<https://github.com/Nterah/ProVisio>

git remote add origin <https://github.com/Nterah/ProVisio>.git

git branch -M main

git push -u origin main

afGaHjyG7eYnJzIA

Supabase

Project name : Nterah-prod

Project ID : fuvzhbuvwpnysluojqni

Password : OLUULO050911141824

API Keys

Anon/public : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6ImZ1dnpoYnV2d3BueXNsdW9qcW5pIiwicm9sZSI6ImFub24iLCJpYXQiOjE3NTQzNjYxMzAsImV4cCI6MjA2OTk0MjEzMH0.HIDTkfm\_glqEQG3JFXqmw6QHRJ6W57hA10TseweH5PA

Service\_role (secrete) : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6ImZ1dnpoYnV2d3BueXNsdW9qcW5pIiwicm9sZSI6InNlcnZpY2Vfcm9sZSIsImlhdCI6MTc1NDM2NjEzMSwiZXhwIjoyMDY5OTQyMTMxfQ.eW9RdbPa2QXmVmGQNvXxlWuV4XdKKanCcOtLrqN0MqI

$env:PGSSLMODE='require'

$env:PGPASSWORD='OLUULO050911141824'

& "C:\Program Files\PostgreSQL\17\bin\psql.exe" `

-h db.fuvzhbuvwpnysluojqni.supabase.co `

-p 5432 -U postgres -d postgres `

-c "select current\_user, now();"

# 1) set creds for this session

$env:PGSSLMODE = 'require'

$env:PGPASSWORD = 'OLUULO050911141824'

# 2) run the test query (use your existing psql path)

& "C:\Program Files\PostgreSQL\17\bin\psql.exe" `

-h db.fuvzhbuvwpnysluojqni.supabase.co `

-p 5432 -U postgres -d postgres `

-c "select current\_user as user, inet\_server\_addr() as host, now();"

# 3) remove env vars when done

Remove-Item Env:\PGPASSWORD; Remove-Item Env:\PGSSLMODE

$env:PGSSLMODE = 'require'

$env:PGPASSWORD = 'OLUULO050911141824' # from Supabase -> Database -> Reset password

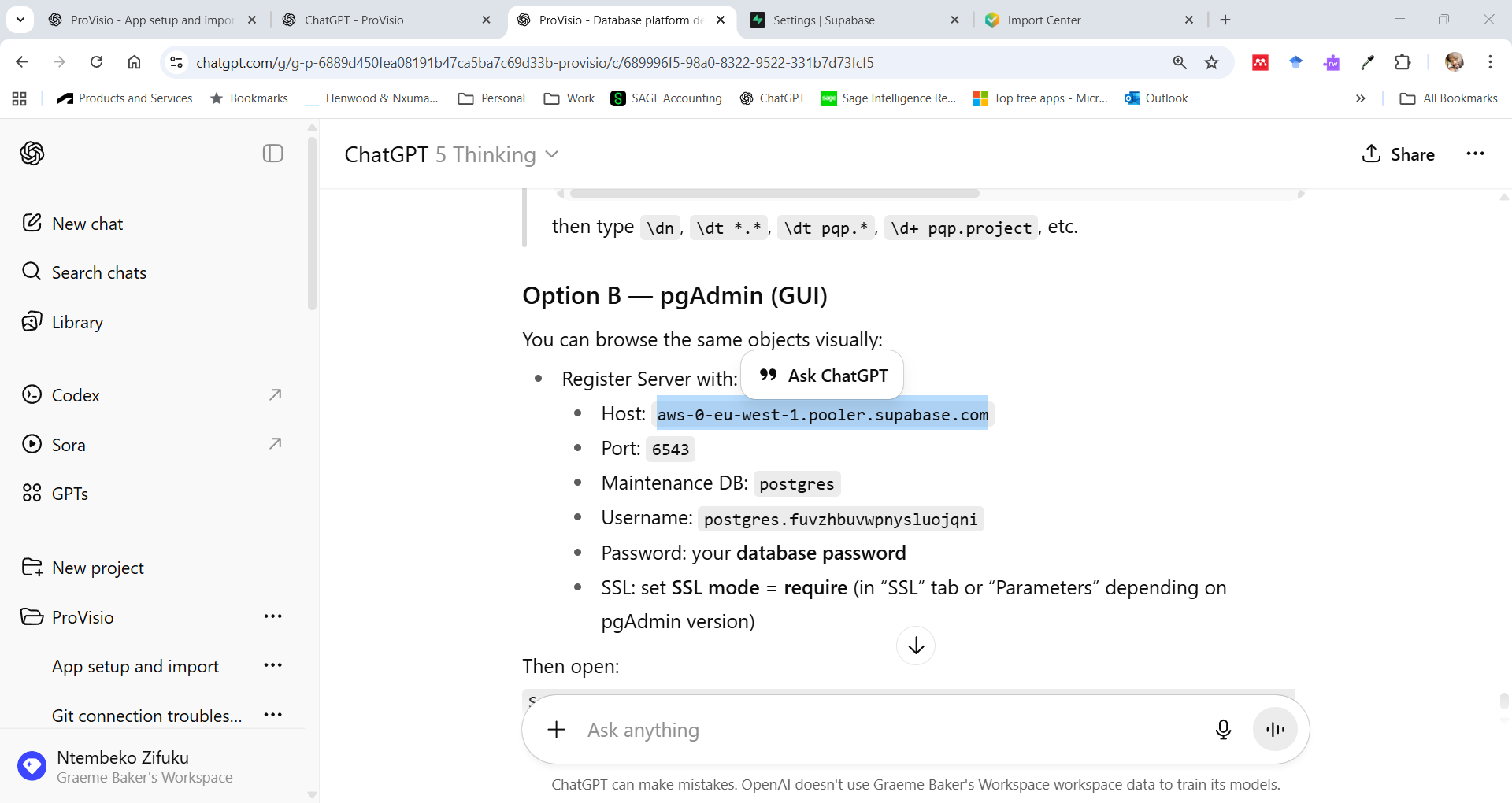
& "C:\Program Files\PostgreSQL\17\bin\psql.exe" `

