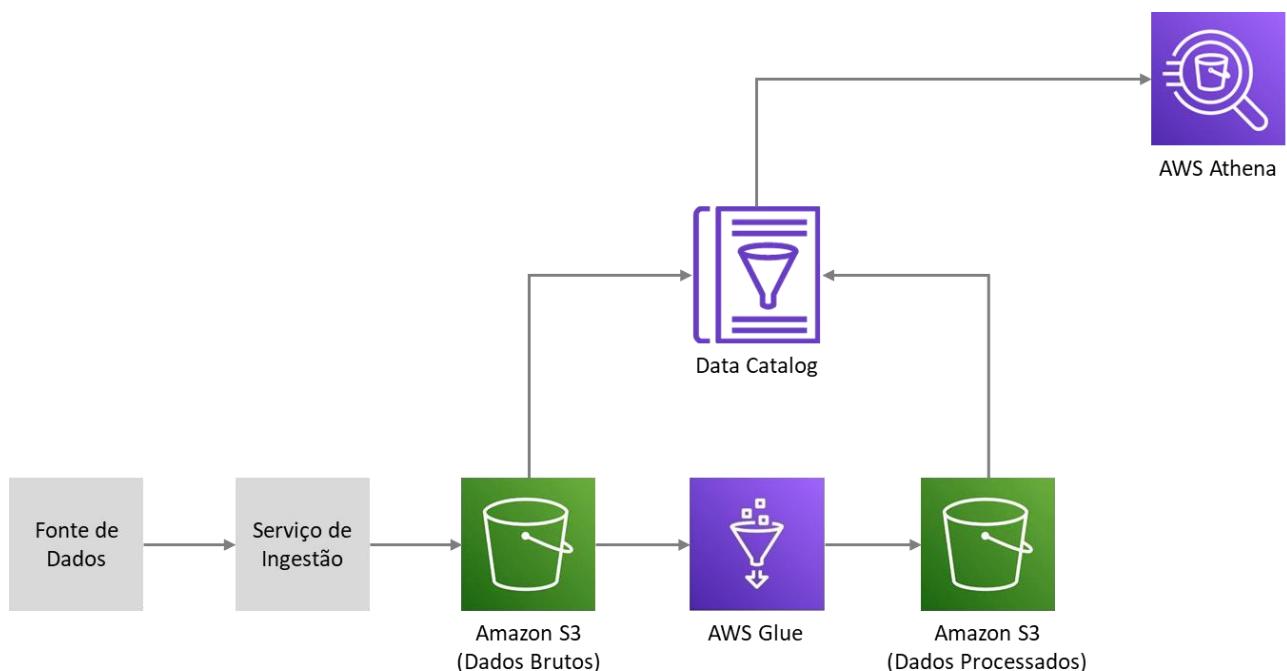


# MVP - Engenharia de Dados

Trabalho de conclusão da Sprint III – Engenharia de Dados, do curso de especialização em Ciência de Dados e Analytics, realizado na PUC Rio, entre agosto e setembro de 2023, por Vinicius de Ávila Ribeiro.

## 1. Objetivo

O objetivo deste trabalho é apresentar um *pipeline* contemplando busca, coleta, modelagem, carga e análise de dados, utilizando as tecnologias em nuvem oferecidas pela Amazon AWS, que será realizado conforme esquema apresentado no diagrama a seguir, destacando-se a catalogação das fontes de dados, com auxílio do AWS Glue, e a consulta aos dados processados com o AWS Athena, para análise e geração de *insights*.



A última etapa desse *pipeline* — análise de dados, se propõe a executar consultas em SQL com o intuito de responder às seguintes perguntas:

- Quais são as regiões do Brasil, Unidades da Federação e municípios que mais transacionaram recursos por Pix?
- Quais são as regiões do Brasil, Unidades da Federação e municípios que apresentam maior valor de recursos transacionados por Pix em relação ao PIB – Produto Interno Bruto das respectivas regiões, Unidades da Federação e municípios?
- Existe relação positiva entre os maiores pagadores e os maiores recebedores de recursos transacionados por Pix?

## **2. Pipeline de Dados**

### **2.1. Busca pelos Dados**

Para responder às perguntas elencadas no item 1, foram buscadas duas bases de dados abertos, disponíveis para *download* no portal do BACEN - Banco Central do Brasil (<https://dadosabertos.bcb.gov.br/dataset/pix>) e no portal Cidades Sustentáveis (<https://www.cidadessustentaveis.org.br/paginas/idsc-br>), descritas abaixo:

- *transacoes\_pix.csv*: dados das transações liquidadas por Pix no SPI – Sistema de Pagamentos Instantâneos.
- *idsc\_ibge\_2023.csv*: coleção de indicadores temáticos que compõem o IDSC – Índice de Desenvolvimento Sustentável das Cidades.

### **2.2. Coleta**

Para coleta dos dados, filtros e seleções foram utilizados com o intuito de reduzir o tamanho das bases e proporcionar melhor desempenho e consumo dos recursos oferecidos gratuitamente pela Amazon AWS, sem causar, no entanto, prejuízo à construção/execução do *pipeline* de dados a que este trabalho se propõe.

Em relação à fonte de dados *transações\_pix.csv*, foi aplicado filtro de data, capturando apenas as transações de agosto e parte das transações de setembro de 2023. Quanto à fonte de dados *idsc\_ibge\_2023.csv*, foram selecionados apenas alguns dos indicadores para composição da base de dados utilizadas neste trabalho.

Os dados foram coletados por meio de *download* de arquivos CSV, nos portais citados no item 2.1, e carregados para o *storage* da Amazon AWS por meio de simples *upload*, realizado manualmente.

Para isso, foi criado um contêiner para os dados (Bucket S3 – Amazon Simple Storage Service) composto pela pasta *mvp-dados* e por suas subpastas *brutos* e *processados*, que receberão, respectivamente, os dados originais coletados nos portais citados e os dados processados e catalogados com o auxílio de um *crawler* criado na Amazon AWS para indexação das fontes de dados e criação de um *database* com tabelas disponíveis para consulta, conforme evidências a seguir.

O *crawler* da Amazon AWS acessa os dados armazenados, classifica-os para determinar o formato, o esquema e as propriedades associadas aos dados brutos, agrupa os dados em tabelas e grava metadados no AWS Glue Data Catalog.

## Evidência 1 – criação do Bucket S3, com pasta e subpastas.

The screenshot shows the AWS S3 console interface. The URL is <https://s3.console.aws.amazon.com/s3/buckets/mvp-pipeline-dados?region=us-east-2&prefix=mvp-dados/&showversions=false>. The page displays the contents of the 'mvp-dados/' folder. There are two items listed:

| Nome         | Tipo  | Última modificação | Tamanho | Classe de armazenamento |
|--------------|-------|--------------------|---------|-------------------------|
| brutos/      | Pasta | -                  | -       | -                       |
| processados/ | Pasta | -                  | -       | -                       |

## Evidência 2 – upload das fontes de dados para a pasta *mvp-dados/brutos/*.

The screenshot shows the AWS S3 console interface. The URL is <https://s3.console.aws.amazon.com/s3/buckets/mvp-pipeline-dados?region=us-east-2&prefix=mvp-dados/brutos/&showversions=false>. The page displays the contents of the 'brutos/' folder. There are two objects listed:

| Nome               | Tipo | Última modificação          | Tamanho  | Classe de armazenamento |
|--------------------|------|-----------------------------|----------|-------------------------|
| idsc_ibge_2023.csv | csv  | 14 Sep 2023 09:01:27 AM -03 | 278.2 KB | Padrão                  |
| transacoes_pix.csv | csv  | 16 Sep 2023 05:03:36 PM -03 | 719.0 KB | Padrão                  |

### Evidência 3 – configuração do crawler - criação do crawler\_dados\_brutos.

The screenshot shows the 'Set crawler properties' step of the AWS Glue crawler creation wizard. The crawler is named 'crawler\_dados\_brutos'. The 'Description' field is empty. The 'Tags' section is optional. Buttons for 'Cancel' and 'Next' are at the bottom.

### Evidência 4 – configuração do crawler - escolha de uma fonte de dados do Amazon S3.

The screenshot shows the 'Choose data sources and classifiers' step of the AWS Glue crawler creation wizard. It shows a single S3 data source configuration. The 'Type' is S3 and the 'Data source' is s3://mvp-pipeline-dados/mvp-dados/brutos/. Buttons for 'Cancel', 'Previous', and 'Next' are at the bottom.

## Evidência 5 – configuração do crawler - criação da identidade de segurança *mvp-glue-role*.

The screenshot shows the AWS Glue console interface for creating a new crawler. The current step is 'Configure security settings'. In the 'IAM role' section, the role 'mvp-glue-role' is selected from a dropdown. Below it are buttons for 'Create new IAM role' and 'Update chosen IAM role'. There are also sections for 'Lake Formation configuration - optional' and 'Security configuration - optional'. At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons.

## Evidência 6 – configuração do crawler - criação do database *mvp\_database*.

The screenshot shows the AWS Glue console interface for creating a new crawler. The current step is 'Set output and scheduling'. In the 'Output configuration' section, the target database is set to 'mvp\_database'. There are buttons for 'Clear selection' and 'Add database'. Below it is a 'Table name prefix - optional' field. In the 'Crawler schedule' section, there is a note about defining a cron-like schedule. The 'Frequency' dropdown is set to 'On demand'. At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons.

## Evidência 7 – configuração do crawler - revisão das informações e criação do crawler\_dados\_brutos.

The screenshot shows the 'Review and create' step of the AWS Glue crawler creation wizard. The crawler is named 'crawler\_dados\_brutos'. It has one data source of type 'S3' pointing to 's3://mvp-pipeline-dados/mvp-dados/brutos/'. The crawler uses the 'mvp-glue-role' IAM role. The security configuration is set to 'Lake Formation configuration'. The output database is 'mvp\_database'. The crawler is scheduled to run 'On demand'. At the bottom right, the 'Create crawler' button is highlighted in orange.

## Evidência 8 – configuração do crawler - crawler criado com sucesso.

The screenshot shows the successful creation of the crawler 'crawler\_dados\_brutos'. A green banner at the top states 'One crawler successfully created'. The crawler properties are listed: Name: 'crawler\_dados\_brutos', IAM role: 'mvp-glue-role', Database: 'mvp\_database', State: 'READY'. The 'Crawler runs' section shows '0' runs, with a note 'You don't have any crawler runs.' and a 'Run crawler' button. The status bar at the bottom indicates the crawler was last updated on September 14, 2023, at 12:34:25.

## Evidência 9 – execução do crawler.

The screenshot shows the AWS Glue console interface. At the top, there are several tabs: 'mvp-pipeline-dados - S3 bucket', 'Crawlers - AWS Glue Console', 'Query editor | Athena | us-east-1', and 'Visual - Editor - AWS Glue Studio'. The main content area displays the configuration for a crawler named 'crawler\_dados\_brutos'. A green banner at the top indicates that the crawler is 'successfully starting'. The crawler properties section includes fields for Name ('crawler\_dados\_brutos'), IAM role ('mvp-glue-role'), Database ('mvp\_database'), and State ('READY'). Advanced settings include options for schema inheritance, partition creation, and table level updates. Below this, the 'Crawler runs' tab is selected, showing one run record from September 14, 2023, at 12:34:46, which is currently running. The status bar at the bottom right shows the date as 'September 14, 2023 at 12:34:25'.

## Evidência 10 – crawler executado com sucesso.

This screenshot shows the AWS Glue console with the same crawler configuration as evidence 9. A modal window titled 'Crawler run details' is open, showing the results of a completed run. The run ID is '46df265-f265-42cf-b4ef-f75e9489a5c2'. The status is 'Completed'. The run occurred on September 14, 2023, between 12:42:01 and 12:42:48, with a duration of 47062 seconds (approximately 7.84 minutes). The log button is visible. The main crawler configuration page is visible in the background, showing the crawler is now 'READY'.

## Evidência 11 – mvp\_database criado com as tabelas processadas.

The screenshot shows the AWS Glue Data Catalog interface. The main view displays the 'mvp\_database' database properties, including its name, location, and creation date. Below this, a table listing shows two tables: 'idsc\_ibge\_2023\_csv' and 'transacoes\_pix\_csv'. Both tables belong to the 'mvp\_database' and are located at 's3://mvp-pipeline-dados/mvp-'. The 'idsc\_ibge\_2023\_csv' table has a 'Classification' of 'CSV' and was last updated on September 14, 2023. The 'transacoes\_pix\_csv' table also has a 'Classification' of 'CSV' and was last updated on September 14, 2023 at 12:33:29. The interface includes standard AWS navigation and search tools.

