

STRUKTUR QUERY LANGUAGE

TYPE DATA SQL (GLOBAL TYPE)

Kategori	Tipe Data	Deskripsi Singkat
Angka (Numerik)	INT	Bilangan bulat (biasanya 4 byte), sangat umum untuk ID atau hitungan
	TINYINT	Bilangan bulat kecil (1 byte), cocok untuk status (0/1) atau boolean
	BIGINT	Bilangan bulat sangat besar (8 byte), cocok untuk jumlah besar atau waktu Unix
	DECIMAL(p, s) / NUMERIC	Bilangan desimal tetap, cocok untuk data keuangan
	FLOAT / DOUBLE	Bilangan pecahan desimal (floating point), cocok untuk data ilmiah atau statistik
Teks (String)	VARCHAR(n)	Teks dengan panjang variabel, paling umum dipakai untuk nama, email, dll
	CHAR(n)	Teks dengan panjang tetap, cocok untuk kode tetap (seperti kode pos, kode negara)
	TEXT	Teks panjang, cocok untuk deskripsi, komentar, atau artikel
Tanggal/Waktu	DATE	Format tanggal (YYYY-MM-DD)
	DATETIME	Format tanggal dan waktu (YYYY-MM-DD HH:MM:SS), sangat umum untuk mencatat waktu

Data:

		id	no	nama	nik	jk	alamat	divisi	level	jabatan	gaji	tanggal_input
<input type="checkbox"/>	Edit Copy Delete	1	A-001	RIFKY	MS-001	pria	Alamat: 2, Jl. RS. Fatmawati Raya No.64, RT.2/RW.5...	Big Data	senior	Manager	10000000	2025-05-28
<input type="checkbox"/>	Edit Copy Delete	2	A-002	HANAFI	MS-002	pria	Alamat: 2, Jl. RS. Fatmawati Raya No.64, RT.2/RW.5...	Backend Dev	middle	Staff	1000000	2025-05-27
<input type="checkbox"/>	Edit Copy Delete	3	A-003	PEPPY	MS-003	pria	Alamat: 2, Jl. RS. Fatmawati Raya No.64, RT.2/RW.5...	Backend Dev	basic	Staff	1000000	2025-05-29
<input type="checkbox"/>	Edit Copy Delete	4	A-004	Penny	MS-004	wanita	Alamat: 2, Jl. RS. Fatmawati Raya No.64, RT.2/RW.5...	Frontend Dev	basic	Staff	1000000	2025-05-25
<input type="checkbox"/>	Edit Copy Delete	5	A-005	BOY	MS-005	pria	Alamat: 2, Jl. RS. Fatmawati Raya No.64, RT.2/RW.5...	Frontend Dev	middle	Staff	1000000	2025-05-30

