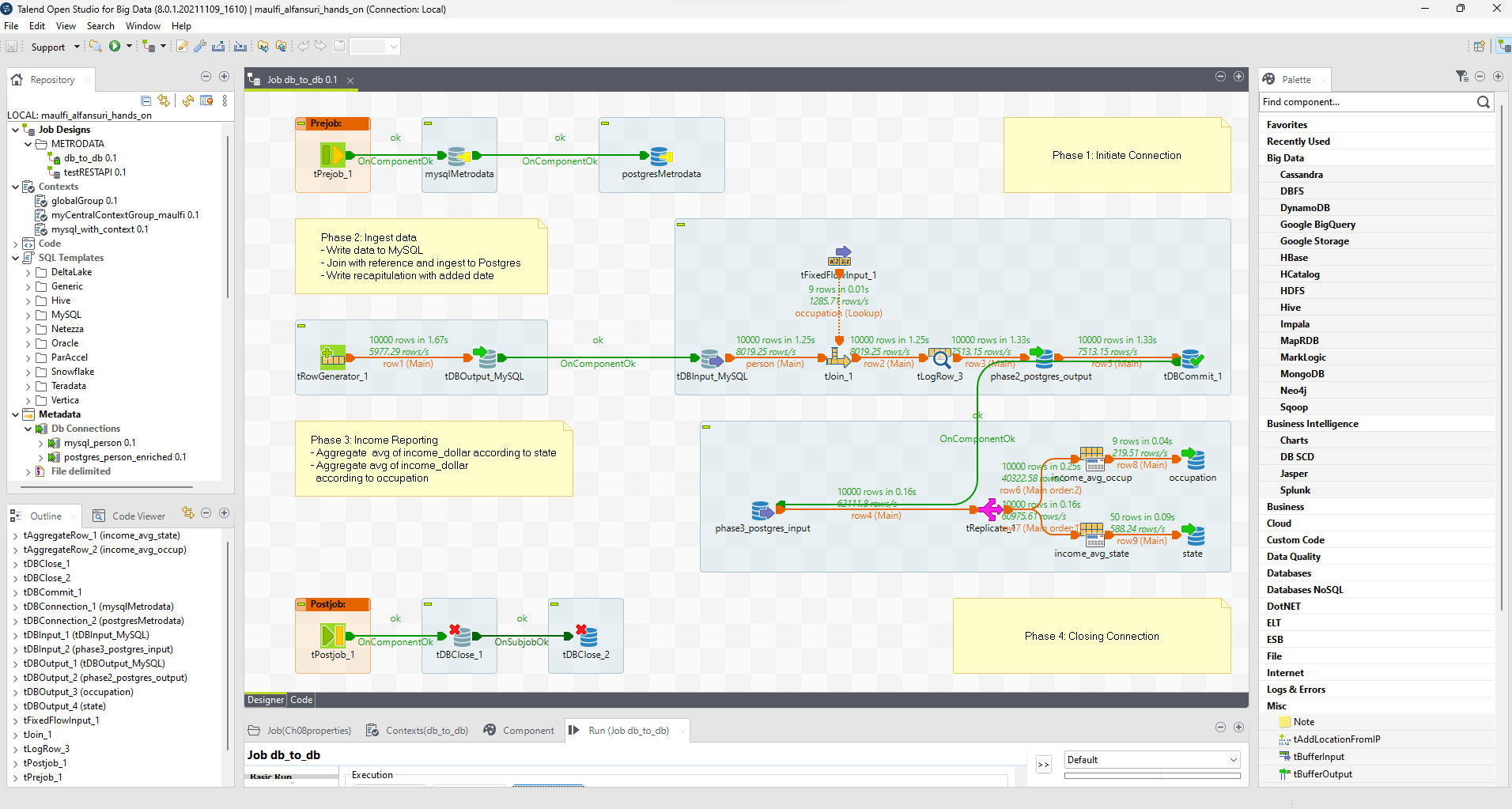
**TALEND HANDS ON**

**MAULFI ALFANSURI**

1. **Overview**



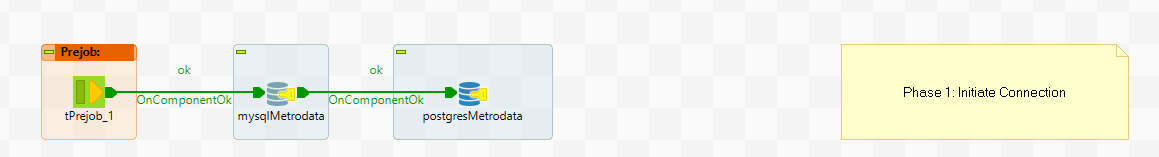
Pada kesempatan ini, akan didesain sebuah *use case* ETL terhadap 10 ribu data penduduk amerika serikta yang memuat *id*(kode unik), *lastName*(nama akhir), *state*(negara bagian), *income*(pendapatan) dan *occupation*(profesi).

