

```

{
  "cells": [
    {
      "cell_type": "markdown",
      "metadata": {
        "application/vnd.databricks.v1+cell": {
          "cellMetadata": {},
          "inputWidgets": {},
          "nuid": "c8c64ea1-339c-483e-bb8a-bd9072207e32",
          "showTitle": false,
          "tableResultSettingsMap": {},
          "title": ""
        }
      },
      "source": [
        "## Tabela: refined.d_empresas_farol\n",
        "\n",
        "### Objetivo:\n",
        "Tabela da dimensÃo de empresas do indicador farol.\n",
        "\n",
        "### Fontes de Dados\n",
        "\n",
        "
        | Origem
        |-----|
        | Trusted.tb_tab_empresa_farol | Tabela de
        | raw.sharepoint_dados_filiais_farol | Tabela do sharepoint com
        | refined.tb_farol_faturamento_farol | Tabela com dados de
        |-----|
        | Data | Desenvolvido por | ModificaÃes
        |-----|-----|-----|
        | 22/05/2025 | Michel Santana | CriaÃo do notebook |
      ],
      "outputs": [],
      "source": [
        "# Importa e executa o notebook `ingestion_function`, localizado em `../
        00_config/`. \n",
        "# \n",
        "# O comando `%run` carrega todas as funÃes, variÃveis e
        configuraÃes definidas no notebook referenciado\n",

```

```

    "# para o ambiente atual. Isso permite reutilizar lógica comum, como
funções de ingestão de dados, sem duplicação de código.\n",
    "# \n",
    "# Estilo para centralizar rotinas reutilizáveis e manter notebooks
modulares e organizados."
]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "464363cf-eb80-4c2a-bd2b-913951b53da3",
            "showTitle": false,
            "tableResultSettingsMap": {},
            "title": ""
        }
    },
    "outputs": [],
    "source": [
        "%run ../00_config/ingestion_function"
    ]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "c5b09eb8-58ba-4455-b75a-e0c5f17b557d",
            "showTitle": false,
            "tableResultSettingsMap": {},
            "title": ""
        }
    },
    "outputs": [],
    "source": [
        "debug = False\n",
        "\n",
        "container_target = 'refined'\n",
        "directory = 'farol'\n",
        "table_name = 'd_empresas_farol'\n",
        "delta_table_name = f'{environment}.{container_target}.{table_name}'\n",
        "delta_file = f\"abfss://{container_target}@{data_lake_name}.dfs.core.wi
ndows.net/{directory}/{table_name}/\"\n",
        "comment_delta_table = 'Tabela de dimensão de Empresas...'\n",
        "\n",
        "print(f'delta_table_name = {table_name}')\n",
        "print(f'delta_file = {delta_file}')"
    ]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "3e503987-e2f8-4717-b6cb-edce0921e7af",
            "showTitle": false,
            "tableResultSettingsMap": {},

```

```

        "title": ""
    }
},
"outputs": [],
"source": [
    "\"\"\"\n",
    "Cria um widget interativo chamado `reprocessar` para controle da carga
da dimensÃo.\n",
    "\n",
    "- Exibe um dropdown com as opÃ§Ães \"True\" e \"False\".\n",
    "- A variÃvel `reprocessar` serÃ `True` apenas se o usuÃrio
selecionar essa opÃÃo no notebook.\n",
    "\n",
    "Objetivo: permitir que o usuÃrio escolha, de forma interativa, se a
carga serÃ completa (`overwrite`) ou incremental (`merge`).\n",
    "\"\"\"\n",
    "\n",
    "dbutils.widgets.dropdown(\"reprocessar\", \"False\", [\"True\",
\"False\"], \"Reprocessar dimensÃo?\")\n",
    "reprocessar = dbutils.widgets.get(\"reprocessar\") == \"True\"
]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "a419ed1e-3d79-49eb-b681-00709176682d",
            "showTitle": false,
            "tableResultSettingsMap": {},
            "title": ""
        }
    },
    "outputs": [],
    "source": [
        "spark.sql(f\"\"\"\n",
        "CREATE TABLE IF NOT EXISTS {delta_table_name} (\n",
        "    sk_empresas BIGINT GENERATED ALWAYS AS IDENTITY (START WITH 1
INCREMENT BY 1),\n",
        "    codigo_empresa          INT,\n",
        "    cnpj_empresa            STRING,\n",
        "    razao_social            STRING,\n",
        "    nome_empresa            STRING,\n",
        "    des_logradouro          STRING,\n",
        "    num_insc_estadual       STRING,\n",
        "    num_insc_municipal      STRING,\n",
        "    des_complemento         STRING,\n",
        "    num_cep                 STRING,\n",
        "    nom_bairro              STRING,\n",
        "    cod_cidade              INT,\n",
        "    area                    STRING,\n",
        "    gerente_area            STRING,\n",
        "    porte                   STRING,\n",
        "    data_abertura_cupom     DATE,\n",
        "    data_abertura_empresa   STRING,\n",
        "    dias_funcionamento     STRING,\n",
        "    horario_funcionamento  STRING,\n",
        "    categoria               STRING,\n",
        "    possui_gnv              STRING,\n",
        "    possui_etanol           STRING,\n",
        "    possui_supervisor       STRING,\n",
        "    quadro_aprovado         STRING,