☐ Check all With selected: Edit Copy Delete Export

Latihan

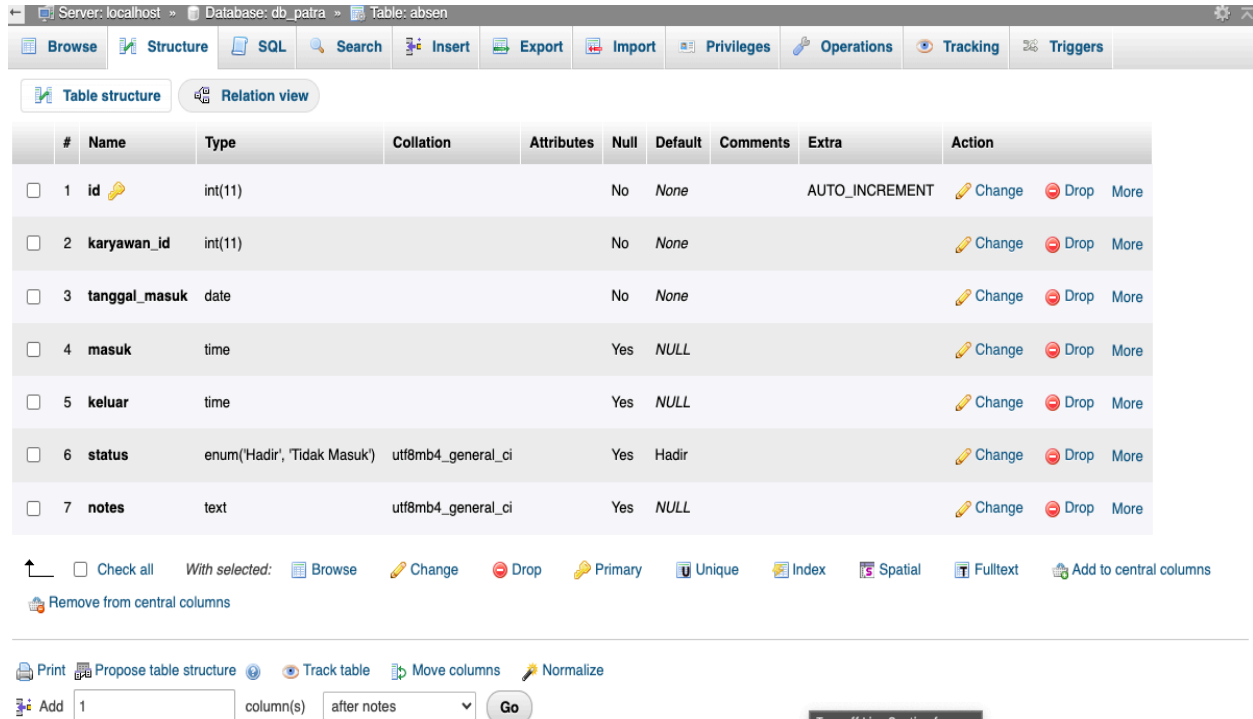
1. Buatlah database patra dan tabel karyawan
2. Buatlah filter data berdasarkan nama depan yang berawalan huruf "R"
3. Buatlah filter data berdasarkan nama belakang yang berawalan huruf "Y"
4. Buatlah urutan data berdasarkan a - z
5. Buatlah urutan data berdasarkan z - a
6. Buatlah query data antara 1 dan 3
7. Buat Konversi Tanggal yang dari angka menjadi huruf supaya lebih dipahami
8. Buatlah Konversi bulan menjadi huruf supaya lebih mudah di pahami
9. Buat tanggal_input menjadi tanggal indonesia
10. Buatlah selisih antara 2 tanggal
11. Buatlah query untuk menghilangkan penggandaan pada data level
12. Buat lah query mencari gaji terbesar
13. Buat lah query mencari gaji terkecil
14. Buat lah query mencari gaji rata-rata
15. Buat lag query mencari jumlah data gaji
16. Buat lah query untuk menampilkan jika wanita mendapatkan cuti hamil sedangkan pria tidak
17. Buatlah query untuk mengubah nama menjadi huruf besar
18. Buatlah query untuk mengubah nama menjadi huruf kecil
19. Buatlah query untuk memotong nama antara 1 dan 3 huruf
20. Buatlah query untuk menggabungkan huruf

Jawaban Soal No.2

- 2. SELECT * FROM `karyawan` where nama LIKE 'R%';**
- 3. SELECT * FROM `karyawan` where nama LIKE '%Y';**
- 4. SELECT * FROM `karyawan` ORDER BY ID ASC**
- 5. SELECT * FROM `karyawan` ORDER BY ID DESC**
- 6. SELECT * FROM `karyawan` WHERE id BETWEEN 1 AND 3**
- 7. SELECT DAYNAME(tanggal_input) as tanggal_huruf FROM `karyawan`**
- 8. SELECT MONTHNAME(tanggal_input) as bulan_huruf FROM `karyawan`;**
- 9. SELECT DATE_FORMAT(tanggal_input, '%d-%M-%Y') as tanggal_full_huruf FROM `karyawan`;**
- 10. SELECT DATEDIFF(tanggal_input, NOW()) as tanggal_selesih FROM `karyawan`;**
- 11. SELECT DISTINCT(level) as hilangkan_penggandaan FROM `karyawan`;**
- 12. SELECT MAX(gaji) as gaji_teratas FROM `karyawan`**
- 13. SELECT MIN(gaji) as gaji_terbawah FROM `karyawan`**
- 14. SELECT AVG(gaji) as gaji_rata_rata FROM `karyawan`**
- 15. SELECT COUNT(gaji) as jumlah_gaji FROM `karyawan`**
- 16. SELECT IF(jk = 'wanita', "cuti hamil", "tidak dapat cuti") as cuti FROM karyawan;**
- 17. SELECT UPPER(nama) as nama_besar FROM `karyawan`**
- 18. SELECT LOWER(nama) as nama_kecil FROM `karyawan`**
- 19. SELECT SUBSTR(nama, 1, 2) as potong_text FROM `karyawan`**
- 20. SELECT CONCAT(nama, "-", nik) AS gabung_text FROM karyawan;**

