastiansitanggang05@gmail.com	bastian.sitanggang	(NULL)	(NULL)
3	Naomi Butar Butar	Naomi29@gmail.com	3	0893245678	endrogel11@gmail.com	endrogel_sibaranii	(NULL)	(NULL)
4	Dealova Zevanya	Dealova11@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
5	Romaito Silaen	Romaito17@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
6	Lely Butar Butar	Lely99@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
7	Enjelita Sitorus	Enjelita19@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
8	Tanisha Aritonang	Tanisha16@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
9	Agita Panjaitan	Agitaa@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
10	Sofia Sianipar	SofiaS@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
11	Dwi Nainggolan	DwiNainggolan@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
12	Calista Panjaitan	Calistaa@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
13	Rizki Manullang	Rizkii@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Hasil dari perintah ini adalah gabungan antara data pengguna dan data admin yang terkait, di mana setiap baris pengguna akan memiliki semua informasi yang sesuai dengan admin yang terkait, dan jika tidak ada admin yang terkait, nilai-nilai admin akan menjadi NULL.

SELECT Pengguna.*, Admin.*

FROM Pengguna

LEFT JOIN Admin ON Pengguna.id_pengguna = Admin.id_Admin;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_pengguna	Nama_Pengguna	Email_Pengguna	id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog
1	Stevi Sianipar	Stevi01@gmail.com	1	0812345678	winagea22@gmail.com	wina_gea	(NULL)	(NULL)
2	Rahel Napitupulu	Rahel23@gmail.com	2	0832459032	bastiansitanggang05@gmail.com	bastian.sitanggang	(NULL)	(NULL)
3	Naomi Butar Butar	Naomi29@gmail.com	3	0893245678	endrogel11@gmail.com	endrogel_sibaranii	(NULL)	(NULL)
4	Dealova Zevanya	Dealova11@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
5	Romaito Silaen	Romaito17@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
6	Lely Butar Butar	Lely99@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
7	Enjelita Sitorus	Enjelita19@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
8	Tanisha Aritonang	Tanisha16@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
9	Agita Panjaitan	Agitaa@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
10	Sofia Sianipar	SofiaS@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
11	Dwi Nainggolan	DwiNainggolan@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
12	Calista Panjaitan	Calistaa@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
13	Rizki Manullang	Rizkii@gmail.com	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

Hasil dari perintah ini adalah gabungan antara data pengguna dan data admin yang terkait, di mana setiap baris pengguna akan memiliki semua informasi yang sesuai dengan admin yang terkait, dan jika tidak ada admin yang terkait, nilai-nilai admin akan menjadi NULL.

SELECT Pengguna.*, Admin.*

FROM Pengguna

INNER JOIN Admin ON Pengguna.id_pengguna = Admin.id_Admin;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_pengguna	Nama_Pengguna	Email_Pengguna	id_Admin	NO_Telp	Email_Admin	Sosial_Media_Admin	Pekerjaan	Blog
1	Stevi Sianipar	Stevi01@gmail.com	1	0812345678	winagea22@gmail.com	wina_gea	(NULL)	(NULL)
2	Rahel Napitupulu	Rahel23@gmail.com	2	0832459032	bastiansitanggang05@gmail.com	bastian.sitanggang	(NULL)	(NULL)
3	Naomi Butar Butar	Naomi29@gmail.com	3	0893245678	endrogel11@gmail.com	endrogel_sibaranii	(NULL)	(NULL)

Hasil dari perintah ini adalah gabungan antara data pengguna dan data admin yang memiliki nilai yang sesuai dalam kedua tabel. Baris-baris yang tidak memiliki pasangan yang sesuai dalam tabel lain akan diabaikan dalam hasilnya.

SELECT Blog.*, Pekerjaan.*

FROM Blog

LEFT JOIN Pekerjaan ON Blog.id_Admin = Pekerjaan.id_Admin;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

id_Blog	Panduan_Karir	pengembangan_diri	id_Admin	id_Pekerjaan	Jenis_Pekerjaan	id_Admin
1	C/V Resume	Cara Pengembangan Diri Untuk Menggali Potensi Diri	(NULL)	(NULL)	(NULL)	(NULL)
2	Cover Letter	Growth Mindset: Arti, Penerapan dan Manfaatnya dalam Karir	(NULL)	(NULL)	(NULL)	(NULL)
3	Portofolio & Personal Branding	Refeleksi Diri: Manfaat, Pengertian, Kepentingannnya, dan Caranya	(NULL)	(NULL)	(NULL)	(NULL)
4	Skkill Interview	Kenali 5 Ciri Ciri Burnout dan Cara Mengatasinya!	(NULL)	(NULL)	(NULL)	(NULL)
5	Tips Cari Kerja	Mau Sukses di Uaia Muda? Tiru 10 Kebiasaan Ini!	(NULL)	(NULL)	(NULL)	(NULL)