## Evidência 12 – mvp\_database - tabela idsc\_ibge\_2023\_csv criada com a identificação automática de parâmetros.

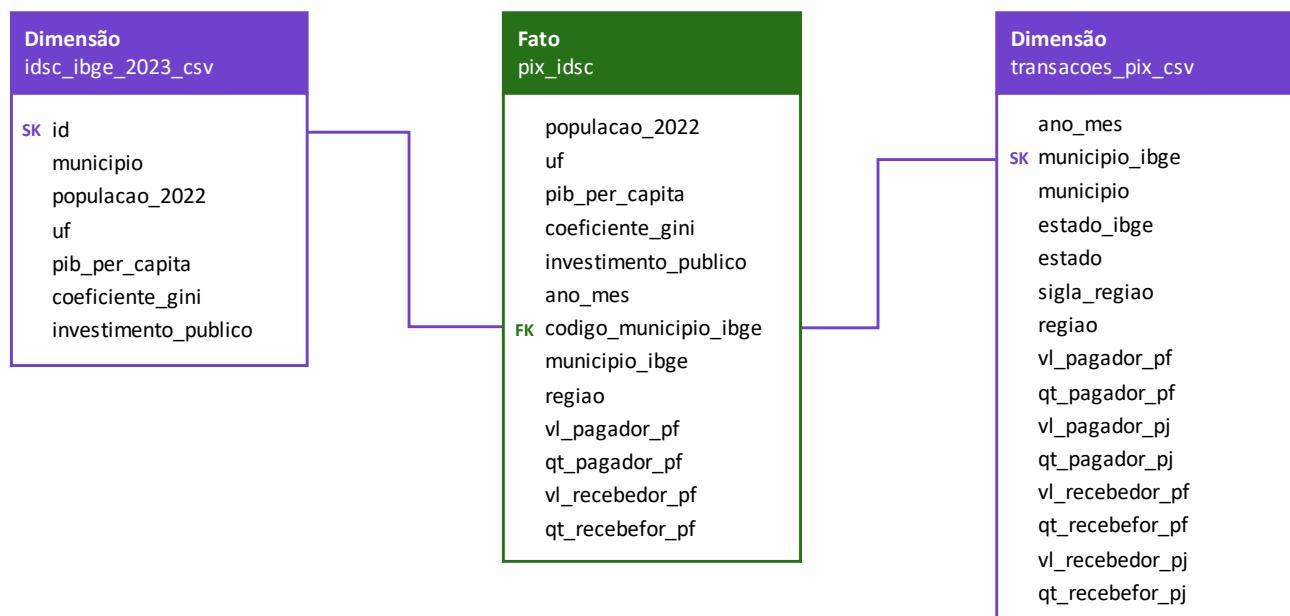
The screenshot shows the AWS Glue Data Catalog interface for the 'idsc\_ibge\_2023\_csv' table within the 'mvp\_database'. The table details page shows the table's name, location ('s3://mvp-pipeline-dados/mvp-dados/brutos/idsc\_ibge\_2023.csv'), connection, and classification ('CSV'). It also shows the input and output formats, and the Serde serialization library used ('org.apache.hadoop.mapred.TextInputFormat' and 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat' respectively). The schema section lists seven columns: 'id' (bigint), 'municipio' (string), 'populacao\_2022' (bigint), 'uf' (string), 'pib\_per\_capita' (double), 'coeficiente\_gini' (double), and 'investimento\_publico' (double). The table was last updated on September 14, 2023 at 12:42:47. The interface includes standard AWS navigation and search tools.

Evidência 13 – *mvp\_database* - tabela *transacoes\_pix\_csv* criada com a identificação automática de parâmetros.

| # | Column name    | Data type | Partition key | Comment |
|---|----------------|-----------|---------------|---------|
| 1 | anomes         | bigint    | -             | -       |
| 2 | municipio_ibge | bigint    | -             | -       |
| 3 | municipio      | string    | -             | -       |
| 4 | estado_ibge    | bigint    | -             | -       |
| 5 | estado         | string    | -             | -       |
| 6 | sigla_regiao   | string    | -             | -       |
| 7 | regiao         | string    | -             | -       |

## 2.3. Modelagem

Para este trabalho, os dados foram modelados tendo como base um esquema Estrela com duas dimensões e uma tabela fato — modelo idealizado para responder às perguntas elencadas no item 1, conforme apresentado abaixo:



Por meio do AWS Glue Data Catalog, pode-se observar a seguir, a descrição detalhada dos dados, com suas principais características e linhagens.

## Evidência 14 – Data Catalog – detalhamento das tabelas do modelo de dados.

The screenshot shows the AWS Glue Data Catalog interface. On the left, a sidebar navigation includes 'AWS Glue', 'Data Catalog', 'Databases', 'Tables', 'Stream schema registries', 'Schemas', 'Connections', 'Crawlers', 'Classifiers', 'Catalog settings', 'Data Integration and ETL', and 'ETL jobs'. The main content area displays the 'mvp\_database' properties (Name: mvp\_database, Description: Dados coletados do Banco Central do Brasil (Transações PIX) e do IBGE (Índice de Desenvolvimento Sustentável das Cidades), Location: s3://mvp-pipeline-dados/mvp-, Created on (UTC): September 14, 2023 at 12:53:29) and a table list titled 'Tables (3)'. The table list includes 'idsc\_ibge\_2023\_csv', 'pix\_idsc', and 'transacoes\_pix\_csv', all located in the 'mvp\_database' with CSV and Parquet formats respectively.

## Evidência 15 – Detalhamento da tabela `idsc_ibge_2023_csv` – Características.

The screenshot shows the detailed properties of the 'idsc\_ibge\_2023\_csv' table within the 'mvp\_database'. It includes sections for 'Table overview' (Data quality: New), 'Table details' (Advanced properties), 'Serde parameters (1)' (Key: field.delim, Value: ','), and 'Table properties (13)'. Key properties include skip.header.line.count (Value: 1), sizeKey (Value: 284903), UPDATED\_BY\_CRAWLER (Value: crawler\_dados\_brutos), recordCount (Value: 1.0), averageRecordSize (Value: 4595), 62, CrawlerSchemaDeserializerVersion (Value: 1.0), compressionType (Value: none), classification (Value: csv), columnsOrdered (Value: true), areColumnsQuoted (Value: false), delimiter (Value: ','), and typeOfData (Value: file).

## Evidência 16 – Detalhamento da tabela *idsc\_ibge\_2023\_csv* – Linhagem, Schema e Dicionário de Dados.

Screenshot of the AWS Glue Console showing the details of the table *idsc\_ibge\_2023\_csv*.

**Table Overview:**

- Name:** idsc\_ibge\_2023\_csv
- Description:** Dados brutos baixados do portal Cidades Sustentáveis (<https://www.cidades sustentaveis.org.br/paginas/idsc-br>) e processados/catalogados pelo crawler\_idsc\_dados\_brutos.
- Database:** mvp\_database
- Classification:** CSV
- Last updated (UTC):** September 30, 2023 at 17:39:01
- Version:** Version 3 (Current version)
- Actions:** Actions

**Table Details:**

|  |  |  |   |
|--|--|--|---|
| <b>Name:</b> idsc_ibge_2023_csv  | <b>Description:</b> Dados brutos baixados do portal Cidades Sustentáveis ( <a href="https://www.cidades sustentaveis.org.br/paginas/idsc-br">https://www.cidades sustentaveis.org.br/paginas/idsc-br</a> ) e processados/catalogados pelo crawler_idsc_dados_brutos. | <b>Database:</b> mvp_database  | <b>Classification:</b> CSV                          |
| <b>Location:</b> s3://mvp-pipeline-dados/mvp-dados/brutos/idsc_ibge_2023.csv | <b>Connection:</b> -   | <b>Deprecated:</b> -   | <b>Last updated:</b> September 30, 2023 at 17:39:01 |
| <b>Input format:</b> org.apache.hadoop.mapred.TextInputFormat                | <b>Output format:</b> org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat   | <b>Serde serialization lib:</b> org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe |   |

**Schema:**

| # | Column name    | Data type | Partition key | Comment                     |
|---|----------------|-----------|---------------|-----------------------------|
| 1 | id             | bigint    | -             | Código IBGE do município    |
| 2 | municipio      | string    | -             | Nome do município           |
| 3 | populacao_2022 | bigint    | -             | População do município      |
| 4 | uf             | string    | -             | Unidade da Federação        |
| 5 | pib_per_capita | double    | -             | PIB per capita do município |

Screenshot of the AWS Glue Console showing the details of the table *idsc\_ibge\_2023\_csv*.

**Table Overview:**

- Name:** idsc\_ibge\_2023\_csv
- Description:** Dados brutos baixados do portal Cidades Sustentáveis (<https://www.cidades sustentaveis.org.br/paginas/idsc-br>) e processados/catalogados pelo crawler\_idsc\_dados\_brutos.
- Database:** mvp\_database
- Classification:** CSV
- Last updated:** September 30, 2023 at 17:39:01

**Table Details:**

|  |  |  |   |
|--|--|--|---|
| <b>Name:</b> idsc_ibge_2023_csv  | <b>Description:</b> Dados brutos baixados do portal Cidades Sustentáveis ( <a href="https://www.cidades sustentaveis.org.br/paginas/idsc-br">https://www.cidades sustentaveis.org.br/paginas/idsc-br</a> ) e processados/catalogados pelo crawler_idsc_dados_brutos. | <b>Database:</b> mvp_database  | <b>Classification:</b> CSV                          |
| <b>Location:</b> s3://mvp-pipeline-dados/mvp-dados/brutos/idsc_ibge_2023.csv | <b>Connection:</b> -   | <b>Deprecated:</b> -   | <b>Last updated:</b> September 30, 2023 at 17:39:01 |
| <b>Input format:</b> org.apache.hadoop.mapred.TextInputFormat                | <b>Output format:</b> org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat   | <b>Serde serialization lib:</b> org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe |   |

**Schema:**

| # | Column name          | Data type | Partition key | Comment                                    |
|---|----------------------|-----------|---------------|--|
| 1 | id                   | bigint    | -             | Código IBGE do município                   |
| 2 | municipio            | string    | -             | Nome do município                          |
| 3 | populacao_2022       | bigint    | -             | População do município                     |
| 4 | uf                   | string    | -             | Unidade da Federação                       |
| 5 | pib_per_capita       | double    | -             | PIB per capita do município                |
| 6 | coeficiente_gini     | double    | -             | Índice de Gini                             |
| 7 | investimento_publico | double    | -             | Valor de investimento público no município |

## Evidência 17 – Detalhamento da tabela *transacoes\_pix\_csv* – Características.

**Serde parameters (1)**

| Key         | Value |
|-------------|-------|
| field.delim | ,     |

**Table properties (13)**

| Key                              | Value                |
|----------------------------------|----------------------|
| skip.header.line.count           | 1                    |
| sizeKey                          | 736206               |
| UPDATED_BY_CRAWLER               | crawler_dados_brutos |
| CrawlerSchemaDeserializerVersion | 1.0                  |
| recordCount                      | 4908                 |
| averageRecordSize                | 150                  |
| CrawlerSchemaDeserializerVersion | 1.0                  |
| compressionType                  | none                 |
| classification                   | csv                  |
| columnsOrdered                   | true                 |
| areColumnsQuoted                 | false                |
| delimiter                        | ,                    |
| typeOfData                       | file                 |

## Evidência 18 – Detalhamento da tabela *transacoes\_pix\_csv* – Linhagem, Schema e Dicionário de Dados.

**Schema (15)**

View and manage the table schema.

| # | Column name    | Data type | Partition key | Comment                    |
|---|----------------|-----------|---------------|----------------------------|
| 1 | anomes         | bigint    | -             | Ano e mês da transação PIX |
| 2 | municipio_ibge | bigint    | -             | Código IBGE do município   |
| 3 | municipio      | string    | -             | Nome do município          |
| 4 | estado_ibge    | bigint    | -             | Código IBGE do Estado      |
| 5 | estado         | string    | -             | Nome do Estado             |

Screenshot of the AWS Glue Console showing the schema for the 'transacoes\_pix' table.

**Table Details:**

- Location:** s3://mvp-pipeline-dados/mvp-dados/brutos/transacoes\_pix.csv
- Connection:** -
- Deprecated:** -
- Last updated:** September 30, 2023 at 17:41:14

**Input format:** org.apache.hadoop.mapred.TextInputFormat  
**Output format:** org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat  
**Serde serialization lib:** org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe

**Schema (15):**

| #  | Column name    | Data type | Partition key | Comment  |
|----|----------------|-----------|---------------|--|
| 1  | anomes         | bigint    | -             | Ano e mês da transação PIX   |
| 2  | municipio_ibge | bigint    | -             | Código IBGE do município   |
| 3  | municipio      | string    | -             | Nome do município  |
| 4  | estado_ibge    | bigint    | -             | Código IBGE do Estado  |
| 5  | estado         | string    | -             | Nome do Estado   |
| 6  | sigla_regioao  | string    | -             | Sigla da região do Brasil  |
| 7  | regiao         | string    | -             | Nome da região   |
| 8  | vl_pagadorpf   | double    | -             | Valor total transacionado por PIX originado no município por Pessoas Físicas   |
| 9  | qt_pagadorpf   | bigint    | -             | Quantidade total de PIX originado no município por Pessoas Físicas             |
| 10 | vl_pagadorpj   | double    | -             | Valor total transacionado por PIX originado no município por Pessoas Jurídicas |
| 11 | qt_pagadorpj   | bigint    | -             | Quantidade total de PIX originado no município por Pessoas Jurídicas           |
| 12 | vl_recebedorpf | double    | -             | Valor total transacionado por PIX destinado ao município por Pessoas Físicas   |
| 13 | qt_recebedorpf | bigint    | -             | Quantidade total de PIX destinado ao município por Pessoas Físicas             |
| 14 | vl_recebedorpj | double    | -             | Valor total transacionado por PIX destinado ao município por Pessoas Jurídicas |
| 15 | qt_recebedorpj | bigint    | -             | Quantidade total de PIX destinado ao município por Pessoas Jurídicas           |

**Actions:** Edit schema as JSON, Edit schema

### Evidência 19 – Detalhamento da tabela pix\_idsc – Características.

Screenshot of the AWS Glue Console showing the schema for the 'pix\_idsc' table.