2 = Case Study Data Karyawan Dan Absen

Join Data Antar Tabel



The screenshot shows the phpMyAdmin interface for a database named 'db_patra'. The 'absen' table is selected, and its structure is displayed in 'Table structure' view. The table has 7 columns: 'id' (int(11), primary key, AUTO_INCREMENT), 'karyawan_id' (int(11)), 'tanggal_masuk' (date), 'masuk' (time), 'keluar' (time), 'status' (enum('Hadir', 'Tidak Masuk'), utf8mb4_general_ci), and 'notes' (text, utf8mb4_general_ci). Below the table structure, there are options to 'Check all', 'With selected', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Spatial', 'Fulltext', and 'Add to central columns'. At the bottom, there is a 'Print' button, a 'Propose table structure' button, a 'Track table' button, a 'Move columns' button, and a 'Normalize' button. A search bar is also present with the text '1' and a 'Go' button.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	karyawan_id	int(11)			No	None			Change Drop More
3	tanggal_masuk	date			No	None			Change Drop More
4	masuk	time			Yes	NULL			Change Drop More
5	keluar	time			Yes	NULL			Change Drop More
6	status	enum('Hadir', 'Tidak Masuk')	utf8mb4_general_ci		Yes	Hadir			Change Drop More
7	notes	text	utf8mb4_general_ci		Yes	NULL			Change Drop More

Latihan

1. Buatlah tabel di atas
2. Gabungkan lah 2 tabel antara karyawan dan absen untuk mendapatkan hasil karyawan yang masuk saja gambaran seperti berikut :
 - a. Nama karyawan, Tanggal Masuk, Masuk , Keluar dan Status Kehadiran.
3. Gabungkan lah 2 tabel antara karyawan dan absen untuk mendapatkan hasil karyawan yang tidak masuk saja gambaran seperti berikut :
 - a. Nama karyawan, Tanggal Masuk, Masuk , Keluar dan Status Kehadiran.
4. Gabungkan lah 2 tabel untuk mendapatkan jumlah karyawan yang masuk dalam waktu sebulan
5. Gabungkan lah 2 tabel untuk mendapatkan jumlah karyawan yang tidak masuk dalam waktu sebulan

6. Gabungkan lah 2 tabel untuk di jadikan sebagai data grafik untuk menampilkan data absen karyawan di bulan yang sedang berjalan

Jawaban

2. **SELECT * FROM `karyawan` LEFT JOIN absen ON karyawan.id = absen.karyawan_id WHERE absen.status = 'hadir' GROUP BY karyawan.id;**

3. **SELECT * FROM `karyawan` LEFT JOIN absen ON karyawan.id = absen.karyawan_id WHERE absen.status = 'tidak_hadir' GROUP BY karyawan.id;**

4. **SELECT *,COUNT(absen.status) as jml FROM `karyawan` LEFT JOIN absen ON karyawan.id = absen.karyawan_id WHERE absen.status = 'hadir' AND MONTH(karyawan.tanggal_input) = MONTH(NOW()) GROUP BY karyawan.id;**

5.**SELECT *,COUNT(absen.status) as jml FROM `karyawan` LEFT JOIN absen ON karyawan.id = absen.karyawan_id WHERE absen.status = 'tidak_hadir' AND MONTH(karyawan.tanggal_input) = MONTH(NOW()) GROUP BY karyawan.id;**