-h aws-0-eu-west-1.pooler.supabase.com `

-p 6543 -U postgres.fuvzhbuvwpnysluojqni -d postgres `

-c "select current\_user, current\_database(), now();"

Remove-Item Env:\PGPASSWORD; Remove-Item Env:\PGSSLMODE



$env:PGSSLMODE = "require"; $env:PGPASSWORD = "OLUULO050911141824";

& "C:\Program Files\PostgreSQL\17\bin\psql.exe" `

-h aws-0-eu-west-1.pooler.supabase.com -p 6543 `

-U postgres.fuvzhbuvwpnysluojqni -d postgres `

-c "\copy stg.excel\_rows (source\_file,sheet\_name,row\_number,row)

FROM 'C:/Users/Ntembeko Zifuku/Downloads/stg\_excel\_rows.csv'

WITH (FORMAT csv, HEADER true);"

"C:\HN\HN\Storage - Admin\99 Temp\pqp\Merged PQP data.xlsm"

# ---- Set these two lines using Copy as path ----

$ExcelPath = "C:\HN\HN\Storage - Admin\99 Temp\pqp\Merged PQP data.xlsm"

$OutDir = "C:\HN\HN\Storage - Admin\99 Temp\pqp\_csv"

# -----------------------------------------------

New-Item -ItemType Directory -Force -Path $OutDir | Out-Null

function Sanitize([string]$n) { # "Section3\_1" -> "section31"

($n.ToLower() -replace '[^a-z0-9]','')

}

# Open Excel and the workbook

$excel = $null; $wb = $null

try {

$excel = New-Object -ComObject Excel.Application

$excel.Visible = $false; $excel.DisplayAlerts = $false

if (-not (Test-Path -LiteralPath $ExcelPath)) {

throw "File not found: $ExcelPath"

}

$wb = $excel.Workbooks.Open($ExcelPath)

foreach ($ws in $wb.Worksheets) {

$name = (Sanitize $ws.Name)

if ([string]::IsNullOrWhiteSpace($name)) { continue }

$csv = Join-Path $OutDir "$name.csv"

if (Test-Path $csv) { Remove-Item $csv -Force }

$ws.SaveAs($csv, 6) # 6 = CSV

}

}

finally {

if ($wb) { $wb.Close($false) }

if ($excel) { $excel.Quit(); [void][Runtime.Interopservices.Marshal]::ReleaseComObject($excel) }

}

Write-Host "CSV files at $OutDir"

UIID Id : fad7d963-e080-4583-9b56-9b6571e0e541

Name : Nterah