**Table Details:**

- Last updated (UTC):** September 21, 2023 at 01:35:52
- Version:** Version 2 (Current version)
- Actions:** Actions

**Serde parameters (0):**

| Key                       | Value |
|---------------------------|-------|
| No parameters             |       |
| No parameters to display. |       |

**Table properties (4):**

| Key                  | Value  |
|----------------------|--|
| CreatedByJobRun      | jr_a64127efce2c5fc370310d0372e1e1e68b75da1d81148d5b8fdf6436209bb3c |
| CreatedByJob         | job_pix_idsc   |
| classification       | parquet  |
| useGlueParquetWriter | true   |

**Schema (13):**

| # | Column name    | Data type | Partition key | Comment                |
|---|----------------|-----------|---------------|------------------------|
| 1 | populacao_2022 | bigint    | -             | População do município |
| 2 | uf             | string    | -             | Unidade da Federação   |

**Actions:** Edit schema as JSON, Edit schema

## Evidência 20 – Detalhamento da tabela *pix\_idsc* – Linhagem, Schema e Dicionário de Dados.

The screenshot shows the AWS Glue Console interface for the pix\_idsc table. At the top, there's a navigation bar with tabs for 'Table overview' and 'Data quality'. Below this, the 'Table details' tab is selected. The table has the following details:

|   |   |  |  |
|---|---|--|--|
| Name: pix_idsc  | Description: Criada com a execução do job_pix_idsc, que possui como origem as tabelas transacoes_pix_csv e idsc_ibge_csv. | Database: mvp_database   | Classification: Parquet                      |
| Location: s3://mvp-pipeline-dados/mvp-dados/processados/                    | Connection: -   | Deprecated: -  | Last updated: September 30, 2023 at 16:21:39 |
| Input format: org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat | Output format: org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat   | Serde serialization lib: org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe |  |

Below the table details, there are tabs for 'Schema', 'Partitions', and 'Indexes'. The 'Schema' tab is selected, showing a table with 13 columns:

| #  | Column name           | Data type | Partition key | Comment  |
|----|-----------------------|-----------|---------------|--|
| 1  | populacao_2022        | bigint    | -             | População do município   |
| 2  | uf                    | string    | -             | Unidade da Federação   |
| 3  | pib_per_capita        | double    | -             | PIB per capita do município  |
| 4  | coeficiente_gini      | double    | -             | Índice de Gini   |
| 5  | investimento_publico  | double    | -             | Investimento público no município  |
| 6  | ano_mes               | bigint    | -             | Ano e mês da transação PIX   |
| 7  | codigo_municipio_ibge | string    | -             | Código IBGE do município   |
| 8  | municipio_ibge        | string    | -             | Nome do município  |
| 9  | regiao                | string    | -             | Nome da região   |
| 10 | vl_pagador_pf         | double    | -             | Valor total transacionado por PIX originado no município por Pessoas Físicas |
| 11 | qt_pagador_pf         | bigint    | -             | Quantidade total de PIX originado no município por Pessoas Físicas           |
| 12 | vl_recebedor_pf       | double    | -             | Valor total transacionado por PIX destinado ao município por Pessoas Físicas |
| 13 | qt_recebedor_pf       | bigint    | -             | Quantidade total de PIX destinado ao município por Pessoas Físicas           |

This screenshot shows the AWS Glue Console interface for the pix\_idsc table, identical to the one above but with a different schema. It displays 13 columns:

| #  | Column name           | Data type | Partition key | Comment  |
|----|-----------------------|-----------|---------------|--|
| 1  | populacao_2022        | bigint    | -             | População do município   |
| 2  | uf                    | string    | -             | Unidade da Federação   |
| 3  | pib_per_capita        | double    | -             | PIB per capita do município  |
| 4  | coeficiente_gini      | double    | -             | Índice de Gini   |
| 5  | investimento_publico  | double    | -             | Investimento público no município  |
| 6  | ano_mes               | bigint    | -             | Ano e mês da transação PIX   |
| 7  | codigo_municipio_ibge | string    | -             | Código IBGE do município   |
| 8  | municipio_ibge        | string    | -             | Nome do município  |
| 9  | regiao                | string    | -             | Nome da região   |
| 10 | vl_pagador_pf         | double    | -             | Valor total transacionado por PIX originado no município por Pessoas Físicas |
| 11 | qt_pagador_pf         | bigint    | -             | Quantidade total de PIX originado no município por Pessoas Físicas           |
| 12 | vl_recebedor_pf       | double    | -             | Valor total transacionado por PIX destinado ao município por Pessoas Físicas |
| 13 | qt_recebedor_pf       | bigint    | -             | Quantidade total de PIX destinado ao município por Pessoas Físicas           |

## 2.4. Carga

A carga dos dados para o *mvp\_database* foi realizada com auxílio do AWS Glue, utilizado para realização do ETL (Extração, Transformação e Carga [*Load*]). Para configuração de um *job* para execução do ETL, foi utilizado o AWS Glue Studio, conforme evidências apresentadas a seguir.

## Evidência 21 – configuração do job\_pix\_idsc.

The screenshot shows the AWS Glue Studio interface. In the top navigation bar, the URL is https://us-east-1.console.aws.amazon.com/gluestudio/home?region=us-east-1#/jobs. The main area displays the configuration for a job named "job\_pix\_idsc".

**Job Configuration:**

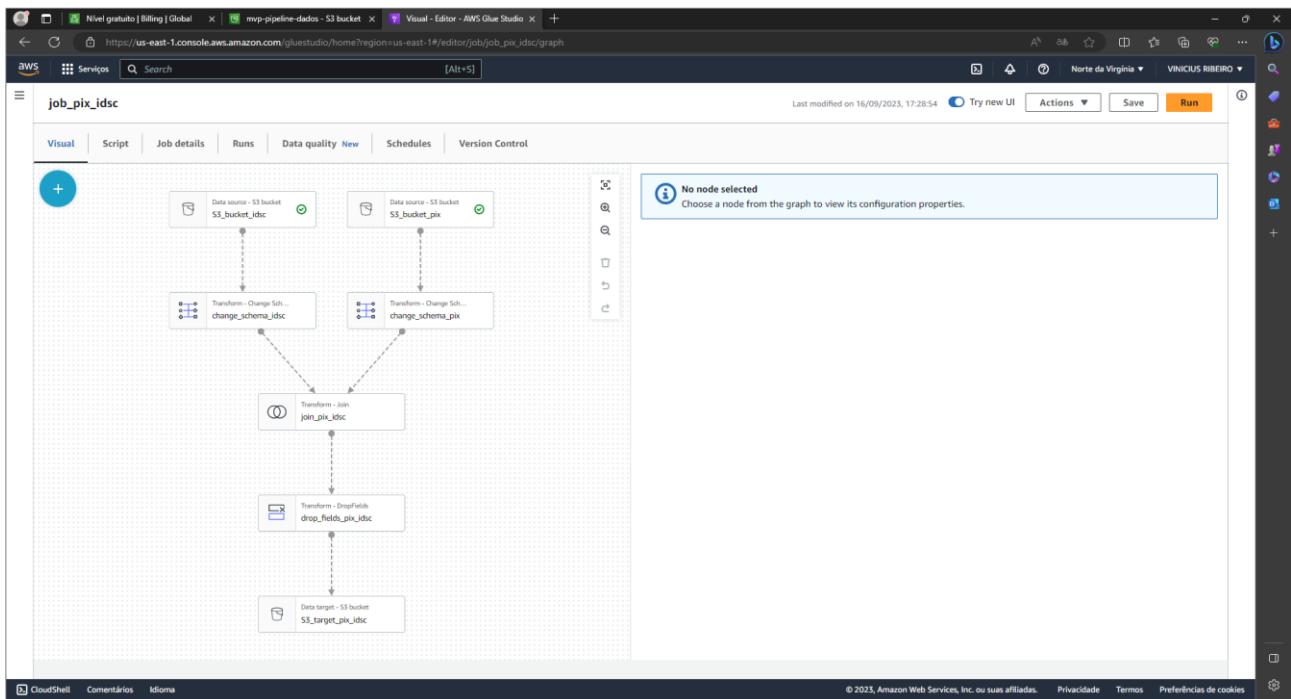
- Source:** Amazon S3 (JSON, CSV, or Parquet Files stored in S3).
- Target:** Amazon S3 (S3 bucket by specifying a bucket path as the data target).

**Your jobs (1) Info**

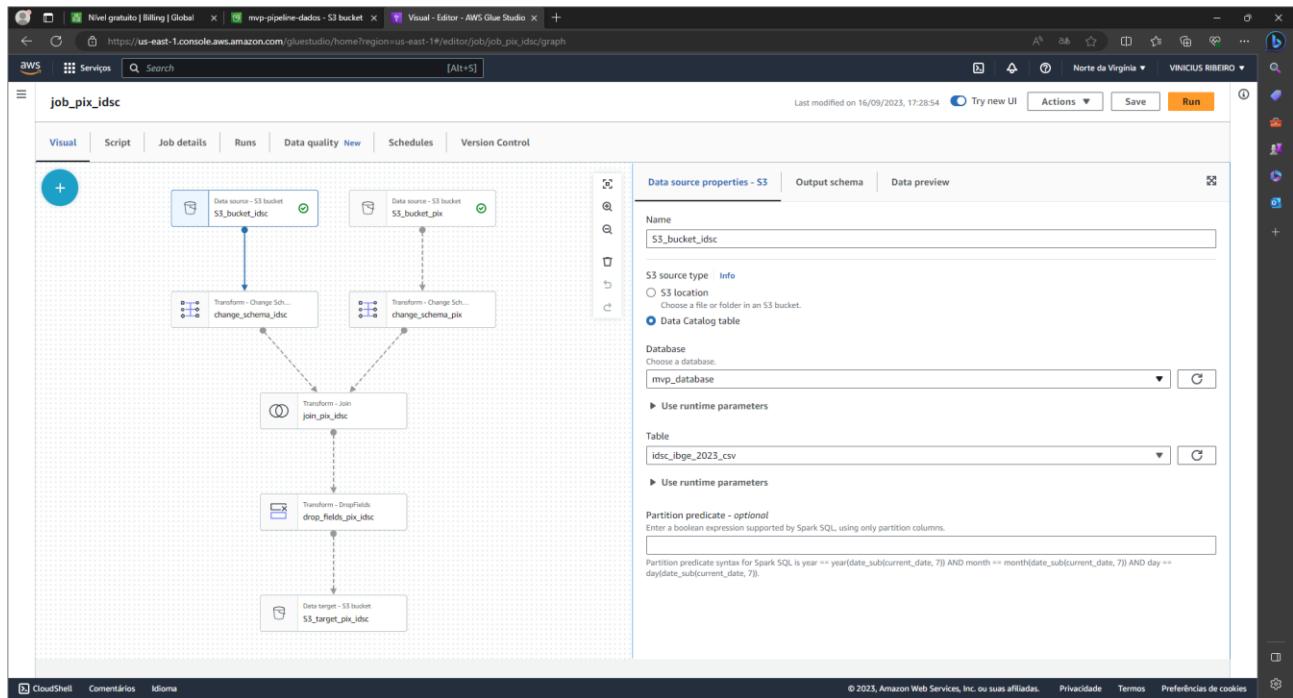
| Job name     | Type     | Last modified        | AWS Glue version |
|--------------|----------|----------------------|------------------|
| job_pix_idsc | Glue ETL | 16/09/2023, 17:28:54 | 4.0              |

**Bottom Navigation:** CloudShell, Comentários, Idioma. © 2023, Amazon Web Services, Inc. ou suas afiliadas. Privacidade, Termos, Preferências de cookies.

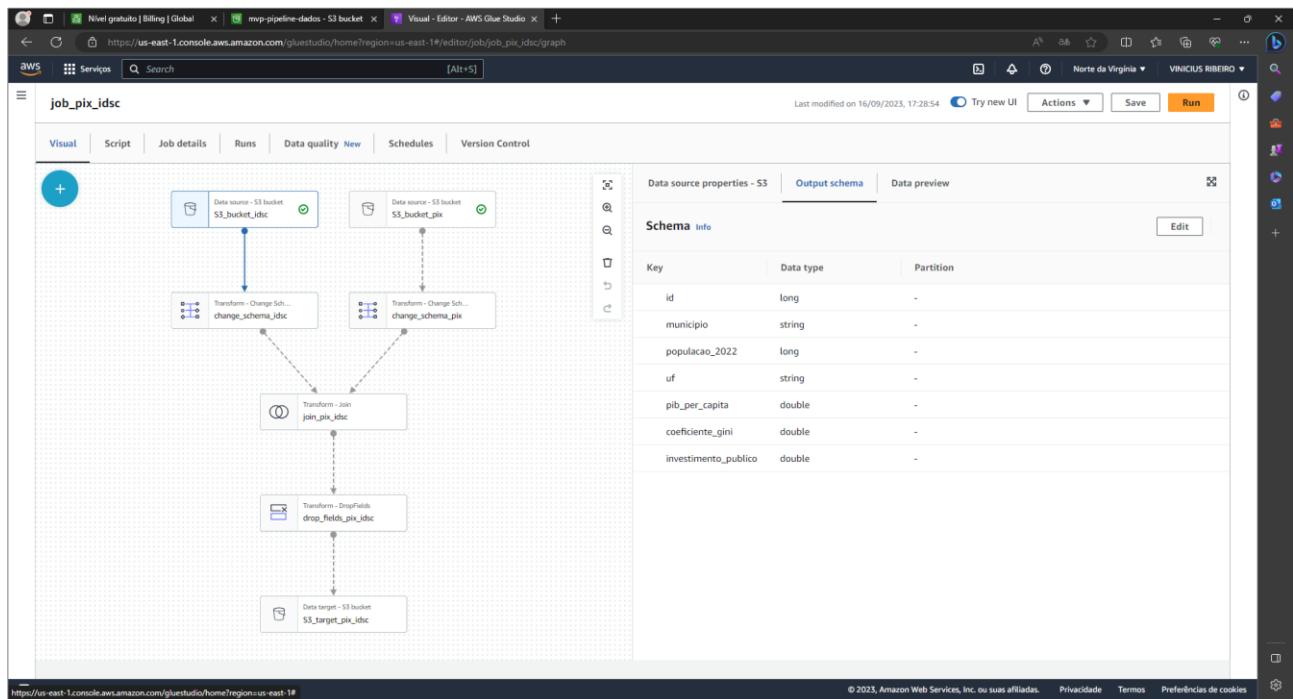
## Evidência 22 – visual do job\_pix\_idsc.



