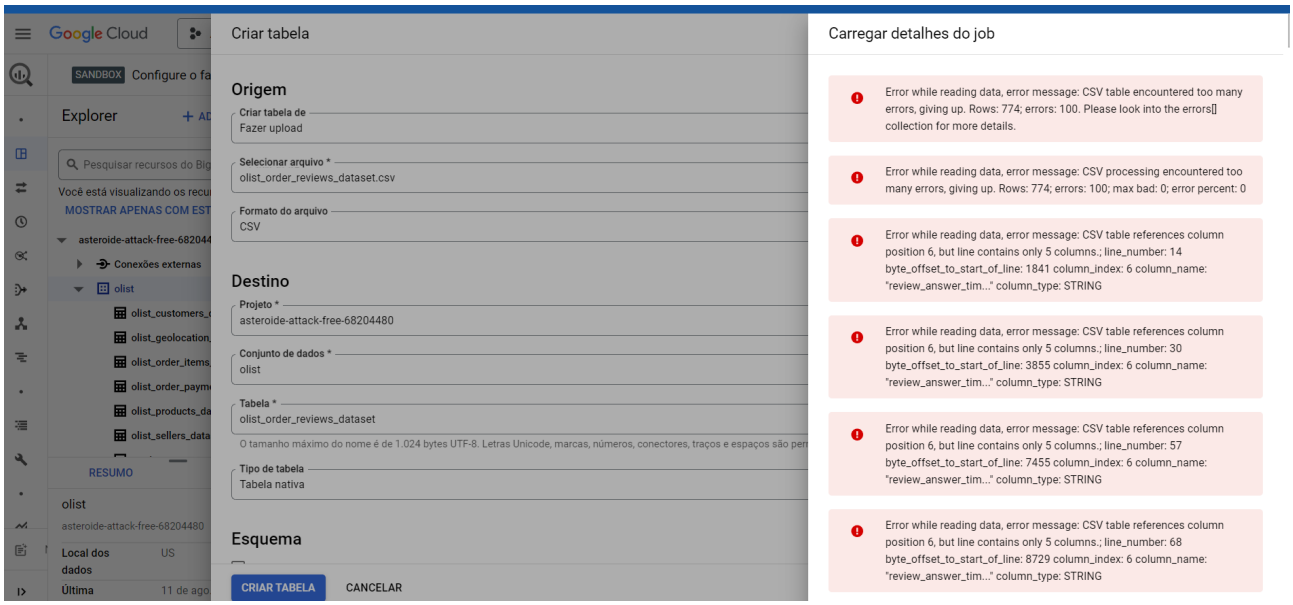
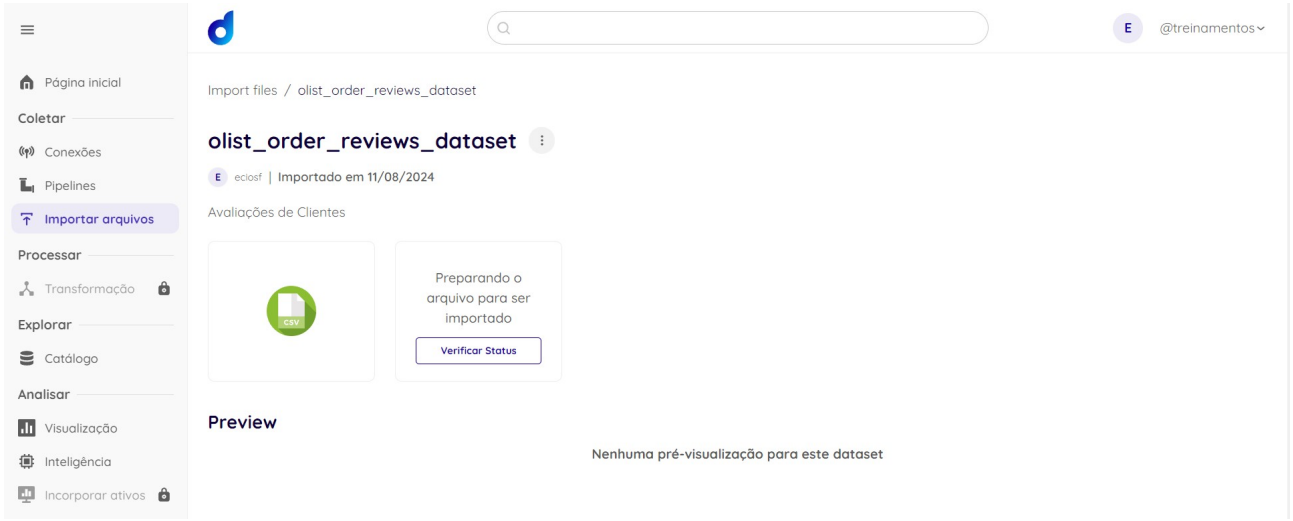
2.1 Integrar