ETL *pipeline* yang dikembangkan ditujukan untuk mengambil data agregasi nilai rata rata pendapatan dari 2 sisi -baik dari negara bagian maupun profesi masing masing-, yang masing masing termuat kedalam dua tabel di postgresql (dari container Docker) yaitu INCOME\_AVG\_STATE dan INCOME\_AVG\_OCCUPATION.

Terdapat 4 tahap utama yang dilalui diantaranya adalah:

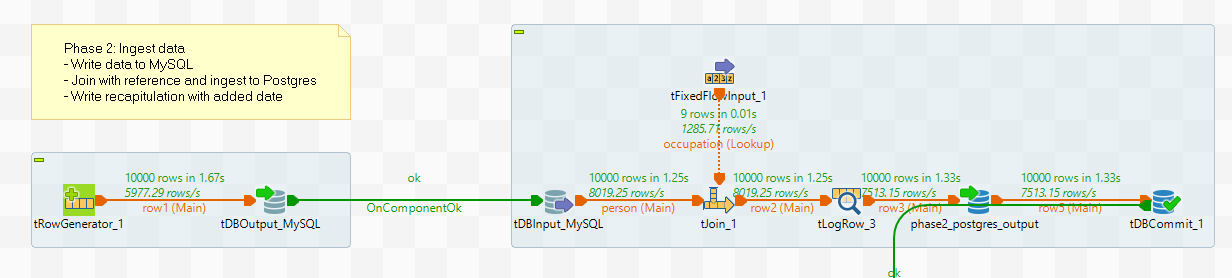
* Phase 1: Inisiasi koneksi terhadap DB MySQL dan PostgreSQL
* Phase 2: Manufaktur data dan *ingestion* kedalam MySQL, transformasi join dan *ingestion* ke dalam PostgreSQL
* Phase 3: Transformasi aggregasi terhadap tabel fase 2 postgres kedalam 2 tabel income\_avg\_state dan income\_avg\_occupation
* Phase 4: Penutupan koneksi DB

1. **Komponen *Job*:**



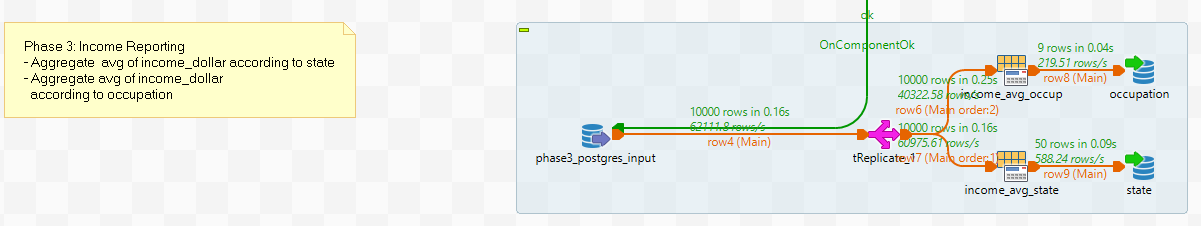
Komponen fase 1:

* tPreJob
* tDBConnection(mysqlMetrodata & postgresqlMetrodata): Untuk inisiasi koneksi MySQL



Komponen fase 2:

* tRowGenerator: Untuk manufaktur data
* tDBOutput\_MySQL: Untuk penulisan data kedalam tabel MySQL
* tDBInput\_MySQL: Untuk pengambilan data dari tabel MySQL
* tFixedFlowInput: Untuk data referensi *occupation code* – *occupation*
* tJoin\_1: Untuk transformasi join.
* tLogRow\_3: Untuk mendapatkan hasil sebelum penulisan kedalam postgres
* tDBCommit\_1: Untuk melakukan *commit* terhadap transaksi yang dibikin



Komponen fase 3:

* phase3\_postgres\_input: Untuk mengambil data dari postgres (tDBInput)
* tReplicate: Untuk memungkinkan agregasi secara simultan kedalam beberapa tabel berbeda
* ‘occupation’ & ‘state’: Untuk penulisan tabel aggregasi kedalam postgres

