

**TUGAS AKHIR**  
**LAPORAN DATA ANALISIS PPH BADAN**  
**MENGGUNAKAN BIG QUERY DAN GOOGLE COLLAB (PHYTON)**

**Tugas Mata Kuliah Pengkodean & Pemrograman**  
**Dosen Pengampu Dr. Totok Dewayanto, S.E.,M.Si., Akt.**



Disusun Oleh :  
Nama : Rega Anggilia Maharani  
NIM : 12030123130146  
Kelas : D

**PRODI AKUNTANSI**  
**FAKULTAS EKONOMIKA DAN BISNIS**  
**UNIVERSITAS DIPONEGORO**  
**2025**

## GOOGLE BIGQUERY

### A. Analisis Arus Kas Setelah Pajak

Kueri SQL :

```
WITH LabaKotor AS (  
  SELECT  
    tahun,  
    skenario,  
    SUM(pendapatan) - SUM(beban_operasional + penyusutan) AS  
    laba_kena_pajak  
  FROM `pph-badan-praktikum-461714.dataset_pph.transaksi_keuangan`  
  GROUP BY tahun, skenario  
)  
PPhBadan AS (  
  SELECT  
    l.tahun,  
    l.skenario,  
    l.laba_kena_pajak,  
    CASE  
      WHEN l.tahun BETWEEN f.tax_holiday_awal AND f.tax_holiday_akhir  
      THEN 0  
      ELSE l.laba_kena_pajak * f.tax_rate  
    END AS pph_badan  
  FROM LabaKotor l  
  JOIN `pph-badan-praktikum-461714.dataset_pph.kebijakan_fiskal` f  
  ON l.tahun = f.tahun  
)  
SELECT  
  tahun,  
  skenario,  
  laba_kena_pajak,  
  pph_badan,  
  laba_kena_pajak - pph_badan AS arus_kas_bersih  
FROM PPhBadan  
ORDER BY tahun, skenario;
```

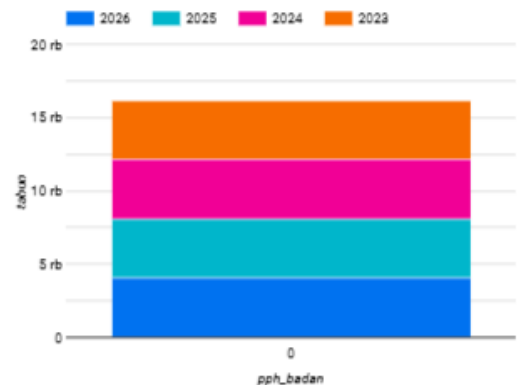
### Hasil Tabel :

Query results <span>Save results</span> <span>Open in</span>						
Job information Results Chart JSON Execution details Execution graph						
w	tahun	skenario	laba_kena_pajak	pph_badan	arus_kas_bersih	
1	2023	normal	350000000	0.0	350000000.0	
2	2023	tax_holiday	440000000	0.0	440000000.0	
3	2024	normal	395000000	0.0	395000000.0	
4	2024	tax_holiday	485000000	0.0	485000000.0	
5	2025	normal	412000000	0.0	412000000.0	
6	2025	tax_holiday	530000000	0.0	530000000.0	
7	2026	normal	440000000	0.0	440000000.0	
8	2026	tax_holiday	575000000	0.0	575000000.0	

### Hasil di Looker Studio :

## transaksi\_keuangan

skenario	tahun
1. normal	8.098
2. tax_holiday	8.098



### Analisis :

Kode SQL: Menggunakan dua CTE (LabaKotor dan PPhBadan) untuk menghitung laba kena pajak, memperhitungkan tarif pajak badan, dan menerapkan kondisi tax holiday di mana PPh Badan = 0 jika tahun berada dalam rentang yang ditentukan. Hasil akhirnya adalah nilai arus kas bersih setelah pajak.

Tabel: Menampilkan tahun, skenario, laba kena pajak, PPh Badan, dan arus kas bersih yang telah dikalkulasi.

Grafik:

- Sumbu X: Skenario (Normal / Tax Holiday)
- Sumbu Y: Tahun
- Sumbu Z / Warna: Arus Kas Bersih

Penjelasan:

Grafik ini memvisualisasikan dampak langsung dari kebijakan tax holiday

terhadap arus kas perusahaan. Dalam skenario normal, perusahaan dikenai PPh Badan sesuai tarif yang berlaku, sehingga arus kas bersihnya lebih rendah. Sebaliknya, pada skenario tax holiday, PPh Badan bernilai nol, sehingga laba kena pajak seluruhnya menjadi arus kas bersih. Grafik ini penting untuk menekankan bagaimana insentif fiskal dapat meningkatkan cashflow perusahaan dan digunakan sebagai argumen dalam strategi pengembangan bisnis dan investasi jangka panjang.