Hasil dari perintah ini adalah gabungan antara data dari tabel Blog dan Pekerjaan. Semua baris dari tabel Blog akan tetap disertakan dalam hasil, bahkan jika tidak ada pasangan yang sesuai dalam tabel Pekerjaan. Jika tidak ada pasangan yang sesuai dalam Pekerjaan, nilai kolom-kolom dari tabel Pekerjaan untuk baris-baris ini akan berisi NULL.

SELECT Blog.*, Pengguna.*

FROM Pengguna

RIGHT JOIN Blog ON Pengguna.id_pengguna = Blog.id_Admin;

1 Result | 2 Profiler | 3 Messages | 4 Table Data | 5 Info | 6 History

(Read Only) All rows Rows in a range First row: 0 1000 rows Refresh

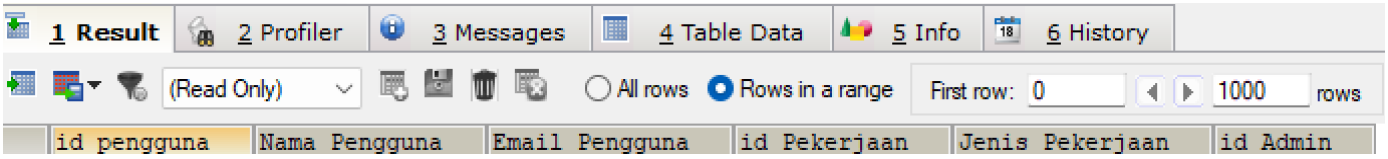
id_Blog	Panduan_Karir	pengembangan_diri	id_Admin	id_pengguna	Nama_Pengguna	Email_Pengguna
1	C/V Resume	Cara Pengembangan Diri Untuk Menggali Potensi Diri	(NULL)	(NULL)	(NULL)	(NULL)
2	Cover Letter	Growth Mindset: Arti, Penerapan dan Manfaatnya dalam Karir	(NULL)	(NULL)	(NULL)	(NULL)
3	Portofolio & Personal Branding	Refeleksi Diri: Manfaat, Pengertian, Kepentingannnya, dan Caranya	(NULL)	(NULL)	(NULL)	(NULL)
4	Skkill Interview	Kenali 5 Ciri Ciri Burnout dan Cara Mengatasinya!	(NULL)	(NULL)	(NULL)	(NULL)
5	Tips Cari Kerja	Mau Sukses di Uaia Muda? Tiru 10 Kebiasaan Ini!	(NULL)	(NULL)	(NULL)	(NULL)

Hasil dari perintah ini adalah gabungan antara data dari tabel Pengguna dan Blog. Semua baris dari tabel Blog akan tetap disertakan dalam hasil, bahkan jika tidak ada pasangan yang sesuai dalam tabel Pengguna. Jika tidak ada pasangan yang sesuai dalam Pengguna, nilai kolom-kolom dari tabel Pengguna untuk baris-baris ini akan berisi NULL.

SELECT Pengguna.*, Pekerjaan.*

FROM Pengguna

INNER JOIN Pekerjaan ON Pengguna.id_pengguna = Pekerjaan.id_Admin;