6.**SELECT karyawan.id,karyawan.nama, absen.status, karyawan.tanggal_input, absen.masuk, absen.keluar, COUNT(absen.status) as jml FROM `karyawan` LEFT JOIN absen ON karyawan.id = absen.karyawan_id WHERE MONTH(karyawan.tanggal_input) = MONTH(NOW()) GROUP BY karyawan.id;**

3 = Case Study Stored Prosedure

Create new routine

Details

Routine name

filter_user

Type

PROCEDURE

Parameters

Direction	Name	Type	Length
IN	filter_username	VARCHAR	255

Add parameter

Definition

```
1 SELECT * FROM users WHERE username = filter_username;
```

Is deterministic

☐

Go

Close

Selanjutnya kita bisa exceute dengan memanggil fungsi tersebut dengan command, CALL, seperti contoh berikut :

Call filter_user()

Cara Lain Untuk Mengolah data stored procedure ini bisa dengan menggunakan full command line dengan mengetikkan perintah seperti ini :

```
DELIMITER //  
CREATE PROCEDURE GetKaryawanByID(IN karyawanID INT)  
BEGIN  
    SELECT * FROM karyawan WHERE id_karyawan = karyawanID;  
END //  
DELIMITER ;
```

Selanjutnya sama seperti di atas dengan memanggil fungsi tersebut dengan call , “call GetKaryawanByID(1)”;

Latihan

1. Buatlah Stored Prosedured Diatas dengan menggunakan command line sesuai dari code diatas.
2. Panggilah dengan nama fungsi , bisa disesuaikan dengan nama fungsi diatas

SOAL SOAL SQL LANJUTAN

1. Total Gaji per Departemen

Tampilkan nama departemen dan total gaji yang dibayarkan ke semua karyawan di masing-masing departemen, urutkan dari yang paling besar.

2. Karyawan Berdasarkan Lama Kerja

Tampilkan nama karyawan, nama departemen, dan jumlah tahun kerja (dibulatkan ke bawah) bagi karyawan yang sudah bekerja lebih dari 5 tahun.

3. Rata-rata Gaji Berdasarkan Jenis Kelamin

Hitung rata-rata gaji karyawan berdasarkan jenis kelamin (L atau P), dan tampilkan dalam urutan rata-rata tertinggi.

4. Karyawan Tertua di Setiap Departemen

Tampilkan satu karyawan tertua dari masing-masing departemen beserta nama dan tanggal lahirnya.

5. Jumlah Karyawan yang Ulang Tahun Bulan Ini

Hitung berapa jumlah karyawan yang berulang tahun di bulan yang sama dengan bulan saat ini (`MONTH(NOW())`).

Jawaban

1. SELECT d.nama_departemen, SUM(k.gaji) AS total_gaji FROM karyawan k LEFT JOIN departemen d ON k.id_departemen = d.id_departemen GROUP BY d.nama_departemen ORDER BY total_gaji DESC;

2. SELECT k.nama, d.nama_departemen,
FLOOR(DATEDIFF(CURDATE(), k.tanggal_masuk) / 365) AS tahun_kerja
FROM karyawan k
JOIN departemen d ON k.id_departemen = d.id_departemen
WHERE DATEDIFF(CURDATE(), k.tanggal_masuk) / 365 > 5;