1. Download da base: Baixar os arquivos CSV do Kaggle.
2. Criação de conexão: No módulo de Coleta da Dadosfera, criar uma nova conexão do tipo "Arquivo" e selecionar os arquivos CSV baixados.
3. Configuração da coleta: Definir o nome da tabela, o delimitador (vírgula) e outras opções relevantes.
4. Execução da coleta: Iniciar o processo de coleta para carregar os dados na Dadosfera.

Das 9 tabelas relacionadas a Olist apenas uma foi carregada como arquivo na plataforma da dadosfera por apresentar problemas de formatação com a plataforma da Google.

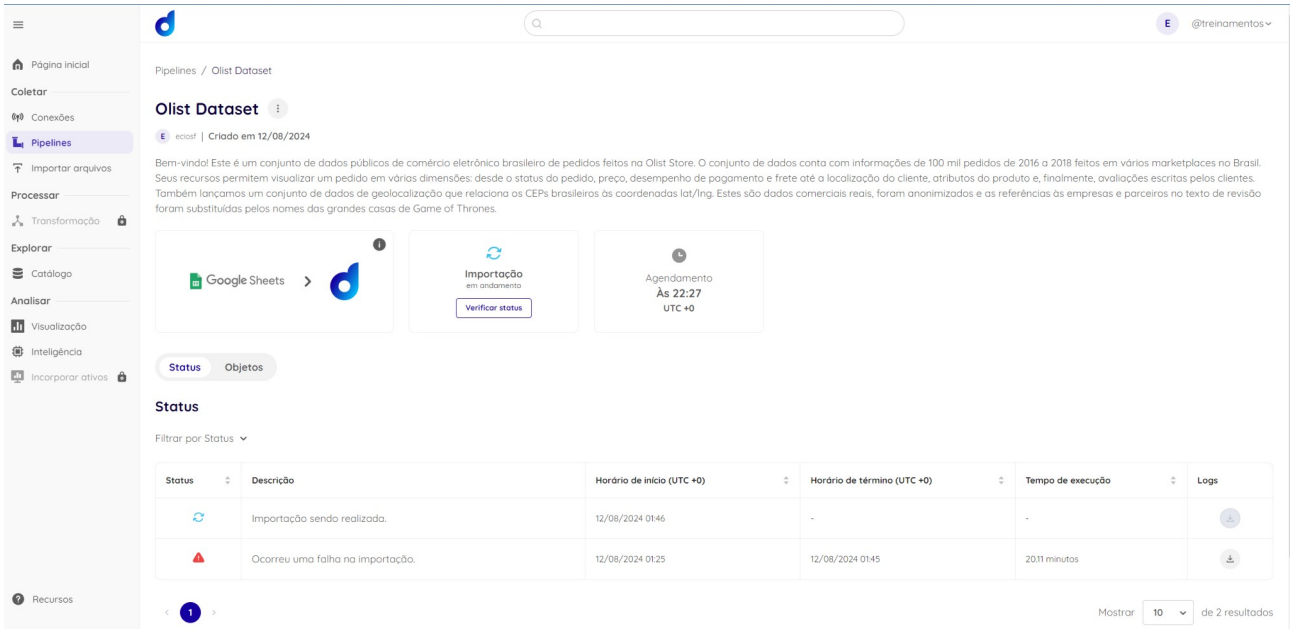


Erros ao tentar carregar aquivo *.csv no bigquery como tabela nativa. Consegui carregar o mesmo arquivo sem erros na plataforma da dadosfera. Esse dataset foi compartilhado com o grupo candidatos.