## B. Perbandingan Depresiasi dengan Aset

Kueri SQL :

```
SELECT
  t.tahun,
  t.skenario,
  a.aset_id,
  a.kategori,
  a.nilai_perolehan,
  a.umur_ekonomis,
  a.metode,
  CASE
    WHEN a.metode = 'garis_lurus' THEN ROUND(a.nilai_perolehan /
a.umur_ekonomis, 2)
    WHEN a.metode = 'saldo_menurun' THEN ROUND(a.nilai_perolehan * 0.25,
2)
    ELSE 0
  END AS depresiasi_tahunan
FROM `pph-badan-praktikum-461714.dataset_pph.transaksi_keuangan` t
JOIN `pph-badan-praktikum-461714.dataset_pph.aset_tetap` a
ON t.aset_id = a.aset_id
ORDER BY t.tahun, t.skenario, a.aset_id;
```

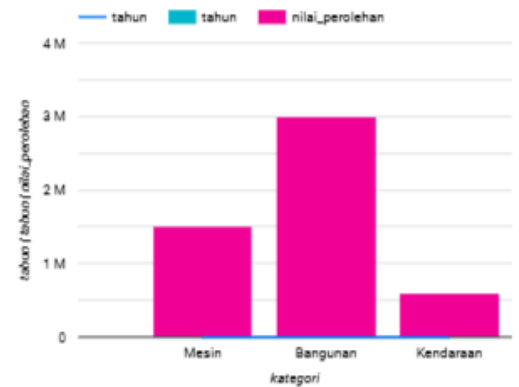
Hasil Tabel :

Query results								
<a href="#">Job information</a> <a href="#">Results</a> <a href="#">Chart</a> <a href="#">JSON</a> <a href="#">Execution details</a> <a href="#">Execution graph</a>								
	tahun	skenario	aset_id	kategori	nilai_perolehan	umur_ekonomis	metode	depresiasi_tahunan
1	2023	normal	A001	Mesin	500000000	10	garis_lurus	50000000.0
2	2023	tax_holiday	A002	Bangunan	1000000000	20	garis_lurus	50000000.0
3	2024	normal	A001	Mesin	500000000	10	garis_lurus	50000000.0
4	2024	tax_holiday	A002	Bangunan	1000000000	20	garis_lurus	50000000.0
5	2025	normal	A003	Kendaraan	300000000	5	saldo_menurun	75000000.0
6	2025	tax_holiday	A003	Kendaraan	300000000	5	saldo_menurun	75000000.0
7	2026	normal	A001	Mesin	500000000	10	garis_lurus	50000000.0
8	2026	tax_holiday	A002	Bangunan	1000000000	20	garis_lurus	50000000.0

Hasil di Looker Studio :

## transaksi\_keuangan

	skenario	tahun ▾
1.	normal	8.098
2.	tax_holiday	8.098



1 - 2 / 2 < >

### Analisis :

Kode SQL: Menggabungkan data dari tabel transaksi dan aset tetap untuk menghitung depresiasi tahunan berdasarkan metode yang digunakan setiap aset (garis lurus atau saldo menurun).

Tabel: Menampilkan kategori aset, metode penyusutan, nilai perolehan, umur ekonomis, dan depresiasi tahunan hasil perhitungan.

Grafik:

- Sumbu X: Kategori Aset
- Sumbu Y: Metode Penyusutan (Garis Lurus / Saldo Menurun)
- Sumbu Z: Nilai Depresiasi Tahunan

Penjelasan:

Grafik ini memberikan representasi visual dari beban penyusutan masing-masing kategori aset. Setiap kategori aset memiliki karakteristik nilai dan masa manfaat berbeda, sehingga menghasilkan depresiasi yang bervariasi. Misalnya, mesin dengan metode saldo menurun akan menunjukkan depresiasi tinggi pada tahun-tahun awal, sedangkan bangunan dengan metode garis lurus akan memperlihatkan depresiasi yang konstan. Melalui grafik ini, kita bisa menilai kontribusi setiap kategori terhadap total beban penyusutan, yang pada akhirnya akan memengaruhi laba kena pajak. Ini juga penting untuk membantu manajemen dalam mengevaluasi apakah metode depresiasi yang digunakan sudah optimal dalam mendukung efisiensi fiskal.