```

```

        pista_diesel          STRING,\n",
        troca_oleo            STRING,\n",
        sss                    STRING,\n",
        bandeira               STRING,\n",
        e_mail                  STRING,\n",
        e_mail_ga               STRING,\n",
        m2_loja                 STRING,\n",
        status                  STRING,\n",
        data_inativacao         STRING,\n",
        insert_date              TIMESTAMP,\n",
        update_date              TIMESTAMP\n",
    ") \n",
    "USING DELTA\n",
    "LOCATION '{delta_file}'\n",
    "COMMENT '{comment_delta_table}';\n",
    "\"\"\"\n",
    "\n"
]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "e5af8779-54c7-4643-a189-c685cfb1bd78",
            "showTitle": false,
            "tableResultSettingsMap": {},
            "title": ""
        }
    },
    "outputs": [],
    "source": [
        "spark.sql(f\"\"\"\n",
        "\n",
        "select\n",
        "    cod_empresa\n",
        "    ,num_cnpj\n",
        "    ,nom_fantasia\n",
        "    ,nom_razao_social\n",
        "    ,des_logradouro\n",
        "    ,num_insc_estadual\n",
        "    ,num_insc_municipal\n",
        "    ,des_complemento\n",
        "    ,num_cep\n",
        "    ,nom_bairro\n",
        "    ,cod_cidade\n",
        "from {environment}.trusted.tb_tab_empresa_farol\n",
        "where 1=1\n",
        "\"\"\".createOrReplaceTempView('tab_empresa')
    ]
},
{
    "cell_type": "code",
    "execution_count": 0,
    "metadata": {
        "application/vnd.databricks.v1+cell": {
            "cellMetadata": {},
            "inputWidgets": {},
            "nuid": "6bc22798-58e2-410f-86d7-80e34ac8c9be",
            "showTitle": false,
            "tableResultSettingsMap": {},
            "title": ""
        }
    },

```

```

    }
  },
  "outputs": [],
  "source": [
    "spark.sql(f\"\"\n",
    "\n",
    "select\n",
    "    codigo\n",
    "    ,area\n",
    "    ,porte\n",
    "    ,m2_loja\n",
    "    ,area\n",
    "    ,ga \n",
    "    ,porte \n",
    "    ,data_abertura\n",
    "    ,dias_de_func \n",
    "    ,horario_de_func \n",
    "    ,categoria\n",
    "    ,possui_gnv\n",
    "    ,possui_etanol\n",
    "    ,possui_supervisor\n",
    "    ,quadro_aprovado\n",
    "    ,pista_diesel\n",
    "    ,troca_oleo\n",
    "    ,sss\n",
    "    ,bandeira\n",
    "    ,e_mail\n",
    "    ,e_mail_ga\n",
    "    ,m2_loja\n",
    "    ,status\n",
    "    ,data_inativacao\n",
    "from {environment}.raw.sharepoint_dados_filiais\n",
    "where 1=1;\n",
    "\n",
    "\"\"\".createOrReplaceTempView('dados_filiais')\"
  ]
},
{
  "cell_type": "code",
  "execution_count": 0,
  "metadata": {
    "application/vnd.databricks.v1+cell": {
      "cellMetadata": {},
      "inputWidgets": {},
      "nuid": "402b3a3e-ec56-47bc-8750-1d9f0640f211",
      "showTitle": false,
      "tableResultSettingsMap": {},
      "title": ""
    }
  },
  "outputs": [],
  "source": [
    "spark.sql(f\"\"\n",
    "\n",
    "select\n",
    "    cod_empresa\n",
    "    ,to_date(min(data), 'yyyy-MM-dd') as data_abertura_filial\n",
    "from {environment}.refined.tb_farol_faturamento\n",
    "where 1=1\n",
    "group by\n",
    "    cod_empresa;\n",
    "\n",
    "\"\"\".createOrReplaceTempView('tab_faturamento')\"
  ]
}