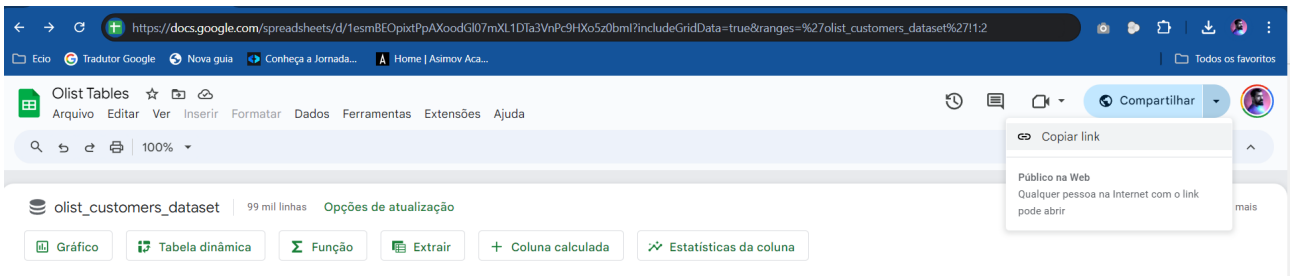
No total o dataset da Olist possui 9 tabelas

olist_customers_dataset.csv
olist_geolocation_dataset.csv
olist_order_items_dataset.csv
olist_order_payments_dataset.csv
olist_order_reviews_dataset.csv – Carregada como Arquivo
olist_orders_dataset.csv – Carregada como Arquivo
olist_products_dataset.csv
olist_sellers_dataset.csv
product_category_name_translation.csv