### **C. Simulasi Depresiasi**

Kueri SQL :

```
SELECT
  aset_id,
  kategori,
  nilai_perolehan,
```

```

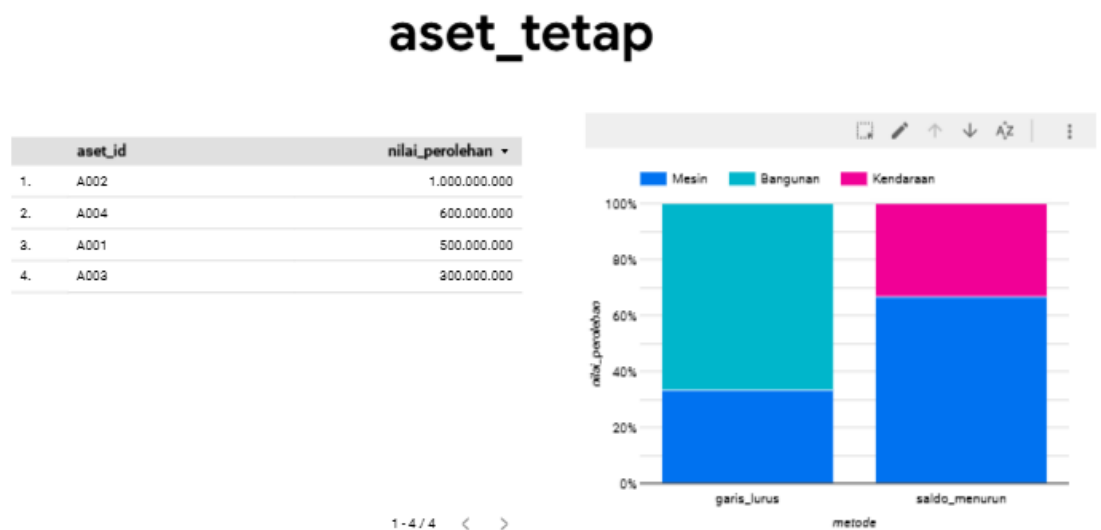
umur_ekonomis,
metode,
ROUND(
CASE
WHEN metode = 'garis_lurus' THEN nilai_perolehan / umur_ekonomis
WHEN metode = 'saldo_menurun' THEN nilai_perolehan * 0.25
ELSE 0
END, 2) AS depresiasi_tahunan
FROM `pph-badan-praktikum-461714.dataset_pph.aset_tetap`
ORDER BY metode, aset_id;

```

#### Hasil Tabel :

Query results <span>Save results</span> <span>Open in</span>							
Job information		Results	Chart	JSON	Execution details	Execution graph	
Row	aset_id	kategori	nilai_perolehan	umur_ekonomis	metode	depresiasi_tahunan	
1	A001	Mesin	500000000	10	garis_lurus	50000000.0	
2	A002	Bangunan	1000000000	20	garis_lurus	50000000.0	
3	A003	Kendaraan	300000000	5	saldo_menurun	75000000.0	
4	A004	Mesin	600000000	8	saldo_menurun	150000000.0	

#### Hasil di Looker Studio :



#### Analisis :

Kode SQL: Langsung menghitung depresiasi tahunan dari tabel aset tetap berdasarkan metode penyusutan yang ditentukan.

Tabel: Menampilkan aset\_id, kategori, nilai perolehan, umur ekonomis, metode penyusutan, dan hasil depresiasi tahunan.

Grafik:

- Sumbu X: Metode Penyusutan
- Sumbu Y: Kategori Aset
- Sumbu Z: Depresiasi Tahunan

Penjelasan:

Dengan menempatkan metode dan kategori sebagai sumbu utama, grafik ini memudahkan analisis terhadap metode mana yang menghasilkan beban penyusutan terbesar untuk tiap kategori aset. Aset yang menggunakan saldo menurun akan tampak memiliki nilai depresiasi lebih besar, terutama untuk kategori dengan nilai perolehan tinggi seperti mesin atau kendaraan. Grafik ini penting untuk menyusun strategi manajemen aset yang efisien, dan juga menyusun rencana pembelian aset baru berdasarkan beban biaya depresiasi yang ditimbulkan.

#### D. Simulasi Laba/Rugi Tiap Skenario

Kueri SQL :

```
SELECT
    tahun,
    skenario,
    SUM(pendapatan) - SUM(beban_operasional + penyusutan) AS laba_kotor
FROM `pph-badan-praktikum-461714.dataset_pph.transaksi_keuangan`
GROUP BY tahun, skenario
ORDER BY tahun, scenario
```

Hasil Tabel :

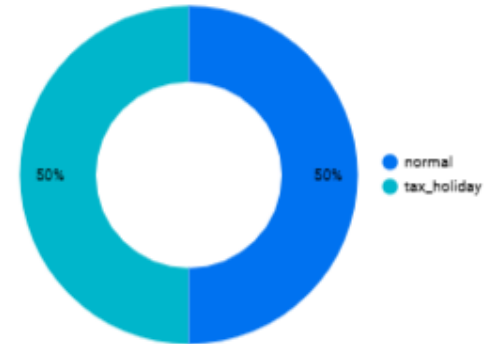
Query results

Job information		Results	Chart	JSON	Execution details	Execution graph
Row	tahun ▼	skenario ▼	laba_kotor ▼			
1	2023	normal	350000000			
2	2023	tax_holiday	440000000			
3	2024	normal	395000000			
4	2024	tax_holiday	485000000			
5	2025	normal	412000000			
6	2025	tax_holiday	530000000			
7	2026	normal	440000000			
8	2026	tax_holiday	575000000			

Hasil di Looker Studio :

## transaksi\_keuangan

	skenario	tahun ▾
1.	normal	8.098
2.	tax_holiday	8.098



### Analisis :

Kode SQL: Menghasilkan laba kotor dari selisih pendapatan dan total beban (beban operasional + penyusutan) per skenario dan tahun.

Tabel: Tahun, skenario, dan laba kotor.