3. SELECT jk, ROUND(AVG(gaji), 2) AS rata_gaji
FROM karyawan
GROUP BY jk
ORDER BY rata_gaji DESC;

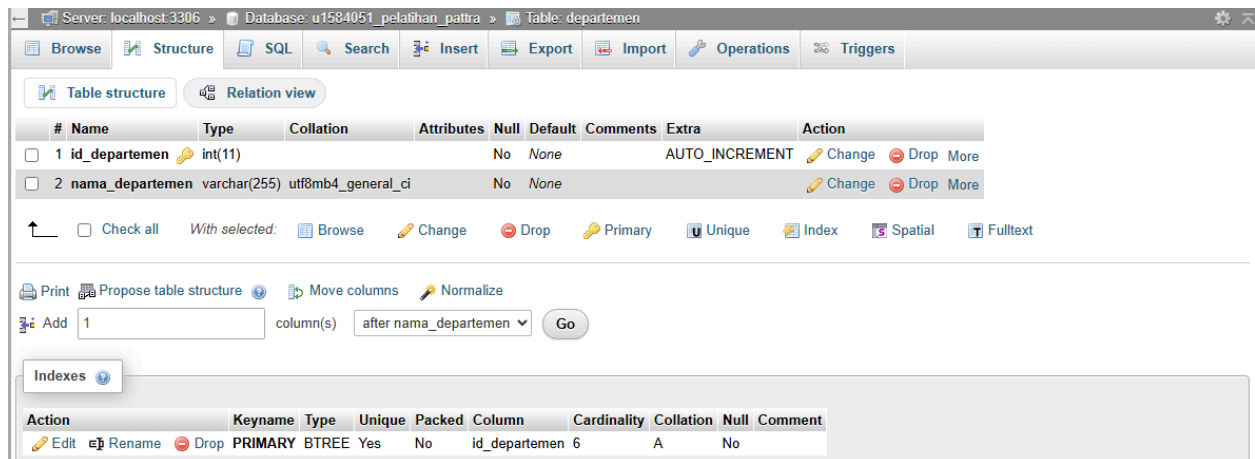
4. SELECT k.id, k.nama, d.nama_departemen, k.tanggal_lahir
FROM karyawan k
JOIN departemen d ON k.id_departemen = d.id_departemen
WHERE k.tanggal_lahir = (
SELECT MIN(k2.tanggal_lahir)
FROM karyawan k2
WHERE k2.id_departemen = k.id_departemen
);

5. SELECT COUNT(*) AS jumlah_ulang_tahun
FROM karyawan
WHERE MONTH(tanggal_lahir) = MONTH(CURDATE());

Operational CRUD

(Create, Read, Update, Delete)

Struktur dan Data



The screenshot shows a database management interface with the following details:

- Server: localhost:3306
- Database: u1584051_pelatihan_patra
- Table: departemen

Table structure:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_departemen	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	nama_departemen	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	id_departemen	6	A	No	

1. Insert / Create
 - a. Buatlah Query Insert data kedalam tabel departemen
2. Update /
 - a. Buatlah Query Update data kedalam tabel departemen
3. Delete
 - a. Buatlah Query Delete data di dalam tabel departemen
4. Select/ Read
 - a. Buatlah Query Select All di dalam tabel departemen