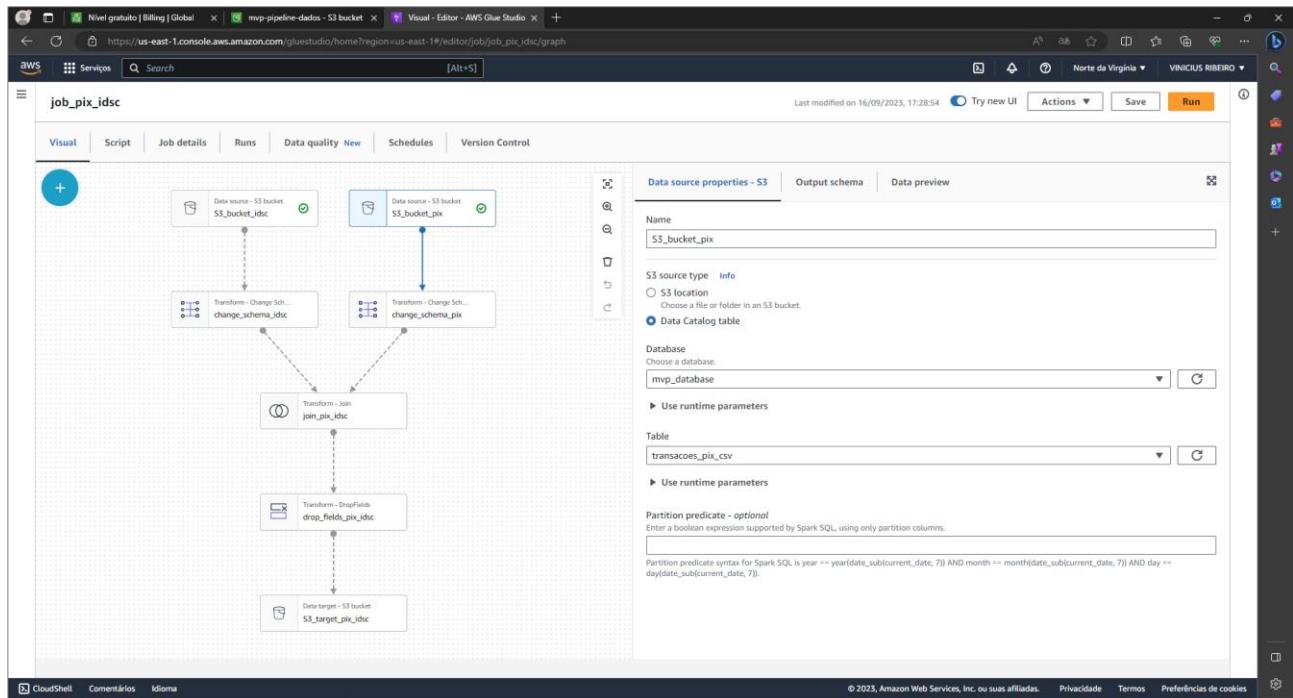
Evidência 23 – configuração da primeira fonte de dados: *idsc\_ibge\_2023\_csv*.



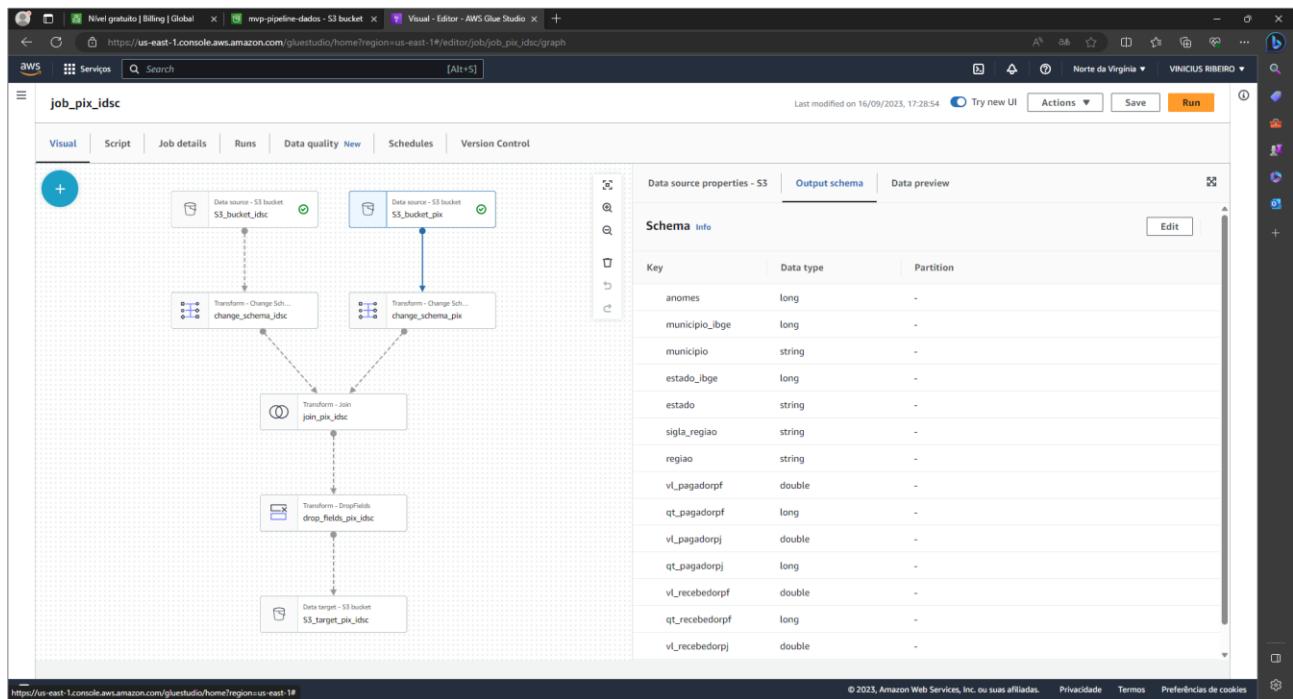
Evidência 24 – configuração da primeira fonte de dados: *idsc\_ibge\_2023\_csv – Output Schema*.



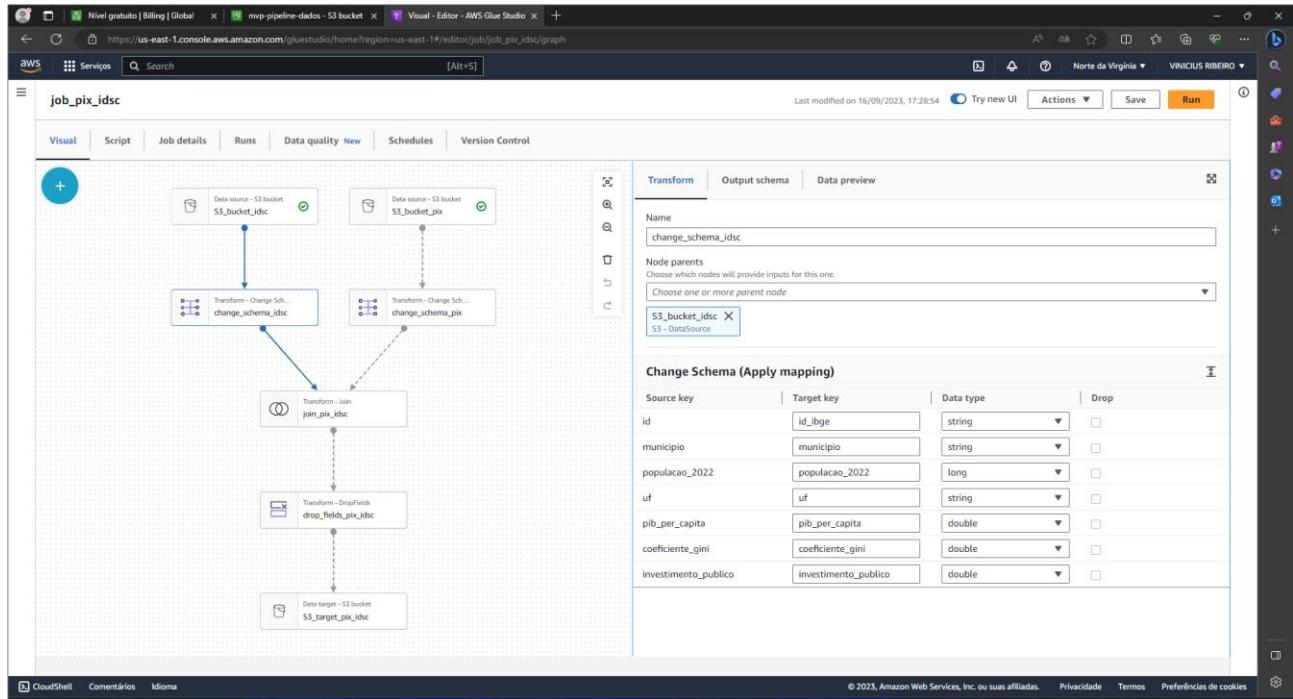
Evidência 25 – configuração da segunda fonte de dados: *transacoes\_pix\_csv*.



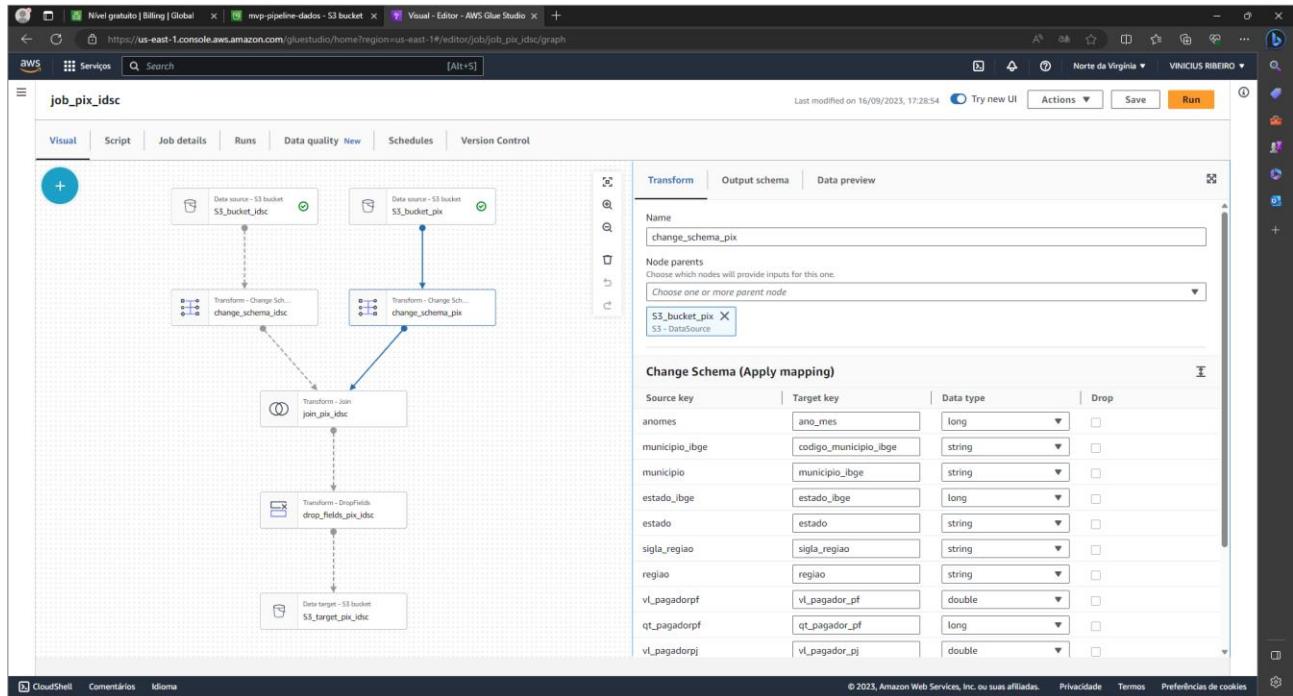
Evidência 26 – configuração da segunda fonte de dados: *transacoes\_pix\_csv* – Output Schema.



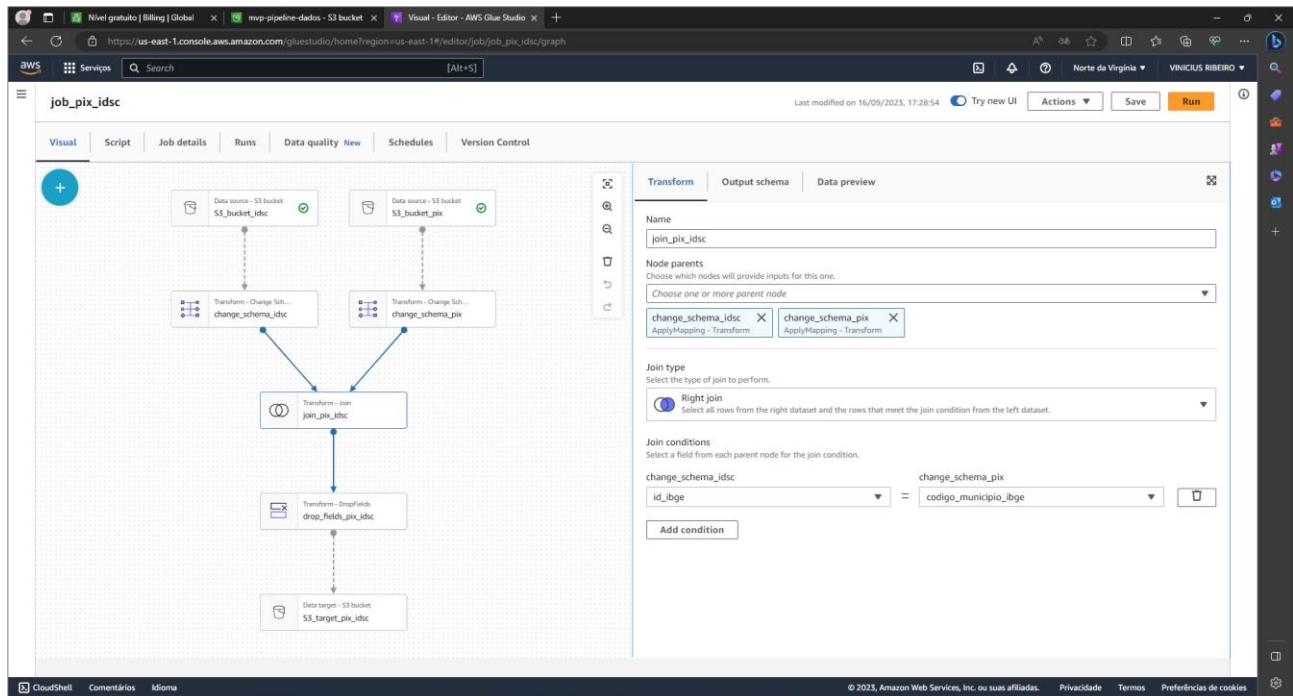
Evidência 27 – transformação da primeira fonte de dados: *idsc\_ibge\_2023\_csv* – ajuste de nome/tipo de variável.