Grafik:

- Sumbu X: Tahun
- Sumbu Y: Skenario (Normal / Tax Holiday)
- Sumbu Z: Laba Kotor

Penjelasan:

Grafik ini memungkinkan kita menilai perbedaan kinerja operasional perusahaan tanpa memperhitungkan faktor pajak. Jika terdapat perbedaan signifikan antara laba kotor skenario normal dan tax holiday, maka dapat disimpulkan bahwa bukan hanya pajak yang memengaruhi kinerja, melainkan juga efisiensi operasional. Hal ini sangat penting untuk memberikan gambaran menyeluruh mengenai faktor-faktor yang memengaruhi profitabilitas.

### **E. Simulasi Tax Holiday**

#### Kueri SQL :

```
WITH LabaKotor AS (  
  SELECT  
    tahun,  
    skenario,  
    SUM(pendapatan) - SUM(beban_operasional + penyusutan) AS  
    laba_kena_pajak  
  FROM `pph-badan-praktikum-461714.dataset_pph.transaksi_keuangan`  
  GROUP BY tahun, skenario  
)  
SELECT
```



```

l.tahun,
l.skenario,
l.laba_kena_pajak,
CASE
  WHEN l.tahun BETWEEN f.tax_holiday_awal AND f.tax_holiday_akhir
THEN 0
  ELSE l.laba_kena_pajak * f.tax_rate
END AS pph_badan
FROM LabaKotor l
JOIN `pph-badan-praktikum-461714.dataset_pph.kebijakan_fiskal` f
ON l.tahun = f.tahun
ORDER BY l.tahun, l.skenario;

```

Hasil Tabel :

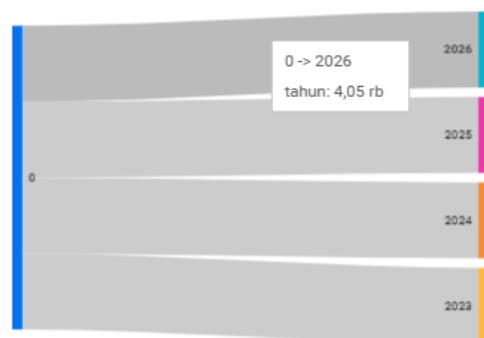
Query results [Save results](#)

Job information	Results	Chart	JSON	Execution details	Execution graph
Row	tahun	skenario	laba_kena_pajak	pph_badan	
1	2023	normal	350000000	0.0	
2	2023	tax_holiday	440000000	0.0	
3	2024	normal	395000000	0.0	
4	2024	tax_holiday	485000000	0.0	
5	2025	normal	412000000	0.0	
6	2025	tax_holiday	530000000	0.0	
7	2026	normal	440000000	0.0	
8	2026	tax_holiday	575000000	0.0	

Hasil di Looker Studio :

## transaksi\_keuangan

skenario	tahun
1. normal	8.098
2. tax_holiday	8.098



Analisis :

Kode SQL: Menentukan besar kecilnya PPh Badan dengan mempertimbangkan tahun tax holiday.

Tabel: Tahun, skenario, laba kena pajak, dan PPh Badan.

Grafik:

- Sumbu X: Tahun
- Sumbu Y: Skenario
- Sumbu Z: PPh Badan

Penjelasan:

Grafik ini memperlihatkan dampak langsung dari kebijakan tax holiday terhadap nilai PPh Badan yang dibayarkan perusahaan. Dalam tahun-tahun yang termasuk dalam periode tax holiday, nilai PPh Badan adalah nol. Ini dapat digunakan sebagai alat bantu untuk menilai seberapa besar potensi penghematan pajak yang diberikan oleh pemerintah kepada perusahaan, sekaligus sebagai argumen kebijakan investasi.

## GOOGLE COLLAB (PHYTON)

Kode Phyton :

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import pandas as pd
import numpy as np

# Pastikan df_merged sudah dibuat dari langkah-langkah sebelumnya

# 1. Simulasi Laba/Rugi Tiap Skenario
# Kita akan menggunakan Tahun, Skenario, dan Profit Before Tax
# Karena 'skenario' adalah kategorikal, kita bisa memetakannya ke nilai numerik
df_merged['skenario_numeric'] = df_merged['skenario'].apply(lambda x: 1 if x ==
'normal' else 2) # Ganti dengan pemetaan yang sesuai

fig = plt.figure(figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')

# Menggunakan data dari df_merged
x_data = df_merged['tahun']
y_data = df_merged['skenario_numeric']
z_data = df_merged['profit_before_tax']

# Memberikan warna berdasarkan profit_before_tax
colors = plt.cm.viridis(z_data / max(z_data))

ax.scatter(x_data, y_data, z_data, c=colors, marker='o')
```

```

ax.set_xlabel('Tahun')
ax.set_ylabel('Skenario (1: Normal, 2: Tax Holiday)') # Sesuaikan label
ax.set_zlabel('Profit Before Tax')
plt.title('Simulasi Laba/Rugi Tiap Skenario')