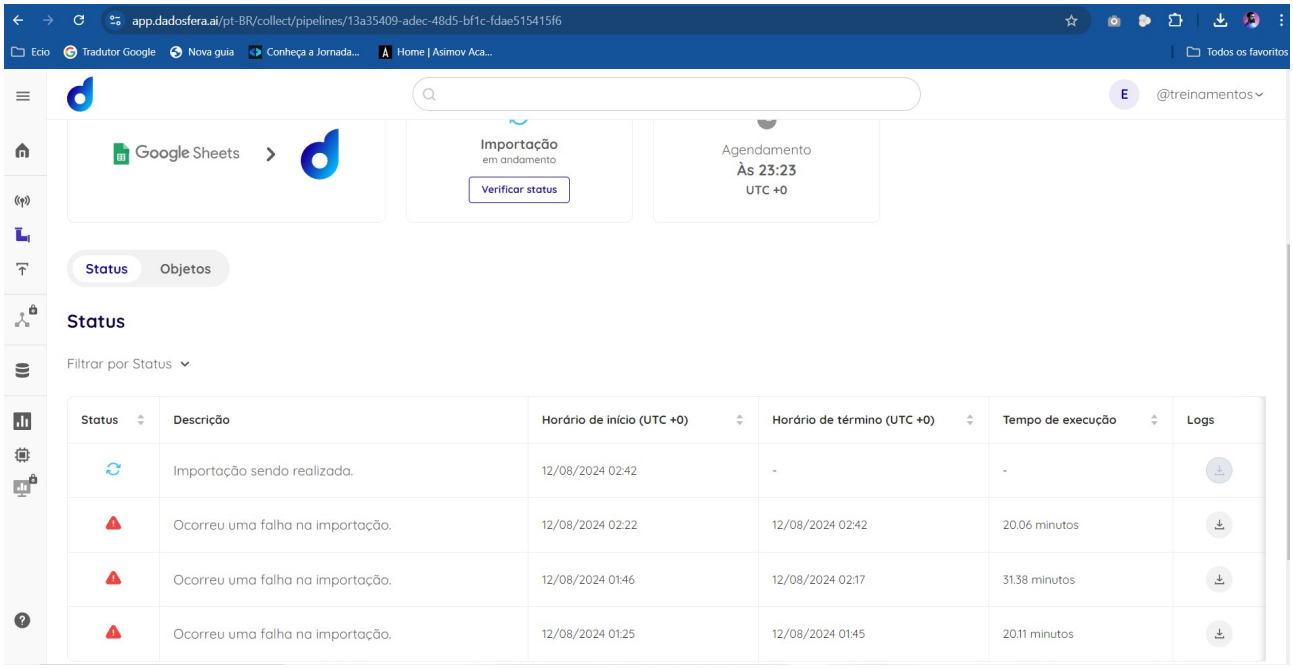


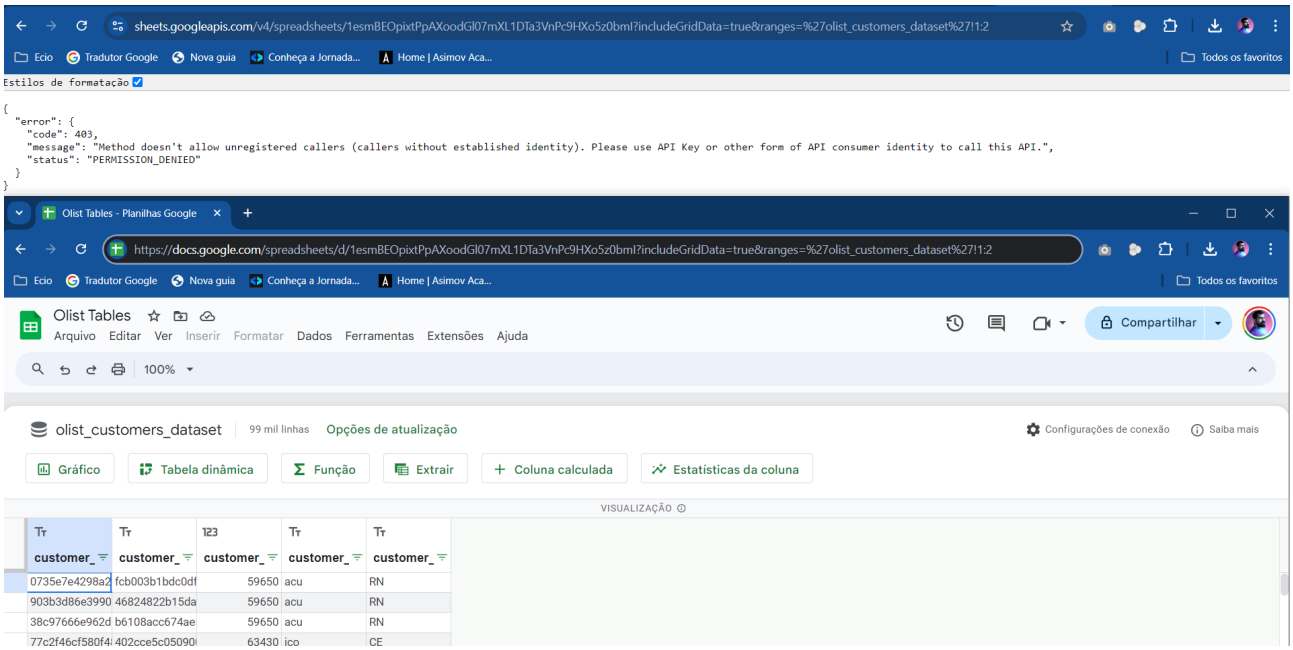
A tabela **olist_order_reviews_dataset.csv** foi carregada como arquivo. As demais tabelas foram criadas no Bigquery, mas como não é possível se conectar ao Bigquery usando a conta de treinamento, então conectei as tabelas do Bigquery ao Google Sheets e crie um pipeline para consumir os dados do Bigquery a partir do Google Sheets. Crie o pipeline com 7 tabelas e depois não consegui localizar na plataforma da Dadosfera como editar a fonte de dados do pipeline. Por esse motivo precisei carregar outra tabela como arquivo csv.