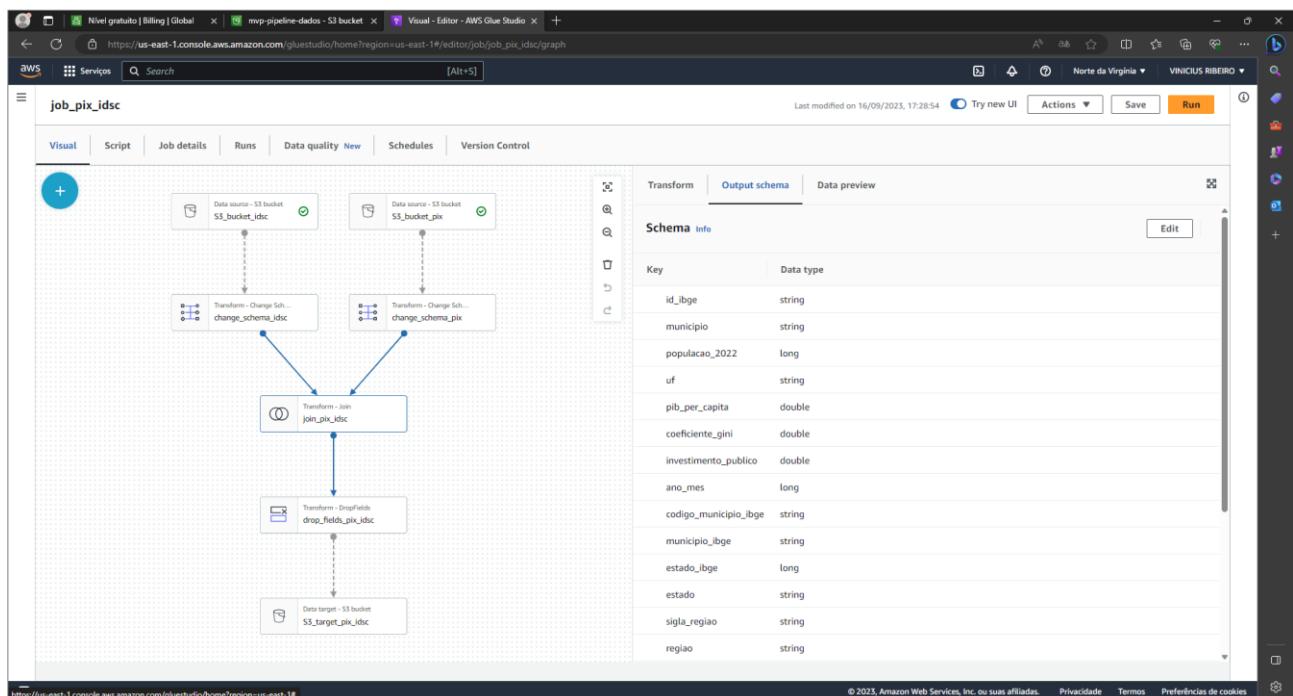
Evidência 28 – transformação da segunda fonte de dados: *transacoes\_pix\_csv* – ajuste de nome/tipo de variável.



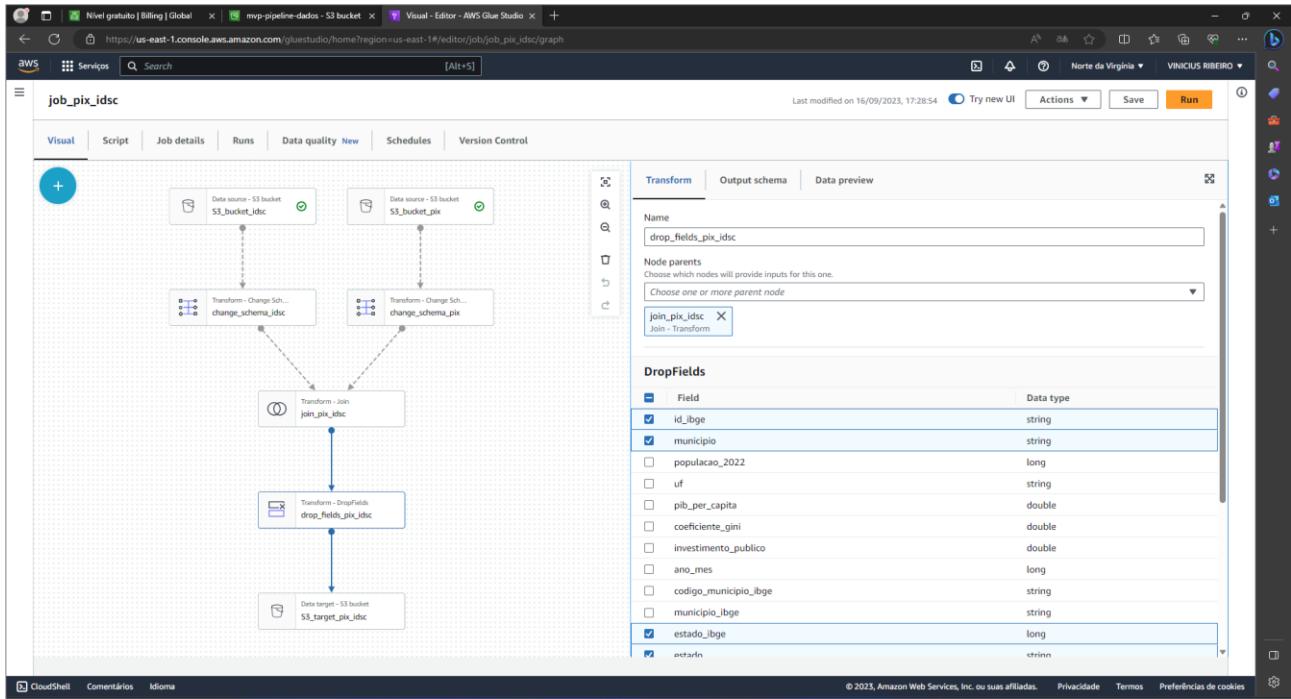
Evidência 29 – configuração de *Join* entre as duas fontes de dados: *idsc\_ibge\_2023\_csv* e *transacoes\_pix\_csv*.



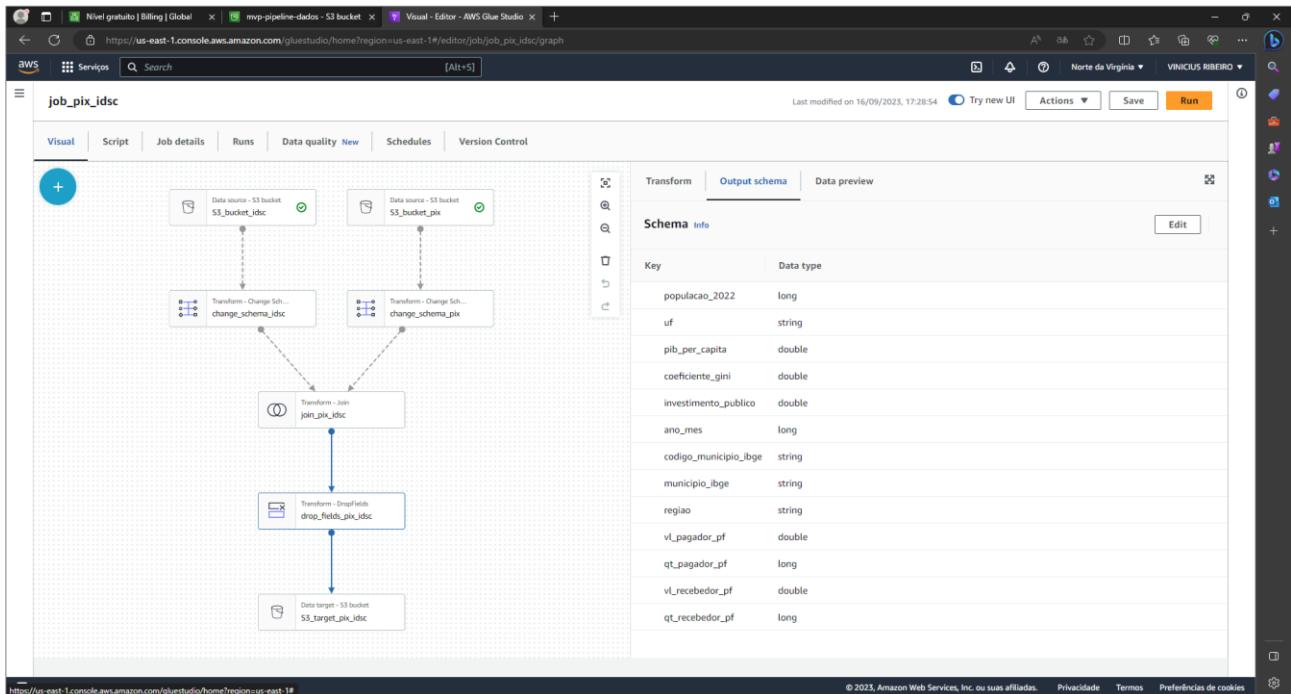
Evidência 30 – configuração de *Join* entre as duas fontes de dados – *Output Schema*.



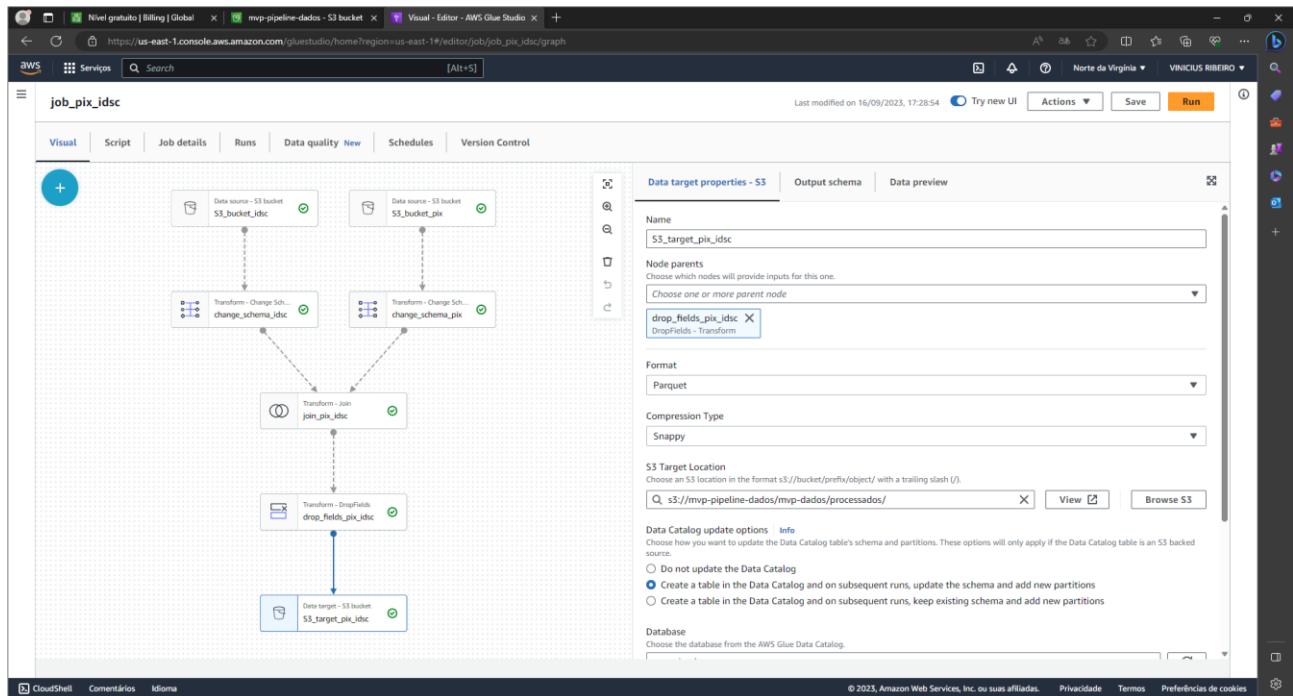
### Evidência 31 – realização de *drop* de alguns campos.



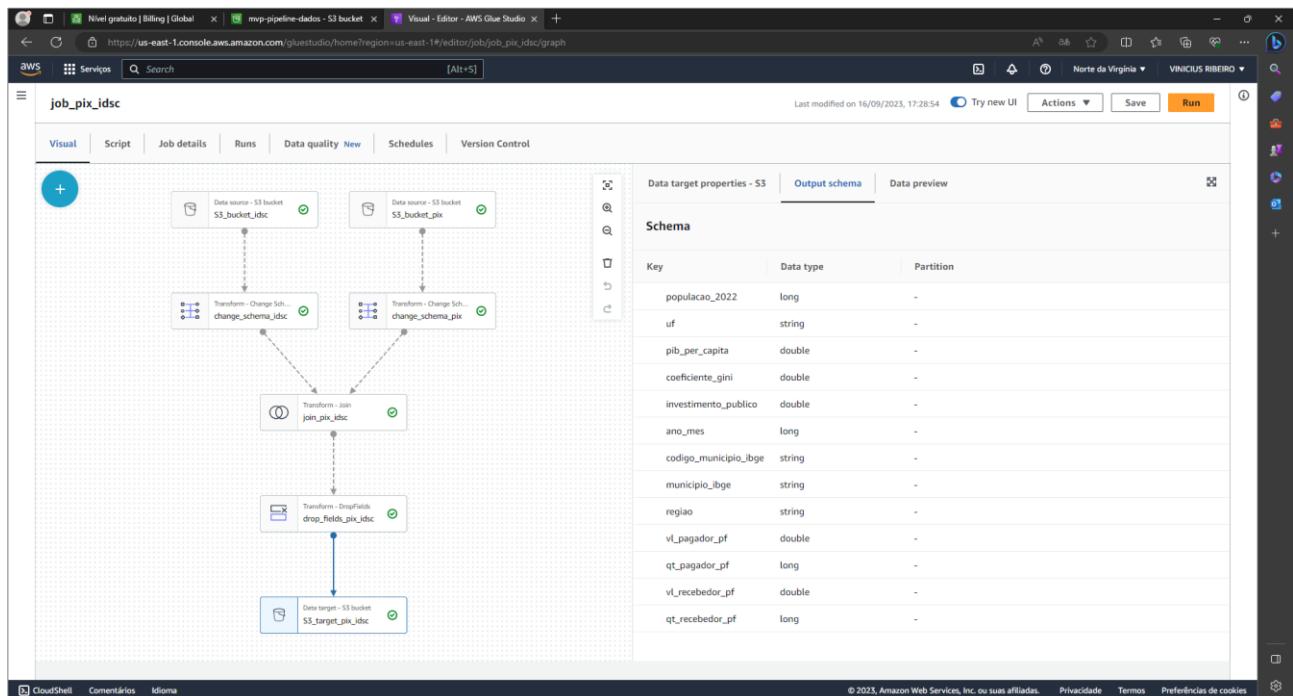
### Evidência 32 – realização de *drop* de alguns campos – *Output Schema*.