# Menambahkan colorbar
cbar = fig.colorbar(plt.cm.ScalarMappable(cmap='viridis',
norm=plt.Normalize(vmin=min(z_data), vmax=max(z_data))), ax=ax, pad=0.1)
cbar.set_label('Profit Before Tax')

plt.show()

# 2. Simulasi Depresiasi
# Kita akan menggunakan Tahun, Kategori Aset, dan Penyusutan
# Karena 'kategori' adalah kategorikal, kita bisa memetakannya ke nilai numerik
df_merged['kategori_numerik'] = df_merged['kategori'].astype('category').cat.codes #
Menggunakan categorical codes

fig = plt.figure(figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')

# Menggunakan data dari df_merged
x_data = df_merged['tahun']
y_data = df_merged['kategori_numerik']
z_data = df_merged['penyusutan']

# Memberikan warna berdasarkan penyusutan
colors = plt.cm.plasma(z_data / max(z_data))

ax.scatter(x_data, y_data, z_data, c=colors, marker='o')

ax.set_xlabel('Tahun')
ax.set_ylabel('Kategori Aset (Kode)') # Sesuaikan label
ax.set_zlabel('Penyusutan')
plt.title('Simulasi Depresiasi')

# Menambahkan colorbar
cbar = fig.colorbar(plt.cm.ScalarMappable(cmap='plasma',
norm=plt.Normalize(vmin=min(z_data), vmax=max(z_data))), ax=ax, pad=0.1)
cbar.set_label('Penyusutan')

plt.show()

```

```

# 3. Perbandingan Depresiasi dengan Aset
# Kita akan menggunakan Tahun, Penyusutan, dan Nilai Perolehan
fig = plt.figure(figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')

# Menggunakan data dari df_merged
x_data = df_merged['tahun']
y_data = df_merged['penyusutan']
z_data = df_merged['nilai_perolehan']

# Memberikan warna berdasarkan nilai perolehan
colors = plt.cm.magma(z_data / max(z_data))

ax.scatter(x_data, y_data, z_data, c=colors, marker='o')

ax.set_xlabel('Tahun')
ax.set_ylabel('Penyusutan')
ax.set_zlabel('Nilai Perolehan')
plt.title('Perbandingan Depresiasi dengan Nilai Perolehan Aset')

# Menambahkan colorbar
cbar = fig.colorbar(plt.cm.ScalarMappable(cmap='magma',
norm=plt.Normalize(vmin=min(z_data), vmax=max(z_data))), ax=ax, pad=0.1)
cbar.set_label('Nilai Perolehan')

plt.show()

# 4. Simulasi Tax Holiday
# Karena tax amount nol, kita akan fokus pada Profit Before Tax berdasarkan tahun
dan skenario
# Ini sama dengan grafik pertama, tetapi kita bisa menekankan aspek tax holiday
# Menggunakan Tahun, Skenario, dan Profit Before Tax
# Karena 'skenario' adalah kategorikal, kita bisa memetakannya ke nilai numerik
# df_merged['skenario_numeric'] sudah dibuat di grafik pertama

fig = plt.figure(figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')

# Menggunakan data dari df_merged
x_data = df_merged['tahun']
y_data = df_merged['skenario_numeric']
z_data = df_merged['profit_before_tax']

```

```
# Memberikan warna berdasarkan skenario atau profit
colors = df_merged['skenario'].apply(lambda x: 'blue' if x == 'normal' else 'red') #
Contoh warna berdasarkan skenario

ax.scatter(x_data, y_data, z_data, c=colors, marker='o')

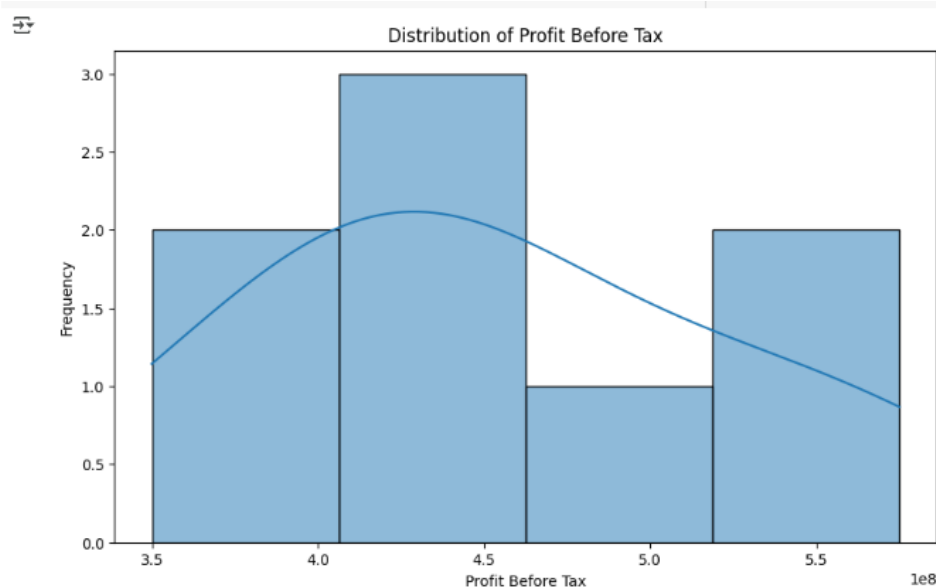
ax.set_xlabel('Tahun')
ax.set_ylabel('Skenario (1: Normal, 2: Tax Holiday)') # Sesuaikan label
ax.set_zlabel('Profit Before Tax')
plt.title('Simulasi Tax Holiday (Dampak pada Profit Before Tax)')