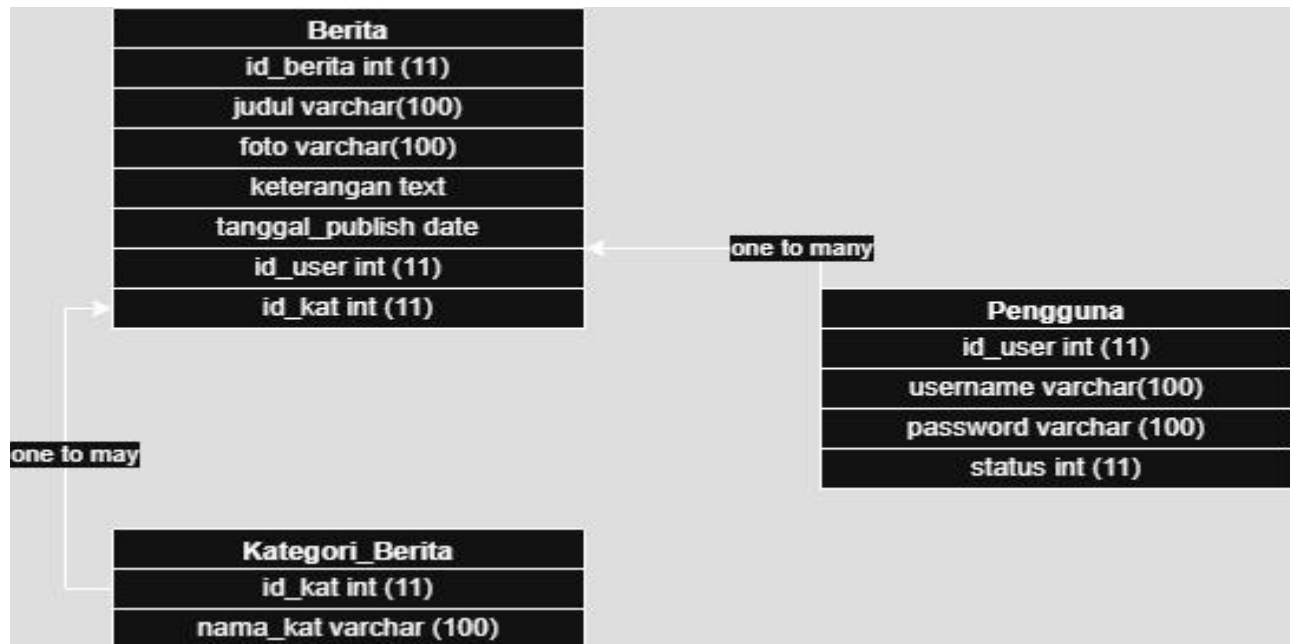
Jawaban

1. `Insert into departemen(nama_departemen)values('Departemen Baru');`
2. `Update departemen SET nama_departemen ='Departemen Baru' WHERE id_departemen = 2;`
3. `Delete FROM departemen WHERE id_departemen = 2;`
4. `SELECT * FROM departemen;`

CLASS DIAGRAM / RELATIONAL DATABASE / ENTITY

RELATIONAL DATABASE

Case Study News Portal Apps



Keterangan

Table terdiri dari 3 yaitu

- Berita
- Kategori Berita
- Pengguna

Relasi antar tabel menggunakan relasi one to many dan setiap tabel diwajibkan mempunyai trigger untuk menghubungkan antar masing masing tabel.

Next().

UJIAN FINAL SQL QUERY

Case Studi, E-Commerce.

1. Normalization
 - a. Database e-commerce
 - i. Table product
 - ii. Table orders
 - iii. Table Users
 - iv. Product_category , next
2. Database Relation Table
3. Xampp Implementation
4. Examination