```

```

},
{
  "cell_type": "code",
  "execution_count": 0,
  "metadata": {
    "application/vnd.databricks.v1+cell": {
      "cellMetadata": {},
      "inputWidgets": {},
      "nuid": "f511a0be-7ef8-4bd0-842e-cf043799e637",
      "showTitle": false,
      "tableResultSettingsMap": {},
      "title": ""
    }
  },
  "outputs": [],
  "source": [
    "source_df = spark.sql(\"\"\"\n",
    "    \n",
    "select \n",
    "    emp.cod_empresa as codigo_empresa\n",
    "    ,emp.num_cnpj as cnpj_empresa\n",
    "    ,emp.nom_razao_social as razao_social\n",
    "    ,emp.nom_fantasia as nome_empresa\n",
    "    ,emp.des_logradouro\n",
    "    ,emp.num_insc_estadual\n",
    "    ,emp.num_insc_municipal\n",
    "    ,emp.des_complemento\n",
    "    ,emp.num_cep\n",
    "    ,emp.nom_bairro\n",
    "    ,emp.cod_cidade\n",
    "    ,fil.area\n",
    "    ,fil.ga as gerente_area\n",
    "    ,fil.porte as porte\n",
    "    ,fat.data_abertura_filial as data_abertura_cupom\n",
    "    ,fil.data_abertura as data_abertura_empresa\n",
    "    ,fil.dias_de_func as dias_funcionamento\n",
    "    ,fil.horario_de_func as horario_funcionamento\n",
    "    ,fil.categoria\n",
    "    ,fil.possui_gnv\n",
    "    ,fil.possui_etanol\n",
    "    ,fil.possui_supervisor\n",
    "    ,fil.quadro_aprovado\n",
    "    ,fil.pista_diesel\n",
    "    ,fil.troca_oleo\n",
    "    ,fil.sss\n",
    "    ,fil.bandeira\n",
    "    ,fil.e_mail\n",
    "    ,fil.e_mail_ga\n",
    "    ,fil.m2_loja\n",
    "    ,fil.status\n",
    "    ,fil.data_inativacao\n",
    "from tab_empresa emp\n",
    "left join dados_filiais fil\n",
    "    on emp.cod_empresa = fil.codigo\n",
    "left join tab_faturamento fat\n",
    "    on emp.cod_empresa = fat.cod_empresa\n",
    "where 1=1\n",
    "order by emp.nom_fantasia\n",
    "\"\"\"\n",
    "\n",
    "source_df.createOrReplaceTempView('source_df')
  ]
},
{

```

```

"cell_type": "code",
"execution_count": 0,
"metadata": {
  "application/vnd.databricks.v1+cell": {
    "cellMetadata": {},
    "inputWidgets": {},
    "nuid": "0931f8ff-8522-4291-b376-864f56ffe04c",
    "showTitle": false,
    "tableResultSettingsMap": {},
    "title": ""
  }
},
"outputs": [],
"source": [
  "if reprocessar:\n",
  "    source_df = source_df.withColumn(\"insert_date\",
lit(current_timestamp() ) )\n",
  "    source_df.write.format(\"delta\")\\\n",
  "        .mode(\"overwrite\")\\\n",
  "        .saveAsTable(f\"{delta_table_name}\", path=f\"{delta_file}\")\n",
  "    print(f\"Carga overwrite realizada com sucesso! {delta_table_name}\")
\n",
  "else:\n",
  "    print(\"Realizando carga em Merge.\")\n",
  "    spark.sql(f\"\"\"\n",
  "\n",
  "MERGE INTO {delta_table_name} AS target\n",
  "USING source_df AS source\n",
  "    ON target.codigo_empresa = source.codigo_empresa\n",
  "WHEN MATCHED THEN\n",
  "    UPDATE SET\n",
  "        target.cnpj_empresa           = source.cnpj_empresa,\n",
  "        target.razao_social            = source.razao_social,\n",
  "        target.nome_empresa            = source.nome_empresa,\n",
  "        target.des_logradouro          = source.des_logradouro,\n",
  "        target.num_insc_estadual       = source.num_insc_estadual,\n",
  "        target.num_insc_municipal      = source.num_insc_municipal,\n",
  "        target.des_complemento        = source.des_complemento,\n",
  "        target.num_cep                 = source.num_cep,\n",
  "        target.nom_bairro              = source.nom_bairro,\n",
  "        target.cod_cidade              = source.cod_cidade,\n",
  "        target.area                   = source.area,\n",
  "        target.gerente_area            = source.gerente_area,\n",
  "        target.porte                   = source.porte,\n",
  "        target.data_abertura_cupom     = source.data_abertura_cupom,\n",
  "        target.data_abertura_empresa  = source.data_abertura_empresa,\n",
  "        target.dias_funcionamento     = source.dias_funcionamento,\n",
  "        target.horario_funcionamento = source.horario_funcionamento,\n",
  "        target.categoria               = source.categoria,\n",
  "        target.possui_gnv              = source.possui_gnv,\n",
  "        target.possui_etanol           = source.possui_etanol,\n",
  "        target.possui_supervisor       = source.possui_supervisor,\n",
  "        target.quadro_aprovado         = source.quadro_aprovado,\n",
  "        target.pista_diesel            = source.pista_diesel,\n",
  "        target.troca_oleo              = source.troca_oleo,\n",
  "        target.sss                     = source.sss,\n",
  "        target.bandeira                = source.bandeira,\n",
  "        target.e_mail                  = source.e_mail,\n",
  "        target.e_mail_ga               = source.e_mail_ga,\n",
  "        target.m2_loja                 = source.m2_loja,\n",
  "        target.status                  = source.status,\n",
  "        target.data_inativacao         = source.data_inativacao,\n",
  "        target.update_date             = current_timestamp()\n",
  "WHEN NOT MATCHED THEN\n",