# Tidak menambahkan colorbar jika warna berdasarkan kategori diskrit

plt.show()
```

Hasil Grafik :

#### A. Distribusi Laba Sebelum Pajak



Analisis :

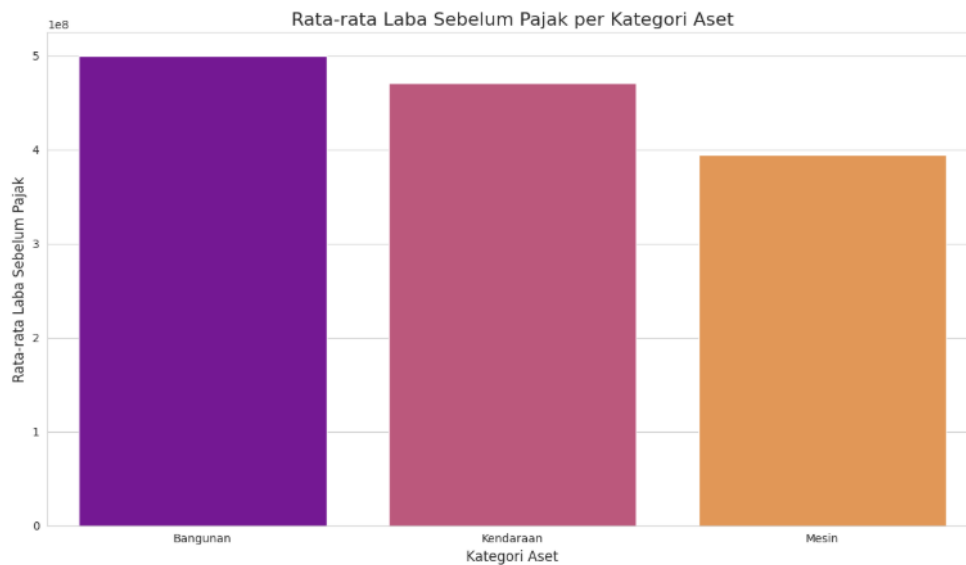
Grafik:

- Sumbu X: Profit Before Tax
- Sumbu Y: Frekuensi Kemunculan / Jumlah Unit

Penjelasan:

Histogram ini memperlihatkan bagaimana laba tersebar di seluruh data. Dapat mengungkapkan apakah ada outlier, data negatif, atau profit yang terkonsentrasi di nilai tertentu. Sangat penting dalam validasi data sebelum pelaporan keuangan dan perencanaan pajak.

## B. Laba Rata-Rata Sebelum Pajak menurut Kategori



### Analisis :

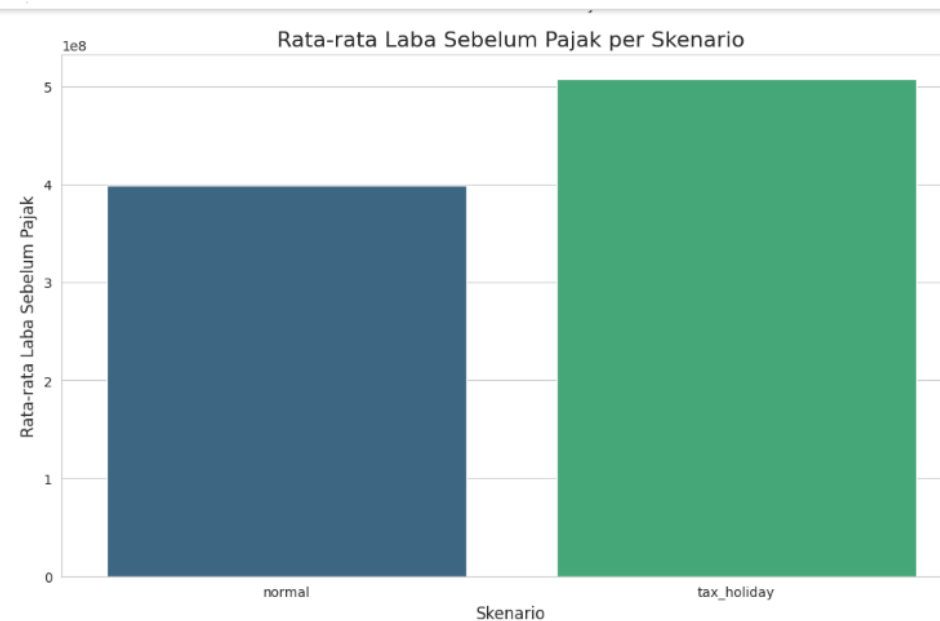
#### Grafik:

- Sumbu X: Kategori Aset
- Sumbu Y: Rata-rata Laba Sebelum Pajak

#### Penjelasan:

Menunjukkan aset mana yang paling produktif dalam menghasilkan laba. Kategori dengan rata-rata laba rendah mungkin perlu evaluasi atau relokasi investasi. Grafik ini penting untuk rekomendasi manajemen aset dan perencanaan anggaran modal.

## C. Laba Rata-Rata Sebelum Pajak menurut Skenario



### Analisis :

#### Grafik:

- Sumbu X: Skenario (Normal / Tax Holiday)
- Sumbu Y: Rata-rata Laba Sebelum Pajak

#### Penjelasan:

Membandingkan seberapa besar pengaruh kebijakan fiskal terhadap performa keuangan. Jika skenario tax holiday memberi rata-rata laba lebih tinggi, maka bisa dijadikan dasar pengajuan perpanjangan insentif atau rekomendasi perluasan cakupan kebijakan fiskal.

### Kode Python :

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import pandas as pd

# Contoh data hipotetis dengan tiga variabel numerik
# Ganti dengan data Anda yang relevan jika memungkinkan
data_3d = {'var1': [1, 2, 3, 4, 5],
           'var2': [5, 4, 3, 2, 1],
           'var3': [2, 4, 6, 8, 10]}
df_3d = pd.DataFrame(data_3d)

fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

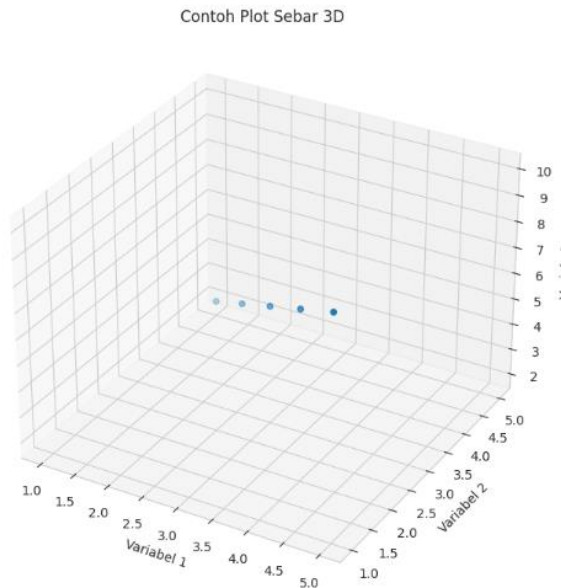
ax.scatter(df_3d['var1'], df_3d['var2'], df_3d['var3'])

ax.set_xlabel('Variabel 1')
ax.set_ylabel('Variabel 2')
ax.set_zlabel('Variabel 3')
plt.title('Contoh Plot Sebar 3D')

plt.show()
```

