

Model Prisma Enum Finansial

Table of Contents

- [Penjelasan:](#)
- [Penjelasan:](#)
- [SQL untuk Menyisipkan Data](#)
- [Penyesuaian:](#)
- [Enum Metric_Balance_Sheet](#)
- [Penjelasan:](#)
- [SQL untuk Menyisipkan Data](#)
- [Penjelasan:](#)
- [SQL untuk Menambahkan Data Kosong pada Jun-2023](#)
- [Penjelasan:](#)
- [Enum Metric_Balance_Sheet](#)
- [Penjelasan:](#)
- [SQL untuk Menyisipkan Data](#)
- [Penjelasan:](#)
- [Enum untuk Metric](#)
- [Enum untuk Units](#)
- [Penyesuaian:](#)

TOTAL ASSETS
Current Assets
Receivables
Inventory

Non-current Assets
TOTAL LIABILITIES
Current Liability
Bank Loan
Shareholderloan
Long-Term Liability
Bank Loan
Shareholderloan
EQUITY
Share Capital
Retained Earning
TOTAL LIABILITIES AND EQUITY

jadikan enum model prisma

Berikut adalah model Prisma dengan enum untuk mencerminkan struktur yang Anda minta:

```
prisma

enum AssetType {
  CURRENT_ASSET
  RECEIVABLES
  INVENTORY
  NON_CURRENT_ASSET
}

enum LiabilityType {
  CURRENT_LIABILITY
  BANK_LOAN
  SHAREHOLDER_LOAN
  LONG_TERM_LIABILITY
}

enum EquityType {
  SHARE_CAPITAL
  RETAINED_EARNING
}

model FinancialRecord {
  id          Int      @id @default(autoincrement())
  type        String    // "ASSET", "LIABILITY", or "EQUITY"
  subtype     String?   // To be assigned based on enums
  description  String
  amount      Float
  createdAt   DateTime  @default(now())
  updatedAt   DateTime  @updatedAt
}
```

Penjelasan:

1. Enums:

- `AssetType` : Mencakup jenis aset seperti *Current Asset*, *Receivables*, *Inventory*, dan *Non-current Asset*.
- `LiabilityType` : Mencakup jenis kewajiban seperti *Current Liability*, *Bank Loan*, *Shareholder Loan*, dan *Long-term Liability*.
- `EquityType` : Mencakup *Share Capital* dan *Retained Earning*.

2. Model `FinancialRecord` :

- `type` : Menandakan kategori utama (misalnya, "ASSET", "LIABILITY", "EQUITY").
- `subtype` : Bisa diisi dengan salah satu nilai dari enum yang relevan.
- `amount` : Untuk menyimpan nilai aset, kewajiban, atau ekuitas.
- `createdAt` dan `updatedAt` : Otomatis mengatur waktu pembuatan dan pembaruan.

Jika diperlukan untuk mengaitkan setiap record ke entitas lain (misalnya, perusahaan atau akun tertentu), tambahkan relasi ke model yang relevan.

duh i mean, wrong enum

```
enum Metric
Revenue
(Cost of Good Sold)
Gross Profit
(Selling, General, and Administrative)
Operating Profit
(Finance Cost)
Finance Income
FX Gain/(Loss)
Other Income
Tax Benefit/(expenses)
Net Income
```

Berikut adalah model Prisma dengan enum untuk metrik yang Anda maksudkan:

prisma

```
enum Metric {
  REVENUE
  COST_OF_GOOD_SOLD
  GROSS_PROFIT
  SELLING_GENERAL_ADMINISTRATIVE
  OPERATING_PROFIT
  FINANCE_COST
  FINANCE_INCOME
  FX_GAIN_LOSS
  OTHER_INCOME
  TAX_BENEFIT_EXPENSES
  NET_INCOME
}
```

```
model FinancialMetric {
  id      Int    @id @default(autoincrement())
  metric   Metric // Enum type from Metric
  description String?
  value    Float  // Stores the amount for this metric
  date     DateTime // Date for the financial metric
  createdAt DateTime @default(now())
  updatedAt DateTime @updatedAt
}
```

Penjelasan:

- 1. **Metric Enum:**
 - Berisi semua metrik keuangan yang telah Anda sebutkan, termasuk *Revenue*, *Gross Profit*, *Operating Profit*, dan sebagainya.
 - Nama enum ditulis dalam huruf besar dengan format snake_case untuk kemudahan identifikasi dalam kode.
- 2. **FinancialMetric Model:**
 - metric** : Menggunakan enum **Metric** untuk mengklasifikasikan metrik keuangan.
 - description** : Opsional, untuk menyimpan detail tambahan.
 - value** : Menyimpan nilai dari metrik tertentu.
 - date** : Mencatat kapan metrik ini berlaku, misalnya untuk laporan bulanan atau tahunan.
 - createdAt** dan **updatedAt** : Untuk mencatat waktu pembuatan dan pembaruan.

Model ini cocok digunakan untuk menyimpan data metrik keuangan dan dapat diperluas sesuai kebutuhan (misalnya, menambahkan relasi ke entitas lain seperti perusahaan).

No	Metric	Units	Jun-2023	Dec-2023	Jun-2024	Dec-2024	Jun-2025
1	Revenue	USD	325,489,625	658,103,051	330,403,888		
2	(Cost of Good Sold)	USD	-208,481,737	-437,353,874			
				-211,826,714			
3	Gross Profit	USD	117,007,888	220,749,177	118,577,174		
4	(Selling, General, and Administrative)	USD	-6,632,711				
			-13,062,223	-8,248,363			
5	Operating Profit	USD	110,375,177	207,686,954	110,328,811		
6	(Finance Cost)	USD	-45,416,299	-91,014,923	-42,702,684		
7	Finance Income	USD	6,924,446	10,478,180	4,133,372		
8	FX Gain/(Loss)	USD	3,510,121	521,095	-9,115,167		
9	Other Income	USD	2,016	2,122,350	-		
10	Tax Benefit/(expenses)	USD	-2,371,996	1,461,080			
			1,504,996				
11	Net Income	USD	73,023,465	131,254,736	64,149,328		

buatkan sql di postgre supabase untuk memasukan ke tabel ini dari data diatas

```

model financial_statement {
  id      Int    @id @default(autoincrement())
  project_id Int?
  metric   Metric?
  units    String? @db.VarChar(10)
  value    Decimal? @db.Decimal(20, 2)
  reportType FinancialReportType? @map("report_type")
  month_period Int?
  year_period Int?
  created_at      DateTime? @default(now()) // Waktu data
dibuat
  updated_at      DateTime? @updatedAt    // Waktu data
di-update
  projects projects? @relation(fields: [project_id], references:
[id], onDelete: NoAction, onUpdate: NoAction)
}

enum Metric {
  REVENUE
  COST_OF_GOOD_SOLD
  GROSS_PROFIT
  SELLING_GENERAL_ADMINISTRATIVE
  OPERATING_PROFIT
  FINANCE_COST
  FINANCE_INCOME
  FX_GAIN_LOSS
  OTHER_INCOME
  TAX_BENEFIT_EXPENSES
  NET_INCOME
}

enum FinancialReportType {
  PROFIT_AND_LOSS
  BALANCE_SHEET
  CASHFLOW
}

```

Berikut adalah SQL untuk menyisipkan data dari tabel di atas ke tabel `financial_statement` pada PostgreSQL (dengan model sesuai dengan skema yang diberikan):

SQL untuk Menyisipkan Data

```

sql

-- Enum and table setup already assumed. Inserting data:

INSERT INTO financial_statement (project_id, metric, units, value, report_type, month_period,
year_period, created_at, updated_at) VALUES
-- Jun-2023
(NULL, 'REVENUE', 'USD', 325489625.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'COST_OF_GOOD_SOLD', 'USD', -208481737.00, 'PROFIT_AND_LOSS', 6, 2023, now(),
now()),
(NULL, 'GROSS_PROFIT', 'USD', 117007888.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),