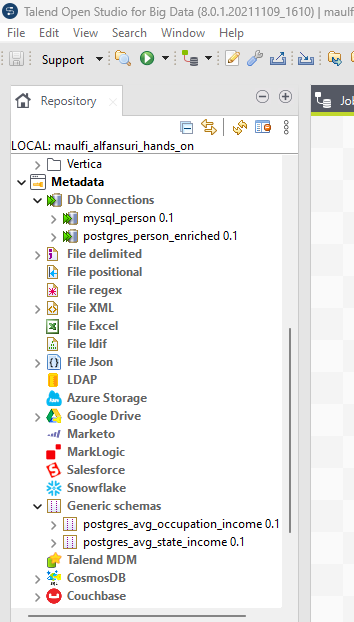


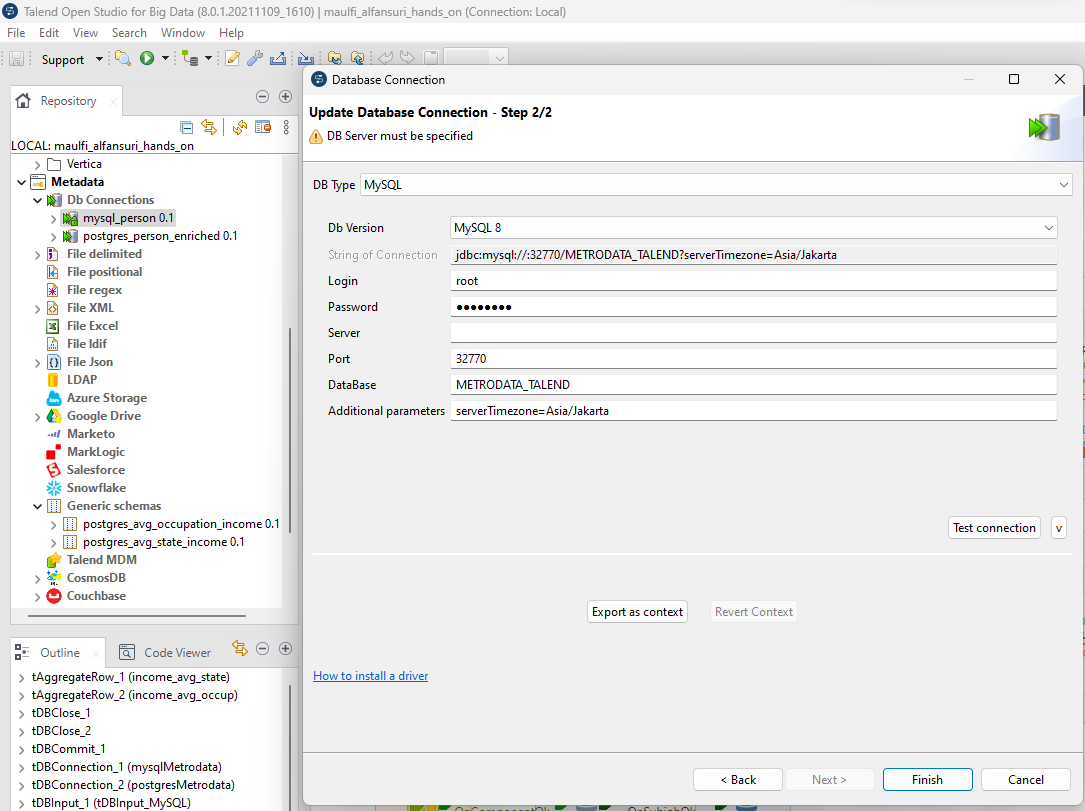
Komponen fase 4:

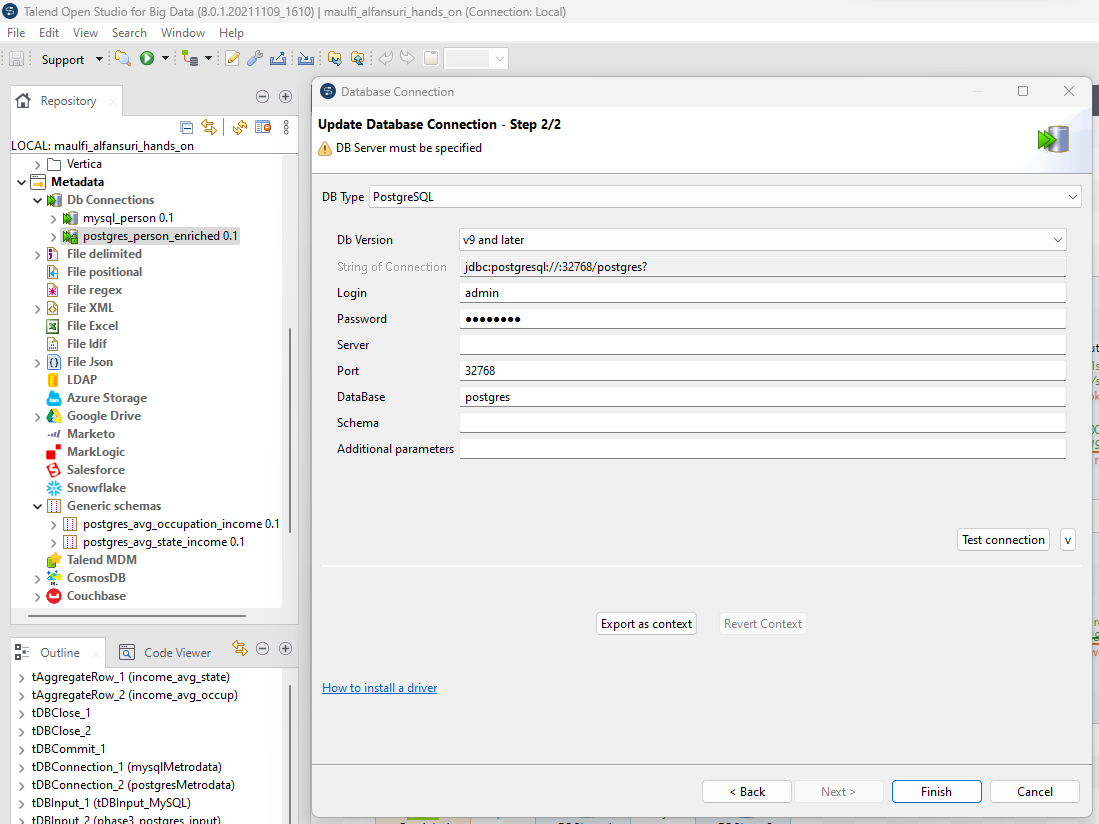
* tPostJob: Untuk finalisasi paska selesainya pekerjaan (*job*)
* tDBClose\_1 & tDBClose\_2: Untuk menutup koneksi ke 2 DB