Evidência 33 – configuração de um Bucket S3 em formato parquet para recepção dos dados transformados.



Evidência 34 – configuração de um Bucket S3 em formato parquet – Output Schema.



### Evidência 35 – execução do job\_pix\_idsc.

The screenshot shows the AWS Glue Studio interface with the 'Runs' tab selected. The table displays 10 job runs, all of which have succeeded. The columns include Run status, Retries, Start time, End time, Duration, Capacity (DPU), Worker type, and Glue version. The most recent run is highlighted with a blue border and timestamped at 09/16/2023 17:28:58. Below the table, detailed information for this specific run is provided, including the job name (job\_pix\_idsc), ID (jr\_a64127efce2c5fc370310d0372e1e1e68b75da1d81148d5b8fdaf6436209bb3c), run status (Succeeded), and glue version (4.0). Other details shown include retry attempt number (0), start time (16 de setembro de 2023 17:28:58), end time (16 de setembro de 2023 17:30:06), and worker type (G.1X).

### Evidência 36 – execução do job\_pix\_idsc – executado com sucesso.

The screenshot shows the AWS Glue Studio interface with the 'Monitoring' tab selected. It displays the 'Job Run - jr\_a64127efce2c5fc370310d0372e1e1e68b75da1d81148d5b8fdaf6436209bb3c'. The 'Run details' section provides comprehensive information about the run, including the job name (job\_pix\_idsc), ID (jr\_a64127efce2c5fc370310d0372e1e1e68b75da1d81148d5b8fdaf6436209bb3c), run status (Succeeded), and glue version (4.0). It also includes details such as retry attempt number (0), start time (16 de setembro de 2023 17:28:58), end time (16 de setembro de 2023 17:30:06), and worker type (G.1X). The 'CloudWatch continuous logs' section shows log entries from the driver and executor log streams, including tasks finished, map output tracker endpoint requests, and multipart upload output stream activity. The logs are timestamped from 23/09/16 20:29:55 to 23/09/16 20:29:51.

Evidência 37 – arquivos gerados e gravados na pasta *mvp-dados/processados*.

The screenshot shows the AWS S3 console interface. The URL in the address bar is <https://s3.console.aws.amazon.com/s3/buckets/mvp-pipeline-dados?region=us-east-2&prefix=mvp-dados/processados&showversions=false>. The page displays a list of objects in the 'processados' folder of the 'mvp-dados' bucket. There are four parquet files listed:

| Nome  | Tipo    | Última modificação          | Tamanho | Classe de armazenamento |
|---|---------|-----------------------------|---------|-------------------------|
| run-1694896180480-part-block-0-r-00000-snappy.parquet | parquet | 16 Sep 2023 05:29:55 PM -03 | 77.5 KB | Padrão                  |
| run-1694896180480-part-block-0-r-00001-snappy.parquet | parquet | 16 Sep 2023 05:29:55 PM -03 | 78.6 KB | Padrão                  |
| run-1694896180480-part-block-0-r-00002-snappy.parquet | parquet | 16 Sep 2023 05:29:55 PM -03 | 78.4 KB | Padrão                  |
| run-1694896180480-part-block-0-r-00003-snappy.parquet | parquet | 16 Sep 2023 05:29:56 PM -03 | 81.1 KB | Padrão                  |

### 3. Análise

#### 3.1. Ferramenta para Análise dos Dados

Para análise dos dados, foi utilizado o Amazon Athena, um serviço de consultas interativas para análise de dados diretamente no Amazon S3, por meio da linguagem SQL padrão. As consultas à tabela *pix\_idsc* foram realizadas após configuração do Query Editor, conforme parâmetros evidenciados a seguir.

Evidência 38 – parâmetros configurados no Query Editor.

The screenshot shows the AWS Management Console interface for Amazon Athena. The top navigation bar includes tabs for 'S3 bucket' and 'Query editor | Athena | us-east-1'. The main area is titled 'Amazon Athena > Query editor'. A sub-menu bar at the top has 'Editor' selected, along with 'Recent queries', 'Saved queries', and 'Settings'. A 'Workgroup' dropdown is set to 'primary'. On the left, a sidebar titled 'Data' shows 'Data source' as 'AwsDataCatalog' and 'Database' as 'mvp\_database'. Below this, 'Tables and views' are listed under 'Tables (3)'. One table, 'idsc\_ibge\_2023\_csv', is expanded to show columns: 'pix\_idsc' (bigint), 'codigo\_municipio' (string), 'municipio' (string), 'populacao\_2022' (bigint), 'uf' (string), 'pib\_per\_capita' (float), 'coeficiente\_gini' (float), 'investimento\_publico' (float), and 'ano\_mes' (bigint). The main workspace is titled 'Query 1' and contains a SQL editor with the text 'SQL Ln 1, Col 1' and a 'Run' button. Below the editor is a 'Query results' section with a 'Results' tab and a search bar. A note says 'No results' and 'Run a query to view results'. The bottom of the screen shows standard browser controls and a footer with copyright information.

### 3.2. Qualidade dos Dados

Os dados utilizados para este trabalho apresentam ótima qualidade, pois foram previamente tratados pelas organizações que os disponibilizaram, BACEN e IBGE.

No entanto, o *join* realizado entre as duas fontes de dados acarretou um quantitativo irrelevante de perda de dados, por falta de correspondência entre os municípios constantes em cada uma das fontes de dados. Esses registros foram desconsiderados quando da realização das consultas via SQL, como será evidenciado no item 3.3, a seguir, e não acarretaram prejuízo à análise proposta.

Além disso, com auxílio do AWS Glue Data Quality, pode-se observar estatísticas e informações sobre a base *pix\_idsc*, resultante do *join* entre as duas fontes de dados originais, tais como amplitude, desvio padrão, valores mínimos, máximos e valores categóricos das variáveis, conforme evidências a seguir.

## Evidência 39 – resultado da análise do AWS Glue Data Quality.

The screenshot shows the AWS Glue Data Quality console interface. A completed recommendation run is displayed, with detailed statistics and rules defined in the DQDL language.

**Recommendation run details:**

- Status: Completed
- Start time (UTC): September 26, 2023 at 23:51:56
- End time (UTC): September 26, 2023 at 23:53:45
- IAM Role: arn:aws:iam:089402387585:role/mvp-glue-role

**Additional details:**

| Number of workers | Run time (seconds)                 | Database                    | Table    |
|-------------------|------------------------------------|-----------------------------|----------|
| 5                 | 91                                 | mvp_database                | pix_idsc |
| Catalog ID        | Filter condition applied at source | Catalog partition predicate | -        |

**Rules (DQDL):**

```
Rules = [
    RowCount between 2785 and 11140,
    Completeness "populacao_2022" >= 0.99,
    StandardDeviation "populacao_2022" between 196208.18 and 216861.68,
    ColumnValues "populacao_2022" between 832 and 11451246,
    Completeness "uf" >= 0.99,
    ColumnValues "uf" in ["MG", "SP", "RS", "BA", "PR", "SC", "GO", "PI", "PB", "MA", "PE", "CE", "RN", "PA", "MT", "TO", "AL", "RJ", "MS", "ES", "SE", "AM", "RO", "AC", "AP", "RR", "DF"],
    ColumnValues "uf" in ["MG", "SP", "RS", "BA", "PR", "SC", "GO", "PI", "PB", "MA", "PE", "CE", "RN", "PA", "MT", "TO", "AL"] with threshold >= 0.9,
    ColumnLength "uf" <= 2,
    Completeness "pib_per_capita" >= 0.99,
    StandardDeviation "pib_per_capita" between 26709.22 and 29520.72,
    ColumnValues "pib_per_capita" between 4923.04 and 591102.11,
    Completeness "coeficiente_gini" >= 0.99,
    ColumnValues "coeficiente_gini" <= 0.8,
    Completeness "investimento_publico" >= 0.99,
    StandardDeviation "investimento_publico" between 293302.68 and 324176.65,
    ColumnValues "investimento_publico" between 7.33 and 6229300,
    Completeness "ano_mes" >= 0.99,
    ColumnValues "ano_mes" between 202307 and 202309,
    Completeness "codigo_municipio_ibge" >= 0.99,
    Completeness "municipio_ibge" >= 0.99,
    ColumnLength "municipio_ibge" <= 32,
    Completeness "regiao" >= 0.99,
    ColumnValues "regiao" in ["NORDESTE", "SUDESTE", "SUL", "CENTRO-OESTE"] with threshold >= 0.91,
    ColumnLength "regiao" <= 13,
    Completeness "vl_pagador_pf" >= 0.99,
    StandardDeviation "vl_pagador_pf" between 802949497.57 and 887470497.32,
    ColumnValues "vl_pagador_pf" between 709056.74 and 48956181822.87,
    Completeness "qt_pagador_pf" >= 0.99,
    StandardDeviation "qt_pagador_pf" between 3440397.89 and 3802545.03,
    ColumnValues "qt_pagador_pf" between 3504 and 181159988,
    Completeness "vl_recebedor_pf" >= 0.99,
    StandardDeviation "vl_recebedor_pf" between 851395858.59 and 941016475.28,
    ColumnValues "vl_recebedor_pf" between 647725.14 and 52790700742.79,
    Completeness "qt_recebedor_pf" >= 0.99,
    StandardDeviation "qt_recebedor_pf" between 2522938.99 and 2788511.52,
    ColumnValues "qt_recebedor_pf" between 1862 and 129123830
]
```

## Evidência 40 – estatísticas e informações sobre a base pix\_idsc.

```
Rules = [
    RowCount between 2785 and 11140,
    Completeness "populacao_2022" >= 0.99,
    StandardDeviation "populacao_2022" between 196208.18 and 216861.68,
    ColumnValues "populacao_2022" between 832 and 11451246,
    Completeness "uf" >= 0.99,
    ColumnValues "uf" in ["MG", "SP", "RS", "BA", "PR", "SC", "GO", "PI", "PB", "MA", "PE", "CE", "RN", "PA", "MT", "TO", "AL", "RJ", "MS", "ES", "SE", "AM", "RO", "AC", "AP", "RR", "DF"],
    ColumnValues "uf" in ["MG", "SP", "RS", "BA", "PR", "SC", "GO", "PI", "PB", "MA", "PE", "CE", "RN", "PA", "MT", "TO", "AL"] with threshold >= 0.9,
    ColumnLength "uf" <= 2,
    Completeness "pib_per_capita" >= 0.99,
    StandardDeviation "pib_per_capita" between 26709.22 and 29520.72,
    ColumnValues "pib_per_capita" between 4923.04 and 591102.11,
    Completeness "coeficiente_gini" >= 0.99,
    ColumnValues "coeficiente_gini" <= 0.8,
    Completeness "investimento_publico" >= 0.99,
    StandardDeviation "investimento_publico" between 293302.68 and 324176.65,
    ColumnValues "investimento_publico" between 7.33 and 6229300,
    Completeness "ano_mes" >= 0.99,
    ColumnValues "ano_mes" between 202307 and 202309,
    Completeness "codigo_municipio_ibge" >= 0.99,
    Completeness "municipio_ibge" >= 0.99,
    ColumnLength "municipio_ibge" <= 32,
    Completeness "regiao" >= 0.99,
    ColumnValues "regiao" in ["NORDESTE", "SUDESTE", "SUL", "CENTRO-OESTE"] with threshold >= 0.91,
    ColumnLength "regiao" <= 13,
    Completeness "vl_pagador_pf" >= 0.99,
    StandardDeviation "vl_pagador_pf" between 802949497.57 and 887470497.32,
    ColumnValues "vl_pagador_pf" between 709056.74 and 48956181822.87,
    Completeness "qt_pagador_pf" >= 0.99,
    StandardDeviation "qt_pagador_pf" between 3440397.89 and 3802545.03,
    ColumnValues "qt_pagador_pf" between 3504 and 181159988,
    Completeness "vl_recebedor_pf" >= 0.99,
    StandardDeviation "vl_recebedor_pf" between 851395858.59 and 941016475.28,
    ColumnValues "vl_recebedor_pf" between 647725.14 and 52790700742.79,
    Completeness "qt_recebedor_pf" >= 0.99,
    StandardDeviation "qt_recebedor_pf" between 2522938.99 and 2788511.52,
    ColumnValues "qt_recebedor_pf" between 1862 and 129123830
]
```

### 3.3. Solução do Problema

A solução do problema, no contexto deste trabalho, refere-se à apresentação de respostas para as questões elencadas no item 1, conforme segue:

- Quais são as regiões do Brasil, Unidades da Federação e municípios que mais transacionaram recursos por Pix?