```

```
(NULL, 'SELLING_GENERAL_ADMINISTRATIVE', 'USD', -6632711.00, 'PROFIT_AND_LOSS', 6, 2023,
now(), now()),
(NULL, 'OPERATING_PROFIT', 'USD', 110375177.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'FINANCE_COST', 'USD', -45416299.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'FINANCE_INCOME', 'USD', 6924446.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'FX_GAIN_LOSS', 'USD', 3510121.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'OTHER_INCOME', 'USD', 2016.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),
(NULL, 'TAX_BENEFIT_EXPENSES', 'USD', -2371996.00, 'PROFIT_AND_LOSS', 6, 2023, now(),
now()),
(NULL, 'NET_INCOME', 'USD', 73023465.00, 'PROFIT_AND_LOSS', 6, 2023, now(), now()),

-- Dec-2023
(NULL, 'REVENUE', 'USD', 658103051.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'COST_OF_GOOD_SOLD', 'USD', -437353874.00, 'PROFIT_AND_LOSS', 12, 2023, now(),
now()),
(NULL, 'GROSS_PROFIT', 'USD', 220749177.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'SELLING_GENERAL_ADMINISTRATIVE', 'USD', -13062223.00, 'PROFIT_AND_LOSS', 12,
2023, now(), now()),
(NULL, 'OPERATING_PROFIT', 'USD', 207686954.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'FINANCE_COST', 'USD', -91014923.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'FINANCE_INCOME', 'USD', 10478180.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'FX_GAIN_LOSS', 'USD', 521095.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'OTHER_INCOME', 'USD', 2122350.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),
(NULL, 'TAX_BENEFIT_EXPENSES', 'USD', 1461080.00, 'PROFIT_AND_LOSS', 12, 2023, now(),
now()),
(NULL, 'NET_INCOME', 'USD', 131254736.00, 'PROFIT_AND_LOSS', 12, 2023, now(), now()),

-- Jun-2024
(NULL, 'REVENUE', 'USD', 330403888.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'COST_OF_GOOD_SOLD', 'USD', -211826714.00, 'PROFIT_AND_LOSS', 6, 2024, now(),
now()),
(NULL, 'GROSS_PROFIT', 'USD', 118577174.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'SELLING_GENERAL_ADMINISTRATIVE', 'USD', -8248363.00, 'PROFIT_AND_LOSS', 6, 2024,
now(), now()),
(NULL, 'OPERATING_PROFIT', 'USD', 110328811.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'FINANCE_COST', 'USD', -42702684.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'FINANCE_INCOME', 'USD', 4133372.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'FX_GAIN_LOSS', 'USD', -9115167.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now()),
(NULL, 'TAX_BENEFIT_EXPENSES', 'USD', 1504996.00, 'PROFIT_AND_LOSS', 6, 2024, now(),
now()),
(NULL, 'NET_INCOME', 'USD', 64149328.00, 'PROFIT_AND_LOSS', 6, 2024, now(), now());
```

Penyesuaian:

- 1. `project_id` : Nilai `NULL` karena tidak ada informasi proyek pada data ini.
- 2. `report_type` : Semua diset sebagai `'PROFIT_AND_LOSS'` , karena tabel ini sesuai dengan laporan laba rugi.
- 3. `month_period` dan `year_period` : Menggunakan data periode waktu dari header (misalnya, Jun-2023 = bulan 6 dan tahun 2023).
- 4. `value` : Disesuaikan dengan format angka desimal (hingga dua desimal jika diperlukan).

Jika Anda ingin menambahkan data lebih banyak atau menyesuaikan data dengan proyek tertentu, tambahkan kolom yang relevan seperti `project_id` .

```
saya sudah rotasi tabel dan menghapus kolom report type

sehingga profit, balance, cashflow memiliki tabel sendiri2,
sekarang buat enum

model financial_statement_balance_sheet {
  id      Int    @id @default(autoincrement())
  project_id Int?
  metric   Metric_Balance_Sheet?
  units    String? @db.VarChar(10)
  value    Decimal? @db.Decimal(20, 2)
  month_period Int?
  year_period Int?
  created_at      DateTime? @default(now()) // Waktu data
dibuat
  updated_at      DateTime? @updatedAt    // Waktu data
di-update
  projects    projects? @relation(fields: [project_id], references:
[id], onDelete: NoAction, onUpdate: NoAction)
}

enum Metric_Balance_Sheet? {

  Cash from Operation
  Cash from Investing
  Cash from Financing
  Repayment Bank Loan
  Repayment Shareholderloan
  Cash Increase/(Decrease)
  Beginning Cash
  FX Gain/(Loss)
  Ending Cash
```