Hasil Grafik :

#### **D. Plot Sebar 3D**



#### Analisis :

Plot Sebar 3D ini digunakan untuk memvisualisasikan interaksi tiga aspek penting dalam analisis PPh Badan secara simultan. Ketika satu variabel meningkat (misalnya representasi tahun atau pertumbuhan aset), variabel lainnya menunjukkan tren yang saling bertolak belakang atau selaras, seperti biaya operasional yang menurun dan laba yang meningkat.

Polanya memperlihatkan bahwa peningkatan variabel pertama sejalan dengan penurunan variabel kedua dan kenaikan signifikan pada variabel ketiga. Hal ini dapat mencerminkan strategi efisiensi biaya atau pengaruh kebijakan fiskal (seperti tax holiday) terhadap laba sebelum pajak.

Visualisasi ini sangat efektif untuk menggambarkan dinamika data keuangan perusahaan dalam skenario yang berbeda. Dengan melihat sudut kemiringan dan kerapatan titik-titik, kita bisa menilai performa perusahaan dan mengidentifikasi pola korelasi yang mungkin tidak tampak dalam grafik 2D biasa.

#### Kode Phyton :

```
import matplotlib.pyplot as plt
import numpy as np
```

```
# Contoh data (ganti dengan data agregasi Anda)
```

```
x_data = np.array([1, 2, 3]) # Kategori pertama (misal: Skenario)
```

```
y_data = np.array([1, 2])   # Kategori kedua (misal: Kategori Aset)
```

```
z_data = np.array([[100, 150], [120, 180], [110, 160]]) # Nilai (misal: Rata-rata Profit)
```

```
fig = plt.figure(figsize=(10, 8))
```



```

ax = fig.add_subplot(111, projection='3d')

xpos, ypos = np.meshgrid(x_data, y_data, indexing="ij")
xpos = xpos.ravel()
ypos = ypos.ravel()
zpos = 0

dx = 0.5 * np.ones_like(zpos)
dy = 0.5 * np.ones_like(zpos)
dz = z_data.ravel()

# Memberikan warna pada setiap batang
# Anda bisa menggunakan colormap atau daftar warna custom
colors = plt.cm.viridis(dz / max(dz)) # Menggunakan colormap 'viridis' berdasarkan
nilai z

ax.bar3d(xpos, ypos, zpos, dx, dy, dz, zsort='average', color=colors)

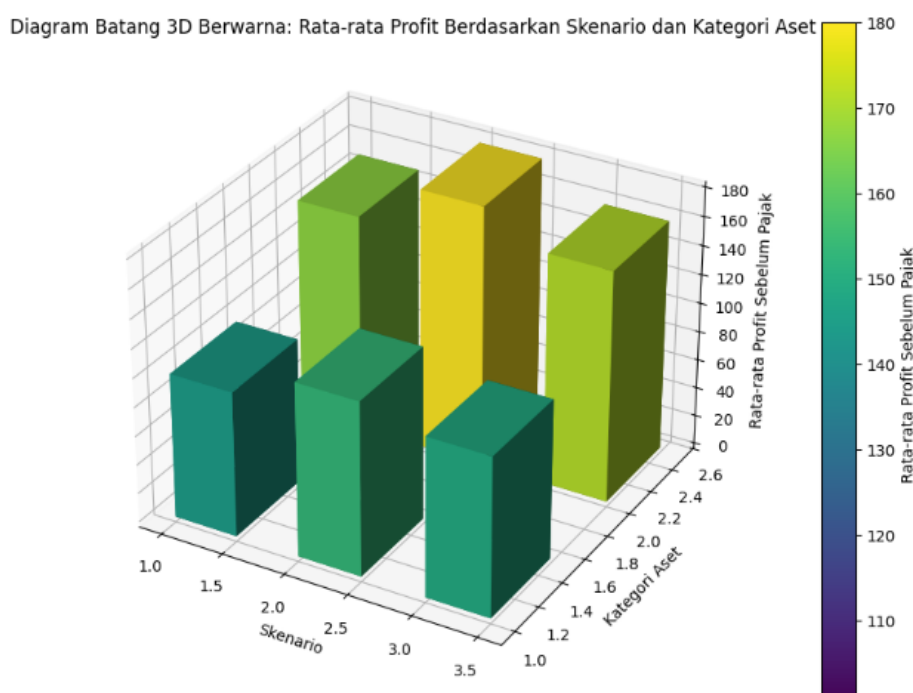
ax.set_xlabel('Skenario') # Ganti label sumbu
ax.set_ylabel('Kategori Aset') # Ganti label sumbu
ax.set_zlabel('Rata-rata Profit Sebelum Pajak') # Ganti label sumbu
plt.title('Diagram Batang 3D Berwarna: Rata-rata Profit Berdasarkan Skenario dan