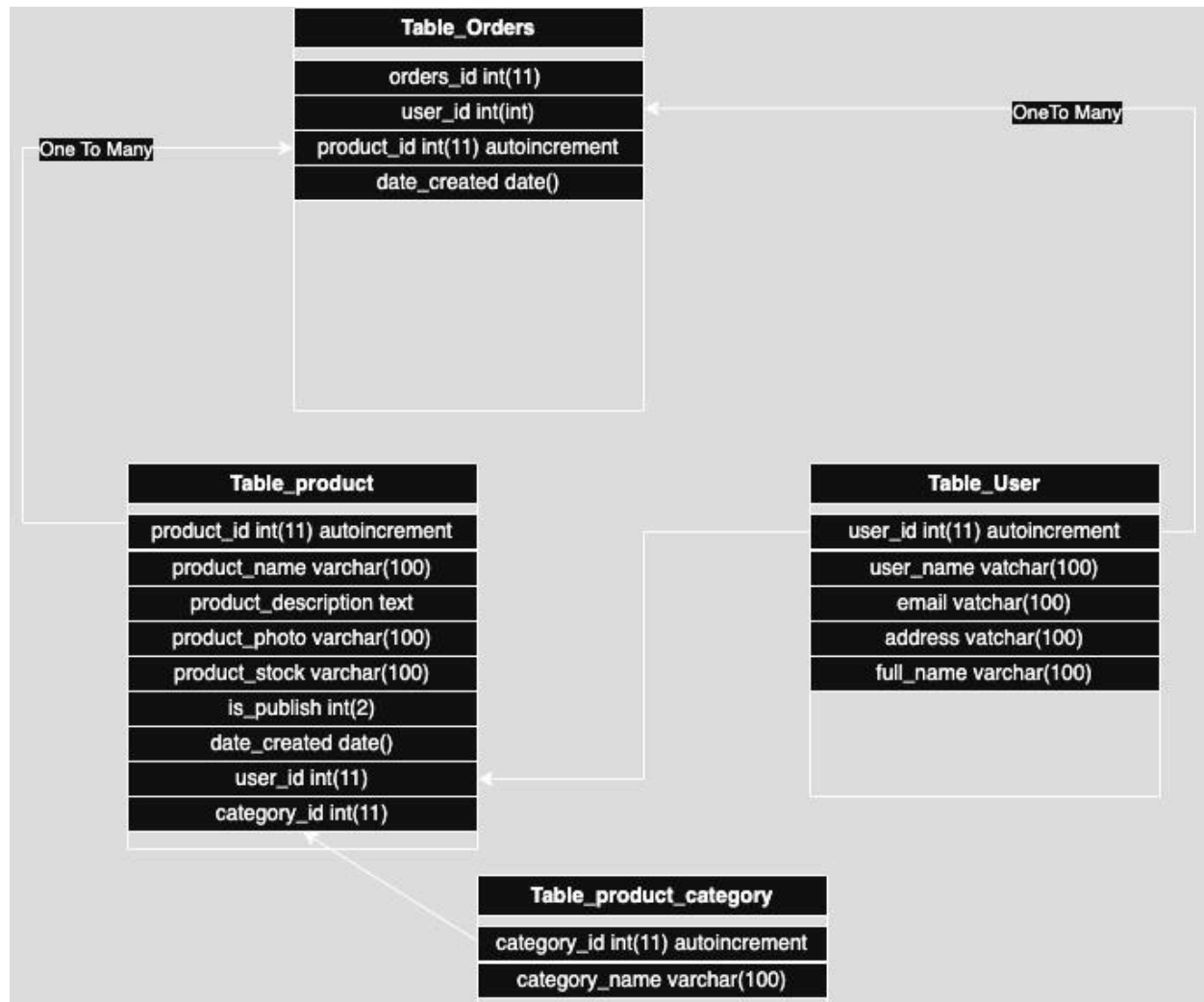
1. Normalization

Adalah Pembuatan Relasi tabel sehingga tidak ada lagi redundan data atau duplikasi tabel antar tabel

Latihan :

- a. Buatlah normalisasi dari database e-commerce
- b. Buatlah Tabel Beserta Relasi Antar Field nya
- c. Design Normalisasi menggunakan
=<https://app.diagrams.net/>=
- d. Jika Ada kekurangan tabel bisa ditambahkan atau di modify
- e. next

2. Class Diagram



Latihan

- 1. Buat lah class diagram diatas dengan menggunakan sql**
- 2. Setelah di buat tolong tampilkan semua list produk**
- 3. Tampilkan produk berdasarkan kategorinya**
- 4. Tampilkan produk yang paling banyak di beli**
- 5. Tampilkan produk yang paling sedikit di beli**
- 6. Tampilkan pengguna yang sering melakukan pemesanan**
- 7. Tampilkan list pengguna secara keseluruhan**
- 8. Create data per masing masing table minimal 5 data dengan validasi dari pengajar**
- 9. Update data per masing masing table dengan validasi dari pengajar**
- 10. Delete data satuan table dengan validasi dari pengajar**

Latihan Kelompok

Kelompok Terdiri dari 5 Orang, Untuk Point 50, dan Masing Masing Orang 10 Point untuk 3 Kelompok.

Setiap Kelompok Akan Di wajibkan Memilih Salah Satu Dari 5 Point Berikut :

- 1. Koleksi buku Apps**
 - 2. Koleksi foto Apps**
 - 3. Koleksi video Apps**
 - 4. Diary Apps**
 - 5. Note Apps**
-
- a. Tugas Per-Masing Masing Kelompok Adalah Membuat Class Diagram Relational Database dengan Menentukan Database dan Tabel dari Masing Masing Kasus.**
 - b. Memberikan Keterangan di setiap tabel beserta atribut nya. Semakin jelas keterangan , maka akan semakin tinggi nilai pointnya.**
 - c. Melakukan Operasi CRUD(Create, Read, Update, Delete) dalam kasus kali ini,**
 - d. Command Line dari CRUD Sebelumnya Harus di tulis dan akan dinilai oleh pemberi materi**
 - e. Setiap Kelompok harus memilih 1 orang untuk mewakili tim nya untuk mempresentasikan hasil dari analisa tim nya.**