Ocorreu erro no pipeline ao tentar importar os dados da primeira aba da planilha com as tabelas da Olist. Ao analisar o log do erro verifiquei que o problema estava no fato de apesar de eu ter me autenticado no momento da criação do pipeline a planilha estava com acesso restrito. Por isso quando a API tentava acessar a primeira célula da planilha apresentou erro, mas quando eu tentava acessar via browser não havia erro. Ao que me parece a autenticação prévia não é suficiente para fazer a importação de dados do google sheets se a planilha não estiver com compartilhamento público.
De qualquer forma mesmo após o compartilhamento público da planilha no google sheet os dados ainda não conseguiram ser importados via pipeline. (Me informar se estou equivocado…)



Erro ao tentar importar





Log do erro:

2024-08-12 02:41:14.707 INFO __main__ Instantiating the JobMetadataService

2024-08-12 02:41:14.719 INFO __main__ Retrieving job_metadata from platform-api

2024-08-12 02:41:14.731 INFO brain.app.services.job_metadata_service Making

request to https://oz8v2zid1e.execute-api.us-east-1.amazonaws.com/jobs/singer/13a35409_adec_48d5_bf1c_fdae515415f6_6
2024-08-12 02:41:14.951 INFO brain.app.services.job_metadata_service Received status_code: 200 from API
2024-08-12 02:41:14.952 INFO __main__ Retrieving credentials from connection manager
2024-08-12 02:41:14.965 INFO brain.app.services.connection_manager_service Making request to https://wv6fw7ayj4.execute-api.us-east-1.amazonaws.com/connection_config/1723425392482_lfqrtl1l_google-sheets-1.0.0
2024-08-12 02:41:15.142 INFO brain.app.services.connection_manager_service Received status_code 200 from the API
2024-08-12 02:41:15.309 INFO __main__ Successfully retrieved the credentials
2024-08-12 02:41:15.311 INFO __main__ Retrieved job_metadata
2024-08-12 02:41:17.732 INFO brain.app.services.job_metadata_service Making request to https://oz8v2zid1e.execute-api.us-east-1.amazonaws.com/data_assets/singer/13a35409_adec_48d5_bf1c_fdae515415f6_6
2024-08-12 02:41:17.911 INFO brain.app.services.job_metadata_service Received status_code 200
2024-08-12 02:41:17.912 INFO brain.app.services.job_metadata_service state_file {'created_at': '2024-08-12T01:22:07.156567+00:00', 'data_asset_id': 11204, 'id': 6714, 'state_file': {}, 'updated_at': None}
2024-08-12 02:41:17.913 INFO __main__ Current state_metadata = {'created_at': '2024-08-12T01:22:07.156567+00:00', 'data_asset_id': 11204, 'id': 6714, 'state_file': {}, 'updated_at': None}
2024-08-12 02:41:17.914 INFO core.meltano.factory Creating a BaseMeltanoExtractor instance.
2024-08-12 02:41:17.915 INFO core.meltano.extractors.base Adding state to config
2024-08-12 02:41:20.957 INFO core.meltano.extractors.base Adding configuration parameters to meltano.
2024-08-12 02:41:48.313 INFO core.meltano.extractors.base Setting replication method FULL_TABLE for product_category_name_translation
2024-08-12 02:41:51.357 INFO core.meltano.extractors.base Selecting entities.
2024-08-12 02:41:51.358 INFO core.meltano.extractors.base Selecting

product_category_name_translation
2024-08-12 02:41:54.400 INFO core.meltano.extractors.base Validating configuration.
2024-08-12 02:41:54.401 INFO core.meltano.extractors.base Executing command: meltano config tap-google-sheets > config.json && .meltano/extractors/tap-google-sheets/venv/bin/tap-google-sheets --config config.json --discover
2024-08-12 02:41:57.421 INFO core.meltano.utils Authorized, token expires = 2024-08-12 03:41:55.704787