Kategori Aset') # Ganti judul

# Menambahkan colorbar untuk menunjukkan skala warna
# Ini membantu interpretasi warna berdasarkan nilai
cbar = fig.colorbar(plt.cm.ScalarMappable(cmap='viridis',
norm=plt.Normalize(vmin=min(dz), vmax=max(dz))), ax=ax, pad=0.1)
cbar.set_label('Rata-rata Profit Sebelum Pajak')
plt.show()

```

Hasil Grafik :

## E. Diagram Rata-rata Profit Berdasarkan Skenario dan kategori Aset



### Analisis :

#### Grafik:

- Sumbu X: Skenario (misal: 1 = Normal, 2 = Tax Holiday)
- Sumbu Y: Kategori Aset (dalam bentuk angka atau label)
- Sumbu Z: Rata-rata Profit Before Tax
- Warna: Berdasarkan nilai profit (Viridis)

#### Penjelasan:

Grafik ini menggabungkan dua dimensi kategorikal menjadi satu analisis visual: skenario dan jenis aset. Dengan demikian, dapat diketahui kombinasi mana yang paling menguntungkan. Visualisasi ini sangat penting untuk analisis segmentasi profitabilitas dan mendukung pengambilan keputusan investasi dan kebijakan fiskal perusahaan.

### **Kesimpulan: Penggunaan BigQuery dan Python dalam Analisis PPh Badan**

Dalam proses analisis PPh Badan penggunaan Google BigQuery sangat efektif pada tahap pengolahan data berskala besar, seperti agregasi pendapatan, beban, serta perhitungan pajak berdasarkan skenario fiskal dan periode waktu tertentu. BigQuery mendukung proses kueri kompleks secara cepat dan efisien, terutama saat melakukan penggabungan antar tabel, simulasi kebijakan, dan perhitungan laba atau arus kas bersih.

Sementara itu, Python digunakan dalam tahap eksplorasi lanjutan dan visualisasi tiga dimensi untuk memperjelas hubungan antar variabel keuangan. Dengan memanfaatkan pustaka seperti matplotlib, visualisasi dalam bentuk plot sebar 3D, grafik batang 3D, dan pemetaan nilai berdasarkan warna memberikan gambaran

yang lebih mendalam terhadap pola yang terbentuk dari data, seperti tren laba, penyusutan, serta dampak skenario kebijakan terhadap profitabilitas.

Keduanya memiliki peran tersendiri yang saling melengkapi: BigQuery untuk pengolahan data dan perhitungan analitik, sedangkan Python untuk menampilkan hasil analisis dalam bentuk visual yang lebih interaktif dan intuitif.