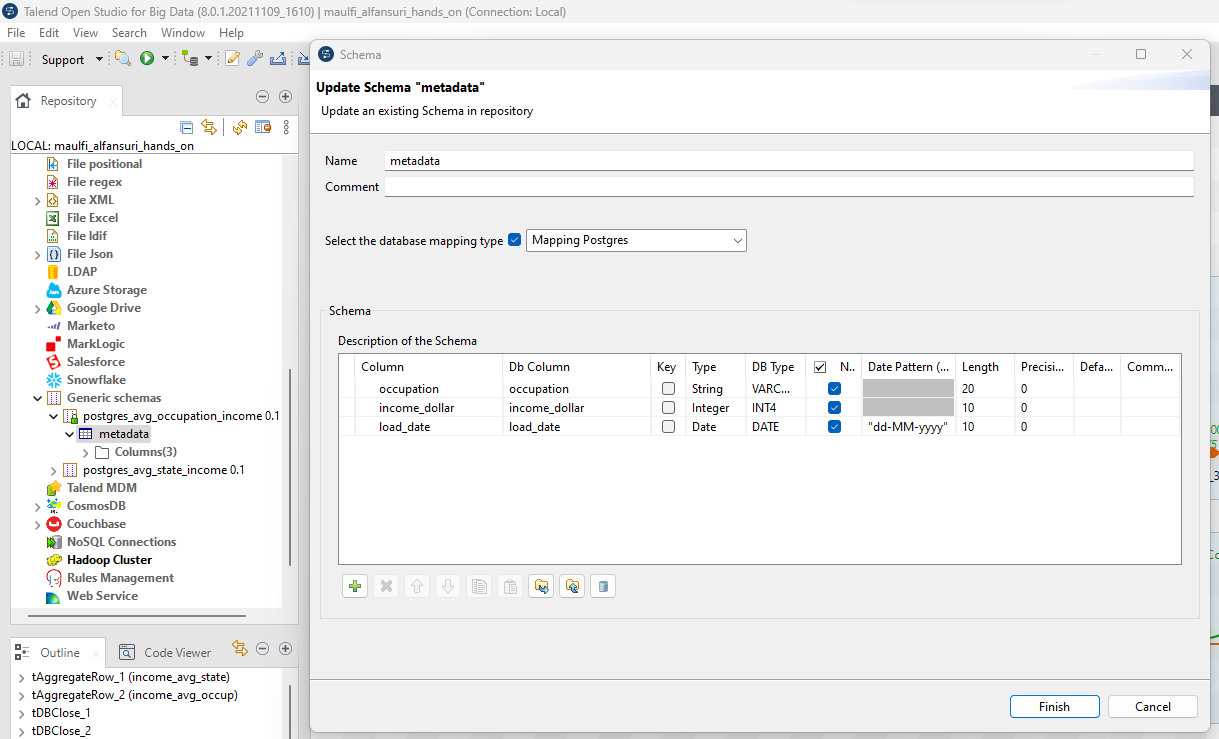
1. **Komponen metadata:**

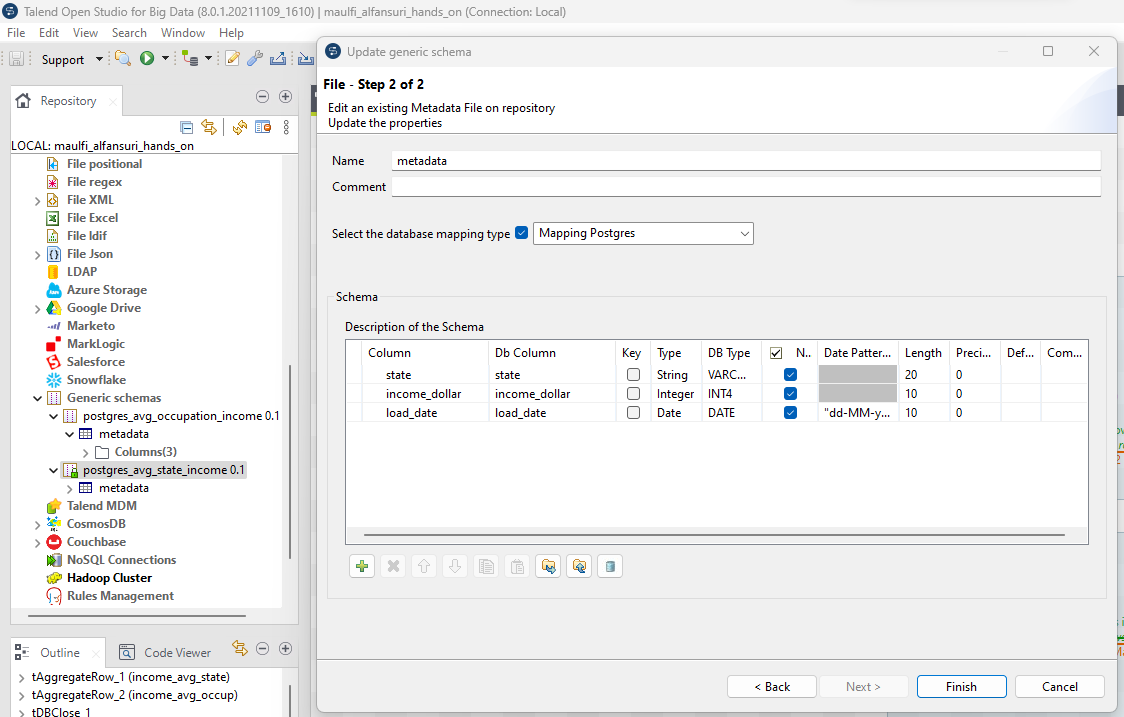
Beberapa konfigurasi seperti koneksi dan schema ditaruh kedalam metadata untuk memudahkan pembuatan ulang komponen dan meminimalisasi *built-in*. Ada 2 metadata koneksi untuk 2 tabel awal -tabel person di MySQL dan tabel person\_enriched di PostgreSQL-. Dan 2 metadata schema untuk 2 tabel akhir -tabel income\_avg\_occupation dan income\_avg\_state.