Berikut adalah definisi enum untuk tabel *balance sheet* dengan metrik yang Anda berikan:

Enum `Metric_Balance_Sheet`

```
prisma

enum Metric_Balance_Sheet {
  CASH_FROM_OPERATION
  CASH_FROM_INVESTING
  CASH_FROM_FINANCING
  REPAYMENT_BANK_LOAN
  REPAYMENT_SHAREHOLDER_LOAN
  CASH_INCREASE_DECREASE
```

```
BEGINNING_CASH
FX_GAIN_LOSS
ENDING_CASH
}
```

Penjelasan:

1. Nama-nama Enum:

- Semua nilai enum ditulis dalam format **SCREAMING_SNAKE_CASE** untuk mematuhi konvensi standar dalam Prisma.
- Penggunaan karakter khusus seperti "/" diubah menjadi `_` agar sesuai dengan sintaks Prisma.

2. Kolom `metric` di model `financial_statement_balance_sheet` :

- Akan menggunakan enum ini untuk menyimpan metrik laporan neraca.

Jika Anda memiliki tabel tambahan seperti untuk laporan *Profit and Loss* atau *Cashflow*, tambahkan enum serupa untuk masing-masing tabel.

```
input data kesini

model financial_statement_cashflow_statement {
  id      Int    @id @default(autoincrement())
  project_id Int?
  metric   Metric_Cashflow_Statement?
  units    String? @db.VarChar(10)
  value    Decimal? @db.Decimal(20, 2)
  month_period Int?
  year_period Int?
  created_at      DateTime? @default(now()) // Waktu data dibuat
  updated_at      DateTime? @updatedAt      // Waktu data di-update
  projects projects? @relation(fields: [project_id], references: [id], onDelete: NoAction, onUpdate: NoAction)
}

enum Metric_Cashflow_Statement {
  CASH_FROM_OPERATION
  CASH_FROM_INVESTING
  CASH_FROM_FINANCING
  REPAYMENT_BANK_LOAN
  REPAYMENT_SHAREHOLDER_LOAN
  CASH_INCREASE_DECREASE
  BEGINNING_CASH
  FX_GAIN_LOSS
  ENDING_CASH
}
```


project id = 1

Cashflow Statement

No	Metric	Units	Jun-2023	Dec-2023	Jun-2024	Dec-2024
Jun-2025						
1	Cash from Operation	USD	185,087,233	130,146,233		
2	Cash from Investing	USD	-50,854,448	-24,537		
3	Cash from Financing	USD	-221,598,763	-73,874,810		
4	Repayment Bank Loan	USD	-118,216,800	-73,662,600		
5	Repayment Shareholderloan	USD	-70,658,188	-		
6	Cash Increase/(Decrease)	USD	-87,365,978	56,246,886		
7	Beginning Cash	USD	152,168,028	114,355,707		
8	FX Gain/(Loss)	USD	-1,199,182	-		
9	Ending Cash	USD	63,602,868	170,602,593		

Berikut adalah SQL untuk menyisipkan data ke tabel `financial_statement_cashflow_statement` menggunakan data yang diberikan:

SQL untuk Menyisipkan Data

```
sql

INSERT INTO financial_statement_cashflow_statement (
  project_id, metric, units, value, month_period, year_period, created_at, updated_at
) VALUES
-- Dec-2023
(1, 'CASH_FROM_OPERATION', 'USD', 185087233.00, 12, 2023, now(), now()),
(1, 'CASH_FROM_INVESTING', 'USD', -50854448.00, 12, 2023, now(), now()),
(1, 'CASH_FROM_FINANCING', 'USD', -221598763.00, 12, 2023, now(), now()),
(1, 'REPAYMENT_BANK_LOAN', 'USD', -118216800.00, 12, 2023, now(), now()),
(1, 'REPAYMENT_SHAREHOLDER_LOAN', 'USD', -70658188.00, 12, 2023, now(), now()),
(1, 'CASH_INCREASE_DECREASE', 'USD', -87365978.00, 12, 2023, now(), now()),
(1, 'BEGINNING_CASH', 'USD', 152168028.00, 12, 2023, now(), now()),
(1, 'FX_GAIN_LOSS', 'USD', -1199182.00, 12, 2023, now(), now()),
(1, 'ENDING_CASH', 'USD', 63602868.00, 12, 2023, now(), now()),

-- Jun-2024
(1, 'CASH_FROM_OPERATION', 'USD', 130146233.00, 6, 2024, now(), now()),
(1, 'CASH_FROM_INVESTING', 'USD', -24537.00, 6, 2024, now(), now()),
(1, 'CASH_FROM_FINANCING', 'USD', -73874810.00, 6, 2024, now(), now()),
(1, 'REPAYMENT_BANK_LOAN', 'USD', -73662600.00, 6, 2024, now(), now()),
(1, 'REPAYMENT_SHAREHOLDER_LOAN', 'USD', 0.00, 6, 2024, now(), now()), -- No value for this
period, setting to 0
(1, 'CASH_INCREASE_DECREASE', 'USD', 56246886.00, 6, 2024, now(), now()),
(1, 'BEGINNING_CASH', 'USD', 114355707.00, 6, 2024, now(), now()),
(1, 'FX_GAIN_LOSS', 'USD', 0.00, 6, 2024, now(), now()), -- No value for this period, setting to 0
(1, 'ENDING_CASH', 'USD', 170602593.00, 6, 2024, now(), now());
```

Penjelasan:

1. `project_id` :
 - Seluruh data menggunakan `project_id = 1` seperti yang diminta.

2. `metric` :

- Nilai diambil dari enum `Metric_Cashflow_Statement` .

3. `units` :

- Semua data menggunakan `'USD'` seperti pada input.

4. `month_period` **dan** `year_period` :

- Periode diatur berdasarkan bulan dan tahun dari kolom di data Anda.

5. Nilai kosong (`-` atau tidak ada):

- Diganti dengan `0.00` pada SQL untuk memastikan tidak ada nilai `NULL` yang dimasukkan ke kolom `value` .

Silakan jalankan query ini untuk menyisipkan data ke tabel `financial_statement_cashflow_statement` .
Jika ada periode tambahan, Anda dapat menambahkan data dengan pola yang sama.

bisa ga tambahin jun 2023 data kosong

Tentu saja, kita bisa menambahkan data untuk periode *Jun-2023* dengan nilai kosong (*default value*). Dalam SQL, nilai kosong dapat diwakili dengan `0.00` untuk kolom `value` . Berikut adalah modifikasi SQL-nya:

SQL untuk Menambahkan Data Kosong pada *Jun-2023*

sql

```
INSERT INTO financial_statement_cashflow_statement (
  project_id, metric, units, value, month_period, year_period, created_at, updated_at
) VALUES
-- Jun-2023 (Data Kosong)
(1, 'CASH_FROM_OPERATION', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'CASH_FROM_INVESTING', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'CASH_FROM_FINANCING', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'REPAYMENT_BANK_LOAN', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'REPAYMENT_SHAREHOLDER_LOAN', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'CASH_INCREASE_DECREASE', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'BEGINNING_CASH', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'FX_GAIN_LOSS', 'USD', 0.00, 6, 2023, now(), now()),
(1, 'ENDING_CASH', 'USD', 0.00, 6, 2023, now(), now());
```

Penjelasan:

1. Periode:

- Data ini untuk periode *Jun-2023* (`month_period = 6` , `year_period = 2023`).

2. Nilai Kosong:

- Semua nilai untuk kolom `value` diatur ke `0.00` sebagai default.

3. `units` :

- Menggunakan 'USD' seperti pada data lain.

4. Waktu:

- Kolom created_at dan updated_at otomatis diatur ke waktu saat ini menggunakan now() .