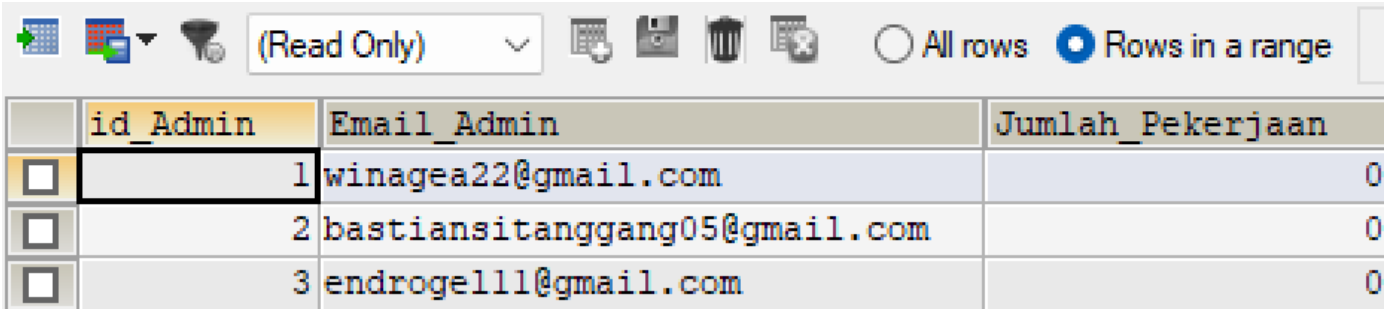
Hasil dari perintah ini adalah gabungan antara data dari tabel Pengguna dan Pekerjaan dimana hanya baris-baris yang memiliki nilai yang sesuai dalam kedua tabel yang akan disertakan. Jika tidak ada pasangan yang sesuai, baris tersebut tidak akan dimasukkan dalam hasil.

d) Buat 5 penggunaan subquery

- **SELECT id_Admin, Email_Admin,**

  (SELECT COUNT(*) FROM Pekerjaan WHERE Pekerjaan.id_Admin = Admin.id_Admin) AS Jumlah_Pekerjaan

  FROM Admin;



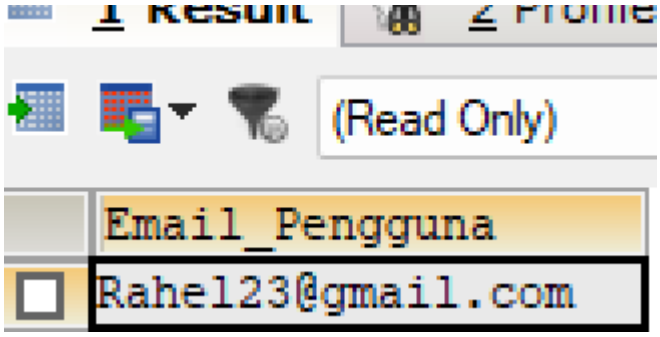
id_Admin	Email_Admin	Jumlah_Pekerjaan
1	winagea22@gmail.com	0
2	bastiansitanggang05@gmail.com	0
3	endrogelll@gmail.com	0

Perintah SQL ini melakukan pencarian dan penghitungan jumlah pekerjaan yang terkait dengan setiap admin di tabel Admin. Ini dilakukan dengan menggunakan subquery di dalam pernyataan SELECT. Hasil akhirnya adalah daftar admin dengan email mereka dan jumlah pekerjaan yang terkait dengan masing-masing admin. Subquery digunakan di sini untuk memberikan informasi tambahan yang relevan tentang setiap admin.

- **SELECT Email_Pengguna**

  FROM Pengguna

  WHERE id_pengguna = (SELECT id_Admin FROM Admin WHERE id_Admin = 2);

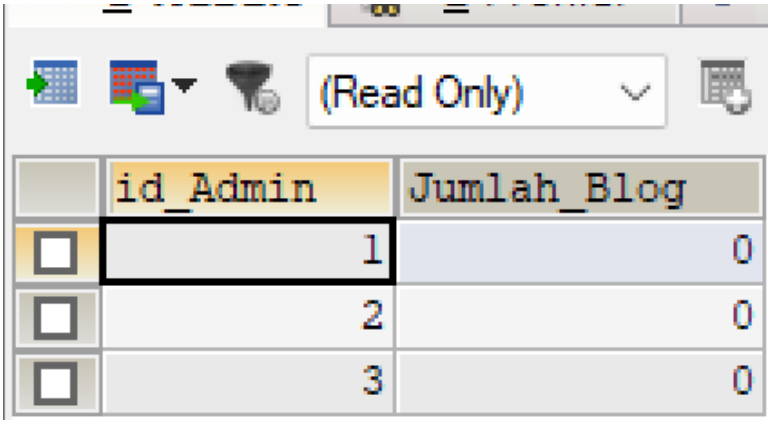


Syntax SQL di atas merupakan sebuah pernyataan SELECT yang digunakan untuk menampilkan email pengguna yang terkait dengan seorang admin tertentu dari tabel Pengguna. Ini dilakukan dengan menggunakan subquery di dalam klausa WHERE. Dengan demikian, syntax tersebut akan mengembalikan email pengguna yang terkait dengan admin yang memiliki id_Admin = 2.