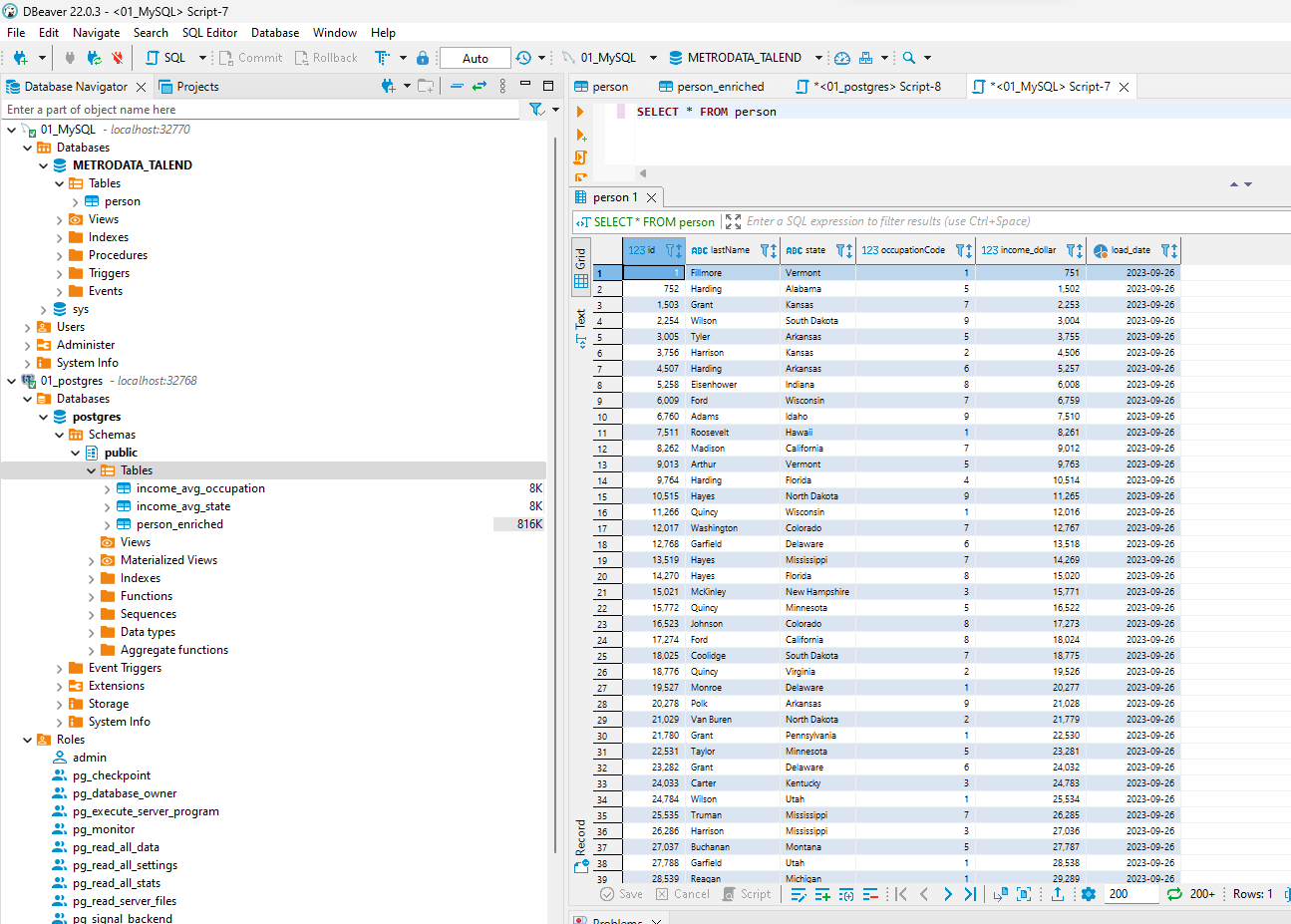




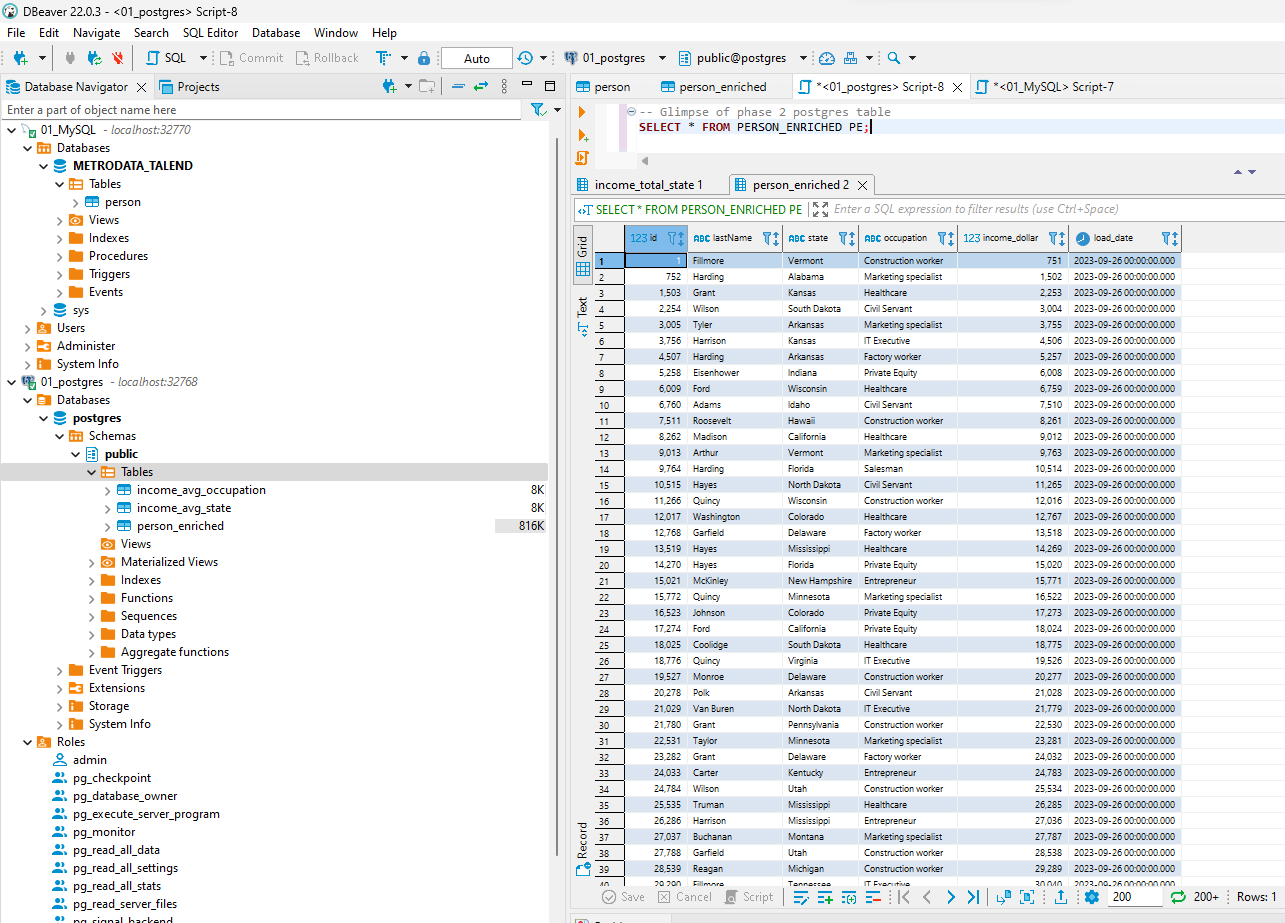


1. **Tables Overview:**