Dengan query ini, data untuk Jun-2023 akan disisipkan meskipun tidak memiliki nilai aktual. Anda bisa menyesuaikan jika di masa depan ada data untuk periode tersebut.

```

buatkan enum

model financial_statement_balance_sheet {
  id      Int    @id @default(autoincrement())
  project_id Int?
  metric   Metric_Balance_Sheet?
  units    String? @db.VarChar(10)
  value    Decimal? @db.Decimal(20, 2)
  month_period Int?
  year_period Int?
  created_at      DateTime? @default(now()) // Waktu data
dibuat
  updated_at      DateTime? @updatedAt      // Waktu data
di-update
  projects    projects? @relation(fields: [project_id], references:
[id], onDelete: NoAction, onUpdate: NoAction)
}

enum Metric_Balance_Sheet {
  REVENUE
  COST_OF_GOOD_SOLD
  GROSS_PROFIT
  SELLING_GENERAL_ADMINISTRATIVE
  OPERATING_PROFIT
  FINANCE_COST
  FINANCE_INCOME
  FX_GAIN_LOSS
  OTHER_INCOME
  TAX_BENEFIT_EXPENSES
  NET_INCOME
}

TOTAL ASSETS
Current Assets
Receivables
Inventory
Non-current Assets
TOTAL LIABILITIES
Current Liability
Bank Loan
Shareholderloan
Long-Term Liability
Bank Loan
```

Shareholderloan
EQUITY
Share Capital
Retained Earning
TOTAL LIABILITIES AND EQUITY

Berikut adalah enum `Metric_Balance_Sheet` yang diperbarui berdasarkan metrik yang Anda berikan untuk laporan neraca (*Balance Sheet*).

Enum `Metric_Balance_Sheet`

```
prisma

enum Metric_Balance_Sheet {
  TOTAL_ASSETS
  CURRENT_ASSETS
  RECEIVABLES
  INVENTORY
  NON_CURRENT_ASSETS
  TOTAL_LIABILITIES
  CURRENT_LIABILITY
  BANK_LOAN
  SHAREHOLDER_LOAN
  LONG_TERM_LIABILITY
  EQUITY
  SHARE_CAPITAL
  RETAINED_EARNING
  TOTAL_LIABILITIES_AND_EQUITY
}
```

Penjelasan:

1. Format Penamaan:

- Menggunakan format **SCREAMING_SNAKE_CASE** sesuai dengan konvensi untuk enum.
- Nama metrik yang panjang atau memiliki spasi diganti dengan `_` (misalnya, `TOTAL LIABILITIES` menjadi `TOTAL_LIABILITIES`).

2. Penyesuaian Nama:

- `Non-current Assets` diubah menjadi `NON_CURRENT_ASSETS` .
- `Shareholderloan` diubah menjadi `SHAREHOLDER_LOAN` agar konsisten.
- `Total Liabilities and Equity` diubah menjadi `TOTAL_LIABILITIES_AND_EQUITY` .

Anda dapat menggunakan enum ini di model Prisma `financial_statement_balance_sheet` untuk menyimpan data laporan neraca dengan metrik yang sudah terstruktur.

sekarang buatkan sqlnya untuk insert data

Balance Sheet

No	Metric	Units	Jun-2023	Dec-2023	Jun-2024	Dec-2024
			Jun-2025			
1	TOTAL ASSETS	USD	2,247,617,664	2,181,071,520	2,163,524,692	
2	Current Assets	USD	688,287,011	590,174,532	615,394,597	
3	Receivables	USD	130,643,420	137,997,702	120,899,713	
4	Inventory	USD	18,265,271	22,010,991	15,246,251	
5	Non-current Assets	USD	1,559,330,653	1,590,896,988	1,548,130,095	
6	TOTAL LIABILITIES	USD	1,264,319,823	1,185,488,353	1,103,792,196	
7	Current Liability	USD	241,349,412	243,927,981	259,057,988	
8	Bank Loan	USD	98,216,800	98,216,800	93,752,400	
9	Shareholderloan	USD	14,491,470	27,183,778	56,166,718	
10	Long-Term Liability	USD	1,022,970,411	941,560,372	844,734,208	
11	Bank Loan	USD	733,882,511	711,595,300	644,534,650	
12	Shareholderloan	USD	190,723,864	134,364,838	105,381,898	
13	EQUITY	USD	983,297,841	995,583,167	1,059,732,496	
14	Share Capital	USD	376,778,224	376,778,224	376,778,224	
15	Retained Earning	USD	606,519,617	618,804,943	682,954,272	
16	TOTAL LIABILITIES AND EQUITY	USD	2,247,617,664	2,181,071,520	2,163,524,692	

Berikut adalah SQL untuk menyisipkan data *Balance Sheet* ke tabel

`financial_statement_balance_sheet` berdasarkan data yang Anda berikan:

SQL untuk Menyisipkan Data

```
sql

INSERT INTO financial_statement_balance_sheet (
  project_id, metric, units, value, month_period, year_period, created_at, updated_at
) VALUES
-- Jun-2023
(1, 'TOTAL_ASSETS', 'USD', 2247617664.00, 6, 2023, now(), now()),
(1, 'CURRENT_ASSETS', 'USD', 688287011.00, 6, 2023, now(), now()),
(1, 'RECEIVABLES', 'USD', 130643420.00, 6, 2023, now(), now()),
(1, 'INVENTORY', 'USD', 18265271.00, 6, 2023, now(), now()),
(1, 'NON_CURRENT_ASSETS', 'USD', 1559330653.00, 6, 2023, now(), now()),
(1, 'TOTAL_LIABILITIES', 'USD', 1264319823.00, 6, 2023, now(), now()),
(1, 'CURRENT_LIABILITY', 'USD', 241349412.00, 6, 2023, now(), now()),
(1, 'BANK_LOAN', 'USD', 98216800.00, 6, 2023, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 14491470.00, 6, 2023, now(), now()),
(1, 'LONG_TERM_LIABILITY', 'USD', 1022970411.00, 6, 2023, now(), now()),
(1, 'BANK_LOAN', 'USD', 733882511.00, 6, 2023, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 190723864.00, 6, 2023, now(), now()),
(1, 'EQUITY', 'USD', 983297841.00, 6, 2023, now(), now()),
(1, 'SHARE_CAPITAL', 'USD', 376778224.00, 6, 2023, now(), now()),
(1, 'RETAINED_EARNING', 'USD', 606519617.00, 6, 2023, now(), now()),
```

```
(1, 'TOTAL_LIABILITIES_AND_EQUITY', 'USD', 2247617664.00, 6, 2023, now(), now()),

-- Dec-2023
(1, 'TOTAL_ASSETS', 'USD', 2181071520.00, 12, 2023, now(), now()),
(1, 'CURRENT_ASSETS', 'USD', 590174532.00, 12, 2023, now(), now()),
(1, 'RECEIVABLES', 'USD', 137997702.00, 12, 2023, now(), now()),
(1, 'INVENTORY', 'USD', 22010991.00, 12, 2023, now(), now()),
(1, 'NON_CURRENT_ASSETS', 'USD', 1590896988.00, 12, 2023, now(), now()),
(1, 'TOTAL_LIABILITIES', 'USD', 1185488353.00, 12, 2023, now(), now()),
(1, 'CURRENT_LIABILITY', 'USD', 243927981.00, 12, 2023, now(), now()),
(1, 'BANK_LOAN', 'USD', 98216800.00, 12, 2023, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 27183778.00, 12, 2023, now(), now()),
(1, 'LONG_TERM_LIABILITY', 'USD', 941560372.00, 12, 2023, now(), now()),
(1, 'BANK_LOAN', 'USD', 711595300.00, 12, 2023, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 134364838.00, 12, 2023, now(), now()),
(1, 'EQUITY', 'USD', 995583167.00, 12, 2023, now(), now()),
(1, 'SHARE_CAPITAL', 'USD', 376778224.00, 12, 2023, now(), now()),
(1, 'RETAINED_EARNING', 'USD', 618804943.00, 12, 2023, now(), now()),
(1, 'TOTAL_LIABILITIES_AND_EQUITY', 'USD', 2181071520.00, 12, 2023, now(), now()),

-- Jun-2024
(1, 'TOTAL_ASSETS', 'USD', 2163524692.00, 6, 2024, now(), now()),
(1, 'CURRENT_ASSETS', 'USD', 615394597.00, 6, 2024, now(), now()),
(1, 'RECEIVABLES', 'USD', 120899713.00, 6, 2024, now(), now()),
(1, 'INVENTORY', 'USD', 15246251.00, 6, 2024, now(), now()),
(1, 'NON_CURRENT_ASSETS', 'USD', 1548130095.00, 6, 2024, now(), now()),
(1, 'TOTAL_LIABILITIES', 'USD', 1103792196.00, 6, 2024, now(), now()),
(1, 'CURRENT_LIABILITY', 'USD', 259057988.00, 6, 2024, now(), now()),
(1, 'BANK_LOAN', 'USD', 93752400.00, 6, 2024, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 56166718.00, 6, 2024, now(), now()),
(1, 'LONG_TERM_LIABILITY', 'USD', 844734208.00, 6, 2024, now(), now()),
(1, 'BANK_LOAN', 'USD', 644534650.00, 6, 2024, now(), now()),
(1, 'SHAREHOLDER_LOAN', 'USD', 105381898.00, 6, 2024, now(), now()),
(1, 'EQUITY', 'USD', 1059732496.00, 6, 2024, now(), now()),
(1, 'SHARE_CAPITAL', 'USD', 376778224.00, 6, 2024, now(), now()),
(1, 'RETAINED_EARNING', 'USD', 682954272.00, 6, 2024, now(), now()),
(1, 'TOTAL_LIABILITIES_AND_EQUITY', 'USD', 2163524692.00, 6, 2024, now(), now());
```