#### Query - Ranking por Região

The screenshot shows the AWS Athena Query Editor interface. The query is a single-line SELECT statement:

```
1 SELECT REGIAO, SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) AS VALOR_PAGADOR_PF
2 FROM PIX_IDSC
3 WHERE PIB_PER_CAPITA IS NOT NULL
4 GROUP BY REGIAO
5 ORDER BY VALOR_PAGADOR_PF DESC
```

The results table displays five rows of data:

| # | REGIAO       | VALOR_PAGADOR_PF |
|---|--------------|------------------|
| 1 | SUDESTE      | 266857391294.55  |
| 2 | NORDESTE     | 131147816872.73  |
| 3 | SUL          | 82410025904.98   |
| 4 | CENTRO-OESTE | 6603078993.13    |
| 5 | NORTE        | 46262660171.48   |

#### Query - Ranking das 10 maiores UF's por valor total transacionado por Pix

The screenshot shows the AWS Athena Query Editor interface. The query is a single-line SELECT statement:

```
1 SELECT REGIAO, UF, SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) AS VALOR_PAGADOR_PF
2 FROM PIX_IDSC
3 WHERE PIB_PER_CAPITA IS NOT NULL
4 GROUP BY REGIAO, UF
5 ORDER BY VALOR_PAGADOR_PF DESC
6 LIMIT 10
```

The results table displays five rows of data:

| # | REGIAO   | UF | VALOR_PAGADOR_PF |
|---|----------|----|------------------|
| 1 | SUDESTE  | SP | 14962226166.51   |
| 2 | SUDESTE  | MG | 57640254928.09   |
| 3 | SUDESTE  | RJ | 48399066765.17   |
| 4 | NORDESTE | BA | 35064102417.58   |
| 5 | SUL      | PR | 33550163531.76   |

## Query - Ranking dos 5 maiores municípios dentre as 10 maiores UF's

Screenshot of the AWS Athena Query Editor showing the execution of a complex SQL query. The query uses multiple CTEs (01\_RANKING\_POR..., 02\_RANKING\_POR\_UF, 03\_RANKING\_POR...) to rank municipalities by population and then filters the top 10 UFs, finally ranking the 5 largest municipalities within those UFs.

```

01_RANKING_POR... :: x | 02_RANKING_POR_UF :: x | 03_RANKING_POR... :: x | 04_RANKING_REL_P... :: x | 05_RANKING_REL_P... :: x | + | 
1 = WITH LISTA_UF AS (
2   SELECT UF, SUM(CAST(VL_PAGADOR_PF AS DECIMAL(28,2))) AS VALOR_PAGADOR_PF
3   FROM PIX_IDSC
4   GROUP BY UF
5   ORDER BY VALOR_PAGADOR_PF DESC
6   LIMIT 10),
7   MUNICIPIOS AS (
8     SELECT UF, MUNICIPIO_INGE AS MUNICIPIO, VL_PAGADOR_PF, RANK() OVER
9       (PARTITION BY UF ORDER BY VL_PAGADOR_PF DESC) AS RANK
10    FROM PIX_IDSC
11   WHERE PIB_PER_CAPITA IS NOT NULL
12
13   SELECT UF, MUNICIPIO, CAST(VL_PAGADOR_PF AS DECIMAL(28,2)) AS VALOR_PAGADOR_PF
14   FROM MUNICIPIOS
15  WHERE RANK <= 5
16  AND UF IN (SELECT DISTINCT UF FROM LISTA_UF)
17  ORDER BY UF, VL_PAGADOR_PF DESC
18
SQL  Ln 18, Col 32
Run again Explain Cancel Clear Create
Reuse query results up to 60 minutes ago

```

The results table shows the top 5 municipalities from Bahia (BA) with their respective population values:

| # | UF | MUNICIPIO            | VALOR_PAGADOR_PF |
|---|----|----------------------|------------------|
| 1 | BA | SALVADOR             | 9657050027.34    |
| 2 | BA | FEIRA DE SANTANA     | 2272744610.88    |
| 3 | BA | VITORIA DA CONQUISTA | 1028633631.85    |
| 4 | BA | CAMAÇARI             | 889029329.61     |
| 5 | BA | LAURO DE FREITAS     | 884709377.85     |

Screenshot of the AWS Athena Query Editor showing the execution of the same query, resulting in identical output. The results table shows the top 5 municipalities from Bahia (BA) with their respective population values.

| # | UF | MUNICIPIO            | VALOR_PAGADOR_PF |
|---|----|----------------------|------------------|
| 1 | BA | SALVADOR             | 9657050027.34    |
| 2 | BA | FEIRA DE SANTANA     | 2272744610.88    |
| 3 | BA | VITORIA DA CONQUISTA | 1028633631.85    |
| 4 | BA | CAMAÇARI             | 889029329.61     |
| 5 | BA | LAURO DE FREITAS     | 884709377.85     |

## Resultados Exportados

### RANKING POR REGIÃO

| REGIAO       | VALOR_PAGADOR_PF   |
|--------------|--------------------|
| SUDESTE      | 266.857.391.294,55 |
| NORDESTE     | 131.147.816.872,73 |
| SUL          | 82.410.025.904,98  |
| CENTRO-OESTE | 66.030.789.993,13  |
| NORTE        | 46.262.660.171,48  |

### RANKING POR REGIÃO E UF

| REGIAO       | UF | VALOR_PAGADOR_PF   |
|--------------|----|--------------------|
| SUDESTE      | SP | 149.622.226.166,31 |
| SUDESTE      | MG | 57.640.254.928,09  |
| SUDESTE      | RJ | 48.399.066.765,17  |
| NORDESTE     | BA | 35.864.102.417,58  |
| SUL          | PR | 33.550.163.531,76  |
| CENTRO-OESTE | GO | 28.204.552.625,51  |
| SUL          | RS | 27.927.547.263,58  |
| NORDESTE     | PE | 22.243.069.035,81  |
| NORDESTE     | CE | 21.235.226.897,13  |
| SUL          | SC | 20.932.315.109,64  |

### RANKING DOS 5 MAIORES MUNICÍPIOS DENTRE AS 10 MAIORES

| UF | MUNICIPIO                 | VALOR_PAGADOR_PF  |
|----|---------------------------|-------------------|
| BA | SALVADOR                  | 9.657.050.027,34  |
| BA | FEIRA DE SANTANA          | 2.272.744.610,88  |
| BA | VITÃ“RIA DA CONQUISTA     | 1.028.633.631,83  |
| BA | CAMAÃ‡ARI                 | 889.029.329,61    |
| BA | LAURO DE FREITAS          | 884.709.377,85    |
| CE | FORTALEZA                 | 9.244.079.384,21  |
| CE | CAUCAIA                   | 844.739.415,94    |
| CE | JUAZEIRO DO NORTE         | 744.297.342,35    |
| CE | MARACANAÃ§                | 645.680.539,09    |
| CE | SOBRAL                    | 465.667.342,37    |
| GO | GOIÃ,NIA                  | 8.877.093.008,99  |
| GO | APARECIDA DE GOIÃ,NIA     | 1.885.671.602,31  |
| GO | ANÃPOLIS                  | 1.682.981.757,38  |
| GO | RIO VERDE                 | 1.402.210.070,17  |
| GO | JATAÃ®                    | 675.160.206,43    |
| MG | BELO HORIZONTE            | 10.500.349.420,18 |
| MG | UBERLÃ,NDA                | 2.862.964.184,65  |
| MG | CONTAGEM                  | 2.227.245.927,82  |
| MG | JUIZ DE FORA              | 1.433.298.542,65  |
| MG | MONTES CLAROS             | 1.248.518.626,21  |
| PE | RECIFE                    | 5.859.488.337,14  |
| PE | JABOTATÃJO DOS GUARARAPES | 1.752.650.566,45  |
| PE | PETROLINA                 | 1.280.303.074,35  |
| PE | CARUARU                   | 1.167.039.562,91  |
| PE | OLINDA                    | 1.093.922.013,70  |
| PR | CURITIBA                  | 7.742.599.935,48  |
| PR | LONDrina                  | 2.061.151.767,46  |
| PR | MARINGÃ®                  | 1.748.252.884,15  |
| PR | CASCABEL                  | 1.198.909.855,09  |
| PR | PONTA GROSSA              | 1.032.674.400,46  |
| RJ | RIO DE JANEIRO            | 22.764.509.092,45 |
| RJ | DUQUE DE CAXIAS           | 2.264.248.700,40  |
| RJ | SÃO GONÃ‡ALO              | 2.172.210.626,96  |
| RJ | NITERÃ“I                  | 2.143.683.687,39  |
| RJ | NOVA IGUAÃ‡U              | 2.070.286.014,16  |
| RS | PORTO ALEGRE              | 5.312.205.459,80  |
| RS | CAXIAS DO SUL             | 1.361.616.414,41  |
| RS | CANOAS                    | 1.118.060.312,05  |
| RS | PELOTAS                   | 830.339.475,28    |
| RS | GRAVATAÃ®                 | 766.947.445,95    |
| SC | FLORIANÃ“POLIS            | 2.391.617.085,42  |
| SC | JOINVILLE                 | 1.998.140.000,87  |
| SC | BLUMENAU                  | 1.154.644.254,00  |
| SC | SÃO JOSÃ‰                 | 952.247.694,09    |
| SC | ITAJAÃ®                   | 933.408.545,86    |
| SP | SÃO PAULO                 | 48.956.181.821,87 |
| SP | GUARULHOS                 | 4.425.611.703,13  |
| SP | CAMPINAS                  | 4.321.288.248,12  |
| SP | SÃO BERNARDO DO CAMPO     | 2.776.288.437,65  |
| SP | RIBEIRÃƒO PRETO           | 2.723.869.934,57  |

Os resultados observados acima demonstram que a região Sudeste é a que mais transacionou recursos por Pix no mês de agosto de 2023, um total de R\$ 266.857.391.294,55. Nesse montante, estão contempladas as 3 maiores UF's em recursos transacionados por Pix em todo o Brasil — SP, MG e RJ, responsáveis por 96% dos valores transacionados na região Sudeste e 57% do total transacionado no Brasil.

Observando-se apenas o Estado de São Paulo, a maior UF no ranking nacional, nota-se que o município de São Paulo é o maior do Estado, apresentando um valor transacionado por Pix mais de 10 vezes maior que o do segundo colocado no ranking do Estado, o município de Guarulhos.

Se observarmos ainda o cenário nacional, perceberemos que o município de São Paulo apresenta um valor transacionado por Pix mais de 2 vezes maior que o do segundo colocado no ranking nacional, o município do Rio de Janeiro.

- Quais são as regiões do Brasil, Unidades da Federação e municípios que apresentam maior valor de recursos transacionados por Pix em relação ao PIB – Produto Interno Bruto das respectivas regiões, Unidades da Federação e municípios?

#### Query - Ranking das regiões por Pix em Relação ao PIB

```

SELECT REGIAO,
       SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) AS VALOR_PAGADOR_PF,
       SUM(CAST((PIB_PER_CAPITA * POPULACAO_2022) AS DECIMAL(20,2))) AS PIB_REGIAO,
       (SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) / SUM(CAST((PIB_PER_CAPITA * POPULACAO_2022) AS DECIMAL(20,2)))) * 100 AS REL_REGIAO_PIB
FROM PIX_DSC
WHERE PIB_PER_CAPITA IS NOT NULL
GROUP BY REGIAO
ORDER BY REL_REGIAO_PIB DESC;
    
```

The screenshot shows the AWS Athena Query Editor interface. The query above is being run against the 'mvp\_database' on the 'pix\_ids' table. The results are displayed in a table titled 'Results (5)' with columns: #, REGIAO, VALOR\_PAGADOR\_PF, PIB\_REGIAO, and REL\_REGIAO\_PIB. The data shows the following ranking:

| # | REGIAO       | VALOR_PAGADOR_PF | PIB_REGIAO       | REL_REGIAO_PIB |
|---|--------------|------------------|------------------|----------------|
| 1 | NORDESTE     | 131147816872.73  | 1035972620985.43 | 13,00          |
| 2 | NORTE        | 46262660171.48   | 482468049160.89  | 10,00          |
| 3 | CENTRO-OESTE | 66030789993.13   | 782280809196.56  | 8,00           |
| 4 | SUDESTE      | 266857391294.55  | 3812423719030.97 | 7,00           |

#### Query - Ranking das UF's por Pix em Relação ao PIB

```

SELECT REGIAO, UF,
       SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) AS VALOR_PAGADOR_PF,
       SUM(CAST((PIB_PER_CAPITA * POPULACAO_2022) AS DECIMAL(20,2))) AS PIB_REGIAO,
       (SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) / SUM(CAST((PIB_PER_CAPITA * POPULACAO_2022) AS DECIMAL(20,2)))) * 100 AS REL_REGIAO_PIB
FROM PIX_DSC
WHERE PIB_PER_CAPITA IS NOT NULL
GROUP BY REGIAO, UF
ORDER BY REL_REGIAO_PIB DESC
LIMIT 10;
    
```

The screenshot shows the AWS Athena Query Editor interface. The query above is being run against the 'mvp\_database' on the 'pix\_ids' table. The results are displayed in a table titled 'Results (10)' with columns: #, REGIAO, UF, VALOR\_PAGADOR\_PF, PIB\_REGIAO, and REL\_REGIAO\_PIB. The data shows the following ranking:

| # | REGIAO       | UF | VALOR_PAGADOR_PF | PIB_REGIAO         | REL_REGIAO_PIB |
|---|--------------|----|------------------|--------------------|----------------|
| 1 | NORDESTE     | MA | 1517471950,59    | 10302568011,81     | 15,00          |
| 2 | CENTRO-OESTE | GO | 28204552625,51   | 222.041.571.524,22 | 13,00          |
| 3 | NORDESTE     | CE | 21235226897,13   | 161.323.848.630,19 | 13,00          |
| 4 | NORTE        | AC | 1967276086,79    | 15168715614,00     | 13,00          |

#### Resultados Exportados

RANKING DAS REGIÕES POR PIX EM RELAÇÃO AO PIB

| REGIAO       | VALOR_PAGADOR_PF   | PIB_REGIAO           | REL_REGIAO_PIB |
|--------------|--------------------|----------------------|----------------|
| NORDESTE     | 131.147.816.872,73 | 1.035.972.620.983,43 | 13,00          |
| NORTE        | 46.262.660.171,48  | 482.468.049.160,89   | 10,00          |
| CENTRO-OESTE | 66.030.789.993,13  | 782.280.809.196,56   | 8,00           |
| SUDESTE      | 266.857.391.294,55 | 3.812.423.719.030,97 | 7,00           |
| SUL          | 82.410.025.904,98  | 1.306.287.612.831,25 | 6,00           |

RANKING DAS UF'S POR PIX EM RELAÇÃO AO PIB

| REGIAO       | UF | VALOR_PAGADOR_PF  | PIB_REGIAO         | REL_REGIAO_PIB |
|--------------|----|-------------------|--------------------|----------------|
| NORDESTE     | MA | 15.174.741.950,59 | 103.025.680.011,81 | 15,00          |
| CENTRO-OESTE | GO | 28.204.552.625,51 | 222.041.571.524,22 | 13,00          |
| NORDESTE     | CE | 21.235.226.897,13 | 161.323.848.630,19 | 13,00          |
| NORTE        | AC | 1.967.276.086,79  | 15.168.715.614,00  | 13,00          |
| NORTE        | RO | 5.765.505.386,94  | 45.825.841.526,86  | 13,00          |
| NORTE        | AP | 2.048.359.762,03  | 15.728.422.921,57  | 13,00          |
| NORDESTE     | SE | 5.686.954.827,40  | 42.998.232.033,42  | 13,00          |
| NORDESTE     | PB | 9.164.255.778,28  | 70.170.260.706,11  | 13,00          |
| NORDESTE     | PI | 6.588.046.617,35  | 56.956.580.753,76  | 12,00          |
| NORDESTE     | AL | 7.127.751.673,15  | 59.808.706.377,84  | 12,00          |

Observa-se nos dados acima que, considerando-se a relação entre o valor transacionado por Pix e o PIB das regiões do Brasil, há algumas inversões de posição no *ranking* das regiões, como no caso da região Sudeste, que detinha o primeiro lugar e passou a ser a penúltima, ou como no caso da região Norte, que passou do último para o segundo lugar do *ranking*.