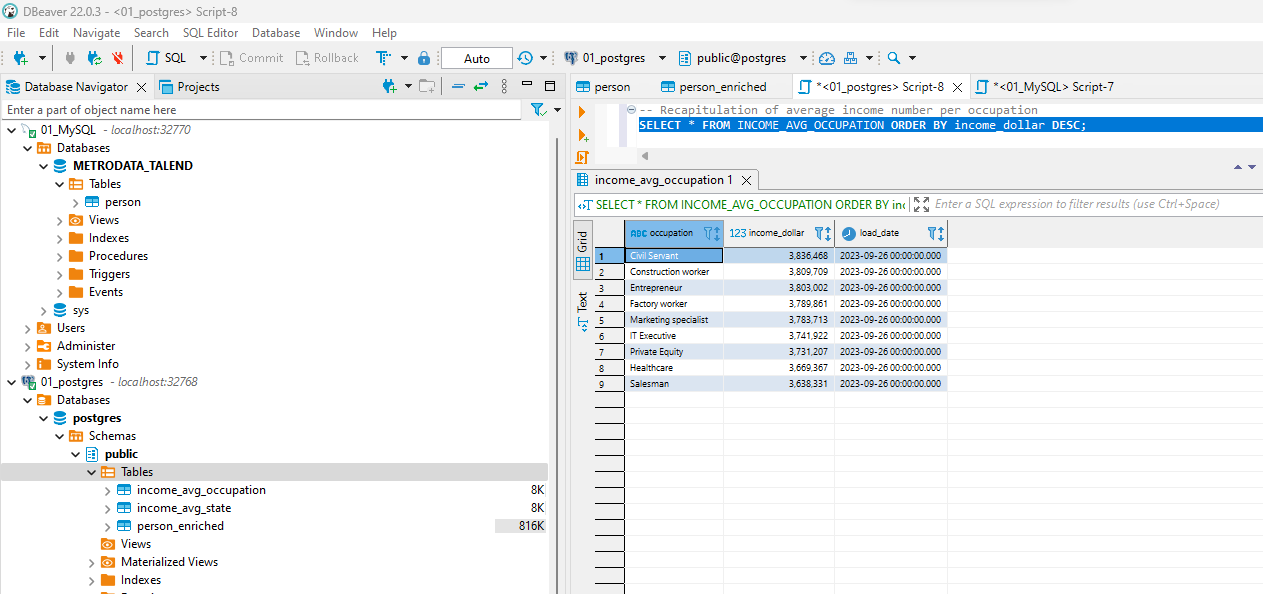
* **person Table**

****

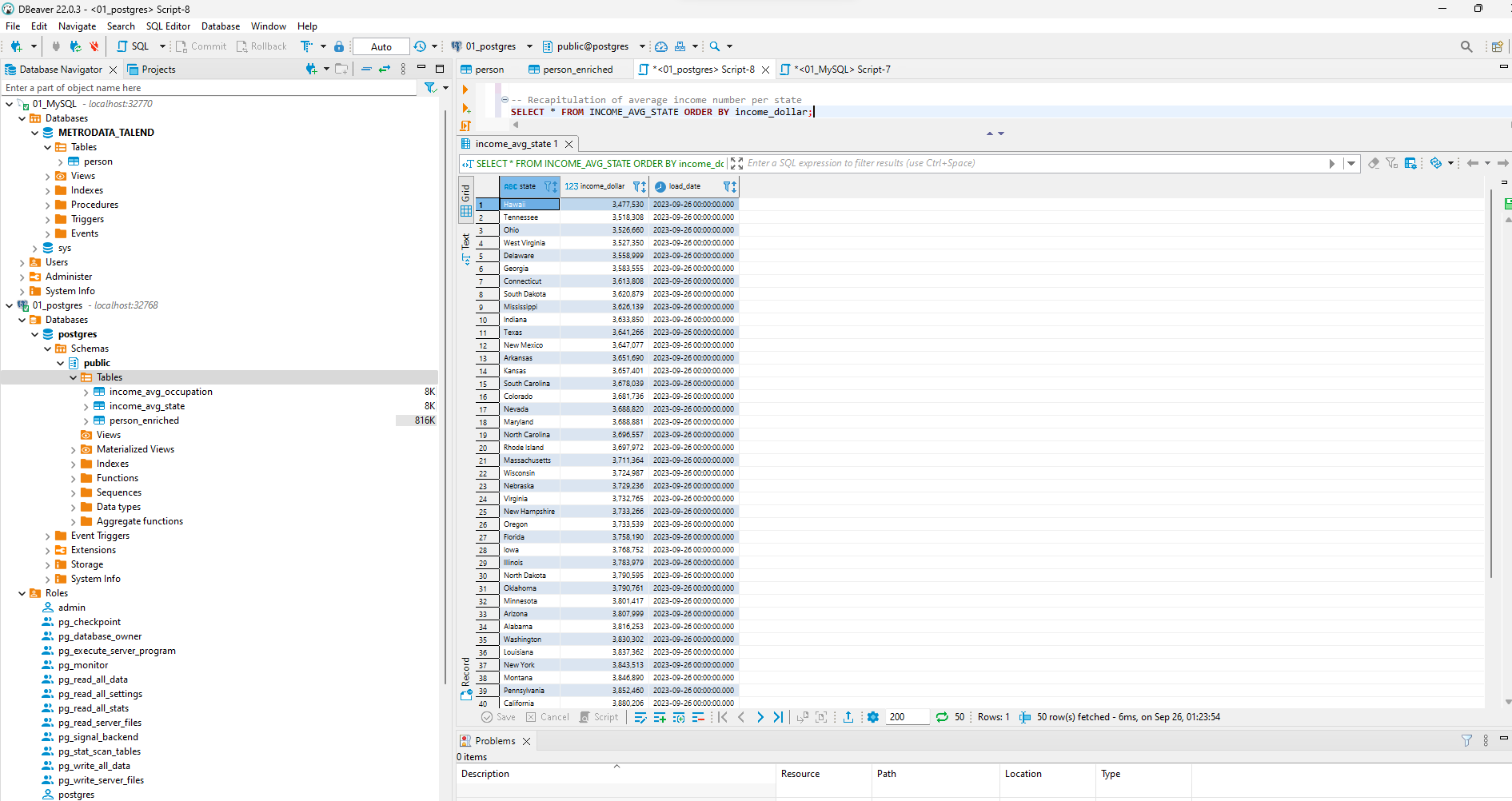
* **person\_enriched Table**