Penjelasan:

- project_id :
 - Semua data menggunakan project_id = 1 .
- Nilai Kosong:
 - Periode tanpa nilai, seperti Dec-2024 dan Jun-2025, tidak disertakan.
- metric :
 - Nilai enum dari Metric_Balance_Sheet .
- month_period dan year_period :
 - Menyesuaikan kolom berdasarkan data dari input Anda.

Jika ingin menambahkan periode tambahan seperti *Dec-2024* atau *Jun-2025*, tambahkan nilai `0.00` untuk setiap metrik yang tidak memiliki data.

```

    buatkan enum, untuk unitnya juga

    model periodic_report_technical_operation {
      id      Int    @id @default(autoincrement())
      project_id Int?
      metric   String? @db.VarChar(255)
      units    String? @db.VarChar(10)
      value    Decimal? @db.Decimal(20, 2)
      month_period Int?
      year_period Int?
      projects projects? @relation(fields: [project_id], references:
[id], onDelete: NoAction, onUpdate: NoAction)
    }

    enum metric_tecnical_operational {

    }
  
```

- Technical Operation - PLTU
- No Metric Unit
- 1 Energy Generated MWh
 - 2 Energy Sales MWh
 - 3 Coal Stock MT
 - 4 HOP Day
 - 5 CF Plan %
 - 6 CF %
 - 7 EAF %
 - 8 EFOR %
 - 9 SdOF kali/Unit
 - 10 NPHR Plan kCal/kWh
 - 11 NPHR kCal/kWh
 - 12 Afpm Declare %
 - 13 Afa %
 - 14 Loss of Production MWh

Berikut adalah enum untuk *metric* dan *unit* yang digunakan pada model `periodic_report_technical_operation` .

Enum untuk *Metric*

```

prisma

enum Metric_Technical_Operational {
  ENERGY_GENERATED
  ENERGY_SALES
  COAL_STOCK
  HOP
  CF_PLAN
}
  
```

```
CF
EAF
EFOR
SDOF
NPHR_PLAN
NPHR
AFPM_DECLARE
AFA
LOSS_OF_PRODUCTION
}
```

Enum untuk *Units*

```
prisma

enum Unit_Technical_Operational {
  MWH
  MT
  DAY
  PERCENT
  KALI_PER_UNIT
  KCAL_PER_KWH
}
```

Penyesuaian:

1. Format Penamaan:

- **Metric:** Menggunakan format **SCREAMING_SNAKE_CASE** untuk mengganti spasi atau simbol dengan `_`.
- **Unit:** Unit disingkat seperti pada tabel input (contoh: `kCal/kWh` menjadi `KCAL_PER_KWH`).

2. Deskripsi untuk Enum Units:

- `MWH` : Untuk metrik yang menggunakan MegaWatt Hour.
- `MT` : Untuk metrik yang menggunakan Metric Ton.
- `DAY` : Untuk HOP yang menggunakan satuan Hari.
- `PERCENT` : Untuk semua metrik berbasis persentase (%).
- `KALI_PER_UNIT` : Untuk *SdOF* yang menggunakan jumlah insiden per unit.
- `KCAL_PER_KWH` : Untuk metrik *NPHR Plan* dan *NPHR*.

Enum ini dapat digunakan untuk menjaga konsistensi dalam input dan penyimpanan data di database Anda.●