```

```

" INSERT (\n",
"     codigo_empresa,\n",
"     cnpj_empresa,\n",
"     razao_social,\n",
"     nome_empresa,\n",
"     des_logradouro,\n",
"     num_insc_estadual,\n",
"     num_insc_municipal,\n",
"     des_complemento,\n",
"     num_cep,\n",
"     nom_bairro,\n",
"     cod_cidade,\n",
"     area,\n",
"     gerente_area,\n",
"     porte,\n",
"     data_abertura_cupom,\n",
"     data_abertura_empresa,\n",
"     dias_funcionamento,\n",
"     horario_funcionamento,\n",
"     categoria,\n",
"     possui_gnv,\n",
"     possui_etanol,\n",
"     possui_supervisor,\n",
"     quadro_aprovado,\n",
"     pista_diesel,\n",
"     troca_oleo,\n",
"     sss,\n",
"     bandeira,\n",
"     e_mail,\n",
"     e_mail_ga,\n",
"     m2_loja,\n",
"     status,\n",
"     data_inativacao,\n",
"     insert_date\n",
" )\n",
" VALUES (\n",
"     source.codigo_empresa,\n",
"     source.cnpj_empresa,\n",
"     source.razao_social,\n",
"     source.nome_empresa,\n",
"     source.des_logradouro,\n",
"     source.num_insc_estadual,\n",
"     source.num_insc_municipal,\n",
"     source.des_complemento,\n",
"     source.num_cep,\n",
"     source.nom_bairro,\n",
"     source.cod_cidade,\n",
"     source.area,\n",
"     source.gerente_area,\n",
"     source.porte,\n",
"     source.data_abertura_cupom,\n",
"     source.data_abertura_empresa,\n",
"     source.dias_funcionamento,\n",
"     source.horario_funcionamento,\n",
"     source.categoria,\n",
"     source.possui_gnv,\n",
"     source.possui_etanol,\n",
"     source.possui_supervisor,\n",
"     source.quadro_aprovado,\n",
"     source.pista_diesel,\n",
"     source.troca_oleo,\n",
"     source.sss,\n",
"     source.bandeira,\n",
"     source.e_mail,\n",

```



```

        "    source.e_mail_ga,\n",
        "    source.m2_loja,\n",
        "    source.status,\n",
        "    source.data_inativacao,\n",
        "    current_timestamp()\n",
        "    )\n",
        "    \n",
        "    \"\"\"\"\"")
    ]
}
],
"metadata": {
  "application/vnd.databricks.v1+notebook": {
    "computePreferences": null,
    "dashboards": [],
    "environmentMetadata": null,
    "inputWidgetPreferences": null,
    "language": "python",
    "notebookMetadata": {
      "pythonIndentUnit": 4
    },
    "notebookName": "d_empresas_farol",
    "widgets": {}
  },
  "language_info": {
    "name": "python"
  }
},
"nbformat": 4,
"nbformat_minor": 0
}

```