****

* **income\_avg\_occupation**

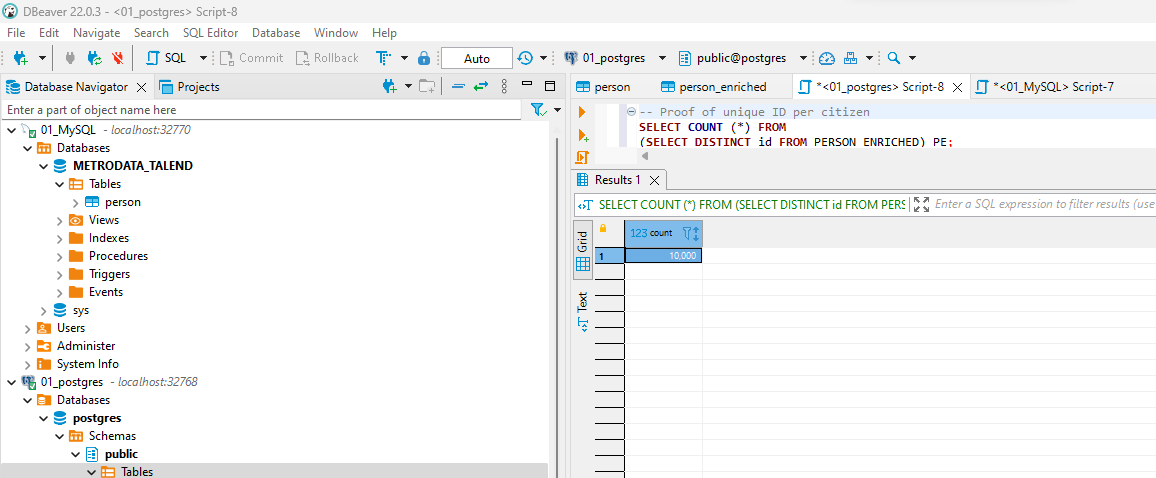
****

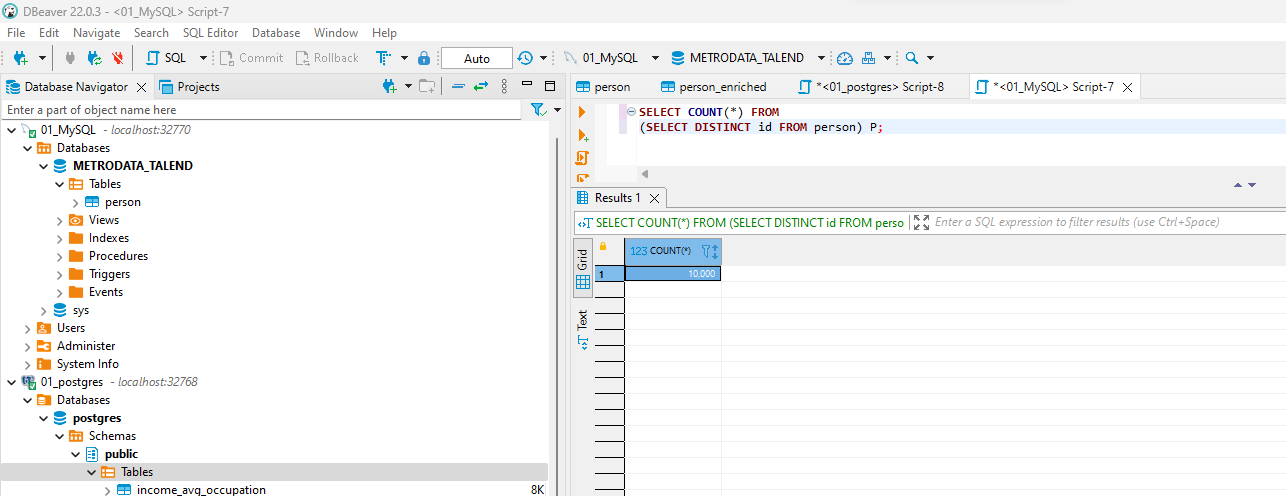
* **income\_avg\_state**

****

1. **Verification:**

* **Kolom id unik:**





* **Output total sama:**