- **SELECT id_Admin,**

  (SELECT COUNT(*) FROM Blog WHERE Blog.id_Admin = Admin.id_Admin) AS Jumlah_Blog

  FROM Admin;



id_Admin	Jumlah_Blog
1	0
2	0
3	0

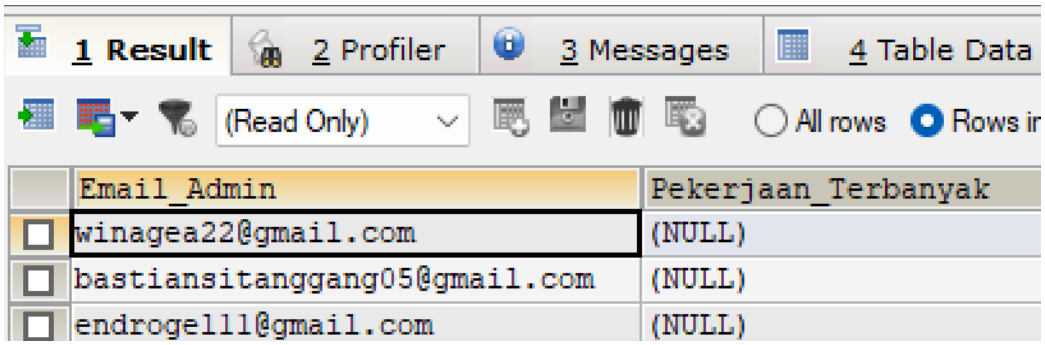
Syntax SQL di atas merupakan sebuah pernyataan SELECT yang digunakan untuk menampilkan jumlah blog yang terkait dengan setiap admin dari tabel Admin. Ini dilakukan dengan menggunakan subquery di dalam klausa SELECT. Dengan demikian, syntax tersebut akan mengembalikan id_Admin dari setiap admin bersama dengan jumlah blog yang terkait dengan admin tersebut.

❖ **SELECT Nama_Pengguna**
FROM Pengguna
WHERE Nama_Pengguna NOT IN (SELECT Email_Admin FROM Admin);

Nama_Pengguna
Stevi Sianipar
Rahel Napitupulu
Naomi Butar Butar
Dealova Zevanya
Romaito Silaen
Lely Butar Butar
Enjelita Sitorus
Tanisha Aritonang
Agita Panjaitan
Sofia Sianipar
Dwi Nainggolan
Calista Panjaitan
Rizki Manullang
Chesia Silaen
Maylani Sagala
Grace Yosevha
Grace Nainggolan

Syntax SQL di atas merupakan sebuah pernyataan SELECT yang digunakan untuk memilih nama pengguna dari tabel Pengguna dimana nama pengguna tidak ada dalam daftar email admin yang terdapat dalam tabel Admin. Ini dilakukan dengan menggunakan subquery di dalam klausa WHERE. Dengan demikian, syntax tersebut akan mengembalikan nama pengguna dari tabel Pengguna yang tidak terkait dengan email admin yang ada dalam tabel Admin.

❖ **SELECT A.Email_Admin,**
(SELECT Jenis_Pekerjaan
FROM Pekerjaan
WHERE Pekerjaan.id_Admin = A.id_Admin
GROUP BY id_Admin
ORDER BY COUNT(*) DESC
LIMIT 1) AS Pekerjaan_Terbanyak
FROM Admin A;



1 Result | 2 Profiler | 3 Messages | 4 Table Data

(Read Only) All rows Rows ir

Email_Admin	Pekerjaan_Terbanyak
winagea22@gmail.com	(NULL)
bastiansitanggang05@gmail.com	(NULL)
endrogelll@gmail.com	(NULL)

Syntax SQL di atas merupakan sebuah pernyataan SELECT yang digunakan untuk mengambil email admin dan jenis pekerjaan yang paling banyak terkait dengan admin tersebut dari tabel Admin dan Pekerjaan. Ini menggunakan subquery di dalam klausa SELECT untuk mencari jenis pekerjaan terbanyak untuk setiap admin.

Link video

https://youtu.be/JOG_oz98KYI