Destaca-se que a região Nordeste, que passou ao primeiro lugar do *ranking*, apresenta uma relação de 13% entre o valor transacionado por Pix e o PIB da região, quase duas vezes maior que a relação medida para a região Sudeste, de 7%.

Por sua vez, o *ranking* das Unidades da Federação não apresenta grandes diferenças entre os Estados, porém, observa-se que 6 dentre os 10 primeiros colocados do *ranking* são Estados da região Nordeste, com destaque para o Maranhão, primeiro colocado no *ranking*.

- Existe relação positiva entre os maiores pagadores e os maiores recebedores de recursos transacionados por Pix?

#### *Query - Comparar o Ranking das UF's pagadoras com as recebedoras - parte 1*

```

WITH TAB_UF AS (SELECT 'PAGADOR' AS ORIGEM, UF, SUM(CAST(VL_PAGADOR_PF AS DECIMAL(20,2))) AS VALOR_PAGADOR_PF
FROM PIX_IDSC
WHERE PIB_PER_CAPITA IS NOT NULL
GROUP BY UF)
SELECT ORIGEM, UF, VALOR_PAGADOR_PF, RANK() OVER
(PARTITION BY ORIGEM ORDER BY VALOR_PAGADOR_PF DESC) AS RANK
FROM TAB_UF
    
```

| # | ORIGEM  | UF | VALOR_PAGADOR_PF | RANK |
|---|---------|----|------------------|------|
| 1 | PAGADOR | SP | 14962226166.31   | 1    |
| 2 | PAGADOR | MG | 57640254928.09   | 2    |
| 3 | PAGADOR | RJ | 48399066765.17   | 3    |
| 4 | PAGADOR | PA | 77654103417.50   | 4    |

## Query - Comparar o Ranking das UF's pagadoras com as recebedoras - parte 2

```

WITH TAB_UF AS (SELECT 'RECEBEDOR' AS ORIGEM, UF, SUM(CAST(VL_RECEBEDOR_PF AS DECIMAL(20,2))) AS VALOR_RECEBEDOR_PF
FROM PIX_IDSC
WHERE PIX_PER_CAPITA IS NOT NULL
GROUP BY UF)
SELECT ORIGEM, UF, VALOR_RECEBEDOR_PF, RANK() OVER (PARTITION BY ORIGEM ORDER BY VALOR_RECEBEDOR_PF DESC) AS RANK
FROM TAB_UF
    
```

The screenshot shows the AWS Management Console with the Athena service selected. The query editor interface is open, displaying the SQL code. The results pane shows a table with columns: #, ORIGEM, UF, VALOR\_RECEBEDOR\_PF, and RANK. The data is as follows:

| #  | ORIGEM    | UF | VALOR_RECEBEDOR_PF | RANK |
|----|-----------|----|--------------------|------|
| 1  | RECEBEDOR | SP | 160893644577.76    | 1    |
| 2  | RECEBEDOR | MG | 60055939615.87     | 2    |
| 3  | RECEBEDOR | RJ | 49017039197.17     | 3    |
| 4  | RECEBEDOR | PA | 766074414E42.5C    | 4    |
| 5  | PAGADOR   | SP | 149.622.226.166,31 | 1    |
| 6  | PAGADOR   | MG | 57.640.254.928,09  | 2    |
| 7  | PAGADOR   | RJ | 48.399.066.765,17  | 3    |
| 8  | PAGADOR   | BA | 35.864.102.417,58  | 4    |
| 9  | PAGADOR   | PR | 33.550.163.531,76  | 5    |
| 10 | PAGADOR   | GO | 28.204.552.625,51  | 6    |
| 11 | PAGADOR   | RS | 27.927.547.263,58  | 7    |
| 12 | PAGADOR   | PE | 22.243.069.035,81  | 8    |
| 13 | PAGADOR   | CE | 21.235.226.897,13  | 9    |
| 14 | PAGADOR   | SC | 20.932.315.109,64  | 10   |
| 15 | PAGADOR   | PA | 19.993.173.781,75  | 11   |
| 16 | PAGADOR   | MT | 15.351.487.811,48  | 12   |
| 17 | PAGADOR   | MA | 15.174.741.950,59  | 13   |
| 18 | PAGADOR   | DF | 12.991.700.917,26  | 14   |
| 19 | PAGADOR   | ES | 11.195.843.434,98  | 15   |
| 20 | PAGADOR   | MS | 9.483.048.638,88   | 16   |
| 21 | PAGADOR   | AM | 9.313.812.475,96   | 17   |
| 22 | PAGADOR   | PB | 9.164.255.778,28   | 18   |
| 23 | PAGADOR   | RN | 8.063.667.675,44   | 19   |
| 24 | PAGADOR   | AL | 7.127.751.673,15   | 20   |
| 25 | PAGADOR   | PI | 6.588.046.617,35   | 21   |
| 26 | PAGADOR   | RO | 5.765.505.386,94   | 22   |
| 27 | PAGADOR   | SE | 5.686.954.827,40   | 23   |
| 28 | PAGADOR   | TO | 5.168.131.457,00   | 24   |
| 29 | PAGADOR   | AP | 2.048.359.762,03   | 25   |
| 30 | PAGADOR   | RR | 2.006.401.221,01   | 26   |
| 31 | PAGADOR   | AC | 1.967.276.086,79   | 27   |

### Resultados Exportados

| ORIGEM  | UF | VALOR_PAGADOR_PF   | RANK | ORIGEM    | UF | VALOR_RECEBEDOR_PF | RANK |
|---------|----|--------------------|------|-----------|----|--------------------|------|
| PAGADOR | SP | 149.622.226.166,31 | 1    | RECEBEDOR | SP | 160.893.644.577,76 | 1    |
| PAGADOR | MG | 57.640.254.928,09  | 2    | RECEBEDOR | MG | 60.055.939.615,87  | 2    |
| PAGADOR | RJ | 48.399.066.765,17  | 3    | RECEBEDOR | RJ | 49.017.039.197,17  | 3    |
| PAGADOR | BA | 35.864.102.417,58  | 4    | RECEBEDOR | BA | 36.592.414.518,56  | 4    |
| PAGADOR | PR | 33.550.163.531,76  | 5    | RECEBEDOR | PR | 35.725.324.607,74  | 5    |
| PAGADOR | GO | 28.204.552.625,51  | 6    | RECEBEDOR | GO | 29.691.622.150,41  | 6    |
| PAGADOR | RS | 27.927.547.263,58  | 7    | RECEBEDOR | RS | 28.103.330.291,20  | 7    |
| PAGADOR | PE | 22.243.069.035,81  | 8    | RECEBEDOR | PE | 22.512.563.103,95  | 8    |
| PAGADOR | CE | 21.235.226.897,13  | 9    | RECEBEDOR | SC | 22.269.025.465,43  | 9    |
| PAGADOR | SC | 20.932.315.109,64  | 10   | RECEBEDOR | CE | 21.468.358.351,19  | 10   |
| PAGADOR | PA | 19.993.173.781,75  | 11   | RECEBEDOR | PA | 19.948.413.900,22  | 11   |
| PAGADOR | MT | 15.351.487.811,48  | 12   | RECEBEDOR | MT | 16.338.110.903,58  | 12   |
| PAGADOR | MA | 15.174.741.950,59  | 13   | RECEBEDOR | MA | 14.747.285.773,16  | 13   |
| PAGADOR | DF | 12.991.700.917,26  | 14   | RECEBEDOR | DF | 13.277.584.598,05  | 14   |
| PAGADOR | ES | 11.195.843.434,98  | 15   | RECEBEDOR | ES | 11.310.977.163,20  | 15   |
| PAGADOR | MS | 9.483.048.638,88   | 16   | RECEBEDOR | MS | 9.823.462.124,25   | 16   |
| PAGADOR | AM | 9.313.812.475,96   | 17   | RECEBEDOR | AM | 9.211.081.768,33   | 17   |
| PAGADOR | PB | 9.164.255.778,28   | 18   | RECEBEDOR | PB | 9.037.098.327,09   | 18   |
| PAGADOR | RN | 8.063.667.675,44   | 19   | RECEBEDOR | RN | 7.959.355.440,85   | 19   |
| PAGADOR | AL | 7.127.751.673,15   | 20   | RECEBEDOR | AL | 7.015.320.314,31   | 20   |
| PAGADOR | PI | 6.588.046.617,35   | 21   | RECEBEDOR | PI | 6.383.758.606,23   | 21   |
| PAGADOR | RO | 5.765.505.386,94   | 22   | RECEBEDOR | RO | 5.814.055.372,73   | 22   |
| PAGADOR | SE | 5.686.954.827,40   | 23   | RECEBEDOR | SE | 5.664.840.529,79   | 23   |
| PAGADOR | TO | 5.168.131.457,00   | 24   | RECEBEDOR | TO | 5.211.302.061,37   | 24   |
| PAGADOR | AP | 2.048.359.762,03   | 25   | RECEBEDOR | AP | 2.025.171.461,72   | 25   |
| PAGADOR | RR | 2.006.401.221,01   | 26   | RECEBEDOR | RR | 1.964.568.215,55   | 26   |
| PAGADOR | AC | 1.967.276.086,79   | 27   | RECEBEDOR | AC | 1.949.797.750,17   | 27   |

Observa-se que existe relação positiva entre os maiores pagadores e os maiores recebedores em 93% dos casos, ou seja, somente duas UF's não mantêm suas posições nos dois rankings apresentados, CE e SC, que inverteram suas posições. Os demais Estados, 25 dentre os 27, mantêm suas posições nos rankings de maiores pagadores e de maiores recebedores de recursos transacionados por Pix.

De forma geral, os resultados obtidos permitem identificar as maiores regiões, Unidades da Federação e municípios, considerando-se o volume de recursos transacionados por Pix, o que possibilita a geração de *insights* e a elaboração de hipóteses quanto às diferenças e aos montantes de recursos observados. Por exemplo, o comparativo dos *rankings* de pagadores e recebedores poderia indicar que existe transferência de recursos entre diferentes Estados de forma desproporcional, o que não se mostrou evidente. Pelo contrário, há um comportamento similar entre todas as Unidades da Federação.

## 4. Autoavaliação

Os objetivos estabelecidos neste trabalho foram alcançados, ou seja, as respostas elaborados no início foram respondidas com certo grau de profundidade. Porém, o trabalho foi realizado com um nível de dificuldade superior ao esperado.

Para a realização das etapas pertinentes ao *pipeline* de dados, inicialmente, foi utilizado o Google Cloud que, apesar de apresentar configurações e etapas simplificadas, apresentou diversos erros quando da execução do *job* para o ETL. Sem sucesso nessa etapa e com os recursos gratuitos quase inteiramente consumidos, a solução foi a realização do *pipeline* de dados no Amazon AWS.

O Amazon AWS, por sua vez, mostrou-se mais exigente quanto à configuração das etapas do processo, mas, nem por isso, as estapas apresentaram difícil entendimento.

No entanto, com o *pipeline* criado, tendo como destino final o Redshift, o mesmo problema foi encontrado quando da execução do *job* para o ETL, ou seja, diversos erros de difícil interpretação foram encontrados.

Ainda com recursos gratuitos para exploração da ferramenta, e sem conseguir alcançar qualquer entendimento que fizesse o Redshift funcionar, uma nova solução foi pensada, a configuração de um Bucket S3 como destino final do processo de ETL e posterior realização de consultas em SQL por meio do AWS Athena, o que possibilitaria a realização da etapa final do trabalho, a análise dos dados para solução do problema inicialmente estabelecido. Para essa solução, um conjunto de dados com menos registros foi utilizado para execução do *pipeline*, visando a economia de recursos computacionais e financeiros.

A solução adotada se mostrou correta e eficiente, proporcionando a execução de todo o processo com sucesso, em um prazo curto de tempo e com os recursos gratuitos da Amazon AWS.

Assim, visando melhorias futuras, poderia ser utilizado um conjunto de dados mais completo, abrangendo, pelo menos, 12 meses de transações por Pix, o que proporcionaria melhores análises e mais *insights*. Porém, seria mais adequada a utilização do Redshift para o processamento desse conjunto de dados.

Seria interessante também a seleção de outros indicadores do IDSC, a fim de realizar análises estatísticas mais apuradas como um mapa de calor das correlações existentes entre as diversas variáveis presentes na base completa do IDSC.