2024-08-12 02:41:57.422 INFO core.meltano.utils Starting discover

2024-08-12 02:41:57.422 INFO core.meltano.utils spreadsheet_metadata URL = https://sheets.googleapis.com/v4/spreadsheets/1esmBEOpixtPpAXoodGl07mXL1DTa3VnPc9HXo5z0bmI?includeGridData=false

2024-08-12 02:41:57.423 INFO core.meltano.utils METRIC: {"type": "timer", "metric": "http_request_duration", "value": 0.3085472583770752, "tags": {"endpoint": "spreadsheet_metadata", "http_status_code": 200, "status": "succeeded"}}

2024-08-12 02:41:57.423 INFO core.meltano.utils sheet_id = 2025639051, sheet_title = olist_customers_dataset

2024-08-12 02:41:57.423 INFO core.meltano.utils olist_customers_dataset URL = https://sheets.googleapis.com/v4/spreadsheets/1esmBEOpixtPpAXoodGl07mXL1DTa3VnPc9HXo5z0bmI?includeGridData=true&ranges='olist_customers_dataset'!1:2

2024-08-12 02:41:57.424 INFO core.meltano.utils METRIC: {"type": "timer", "metric": "http_request_duration", "value": 0.15347743034362793, "tags": {"endpoint": "olist_customers_dataset", "http_status_code": 400, "status": "succeeded"}}

2024-08-12 02:41:57.424 CRITICAL core.meltano.utils HTTP-error-code: 400 {'code': 400, 'message': "Unable to parse range: 'olist_customers_dataset'!1:2", 'status': 'INVALID_ARGUMENT'}: Unknown Error

```
Traceback (most recent call last):
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/client.py", line 111, in raise_for_error
    response.raise_for_status()
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/requests/models.py", line 940, in raise_for_status
    raise HTTPError(http_error_msg, response=self)
requests.exceptions.HTTPError: 400 Client Error: Bad Request for url: https://sheets.googleapis.com/v4/spreadsheets/1esmBEOpixtPpAXoodGl07mXL1DTa3VnPc9HXo5z0bmI?includeGridData=true&ranges='olist_customers_dataset'!1:2

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File ".meltano/extractors/tap-google-sheets/venv/bin/tap-google-sheets", line 8, in <module>
    sys.exit(main())
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/singer/utils.py", line 229, in wrapped
    return fnc(*args, **kwargs)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/__init__.py", line 51, in main
    do_discover(client, spreadsheet_id)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/__init__.py", line 26, in do_discover
    catalog = discover(client, spreadsheet_id)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/discover.py", line 6, in discover
    schemas, field_metadata = get_schemas(client, spreadsheet_id)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/schema.py", line 314, in get_schemas
    sheet_json_schema, columns = get_sheet_metadata(sheet, spreadsheet_id, client)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/schema.py", line 255, in get_sheet_metadata
    sheet_md_results = client.get(path=path, api=api, endpoint=sheet_title_escaped)
```

```
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/client.py", line 265, in get
    return self.request(method='GET', path=path, api=api, **kwargs)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/backoff/_sync.py", line 94, in retry
    ret = target(*args, **kwargs)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/backoff/_sync.py", line 94, in retry
    ret = target(*args, **kwargs)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/singer/utils.py", line 95, in wrapper
    return func(*args, **kwargs)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/client.py", line 259, in request
    raise_for_error(response)
  File "/app/.meltano/extractors/tap-google-sheets/venv/lib/python3.7/site-packages/tap_google_sheets/client.py", line 128, in raise_for_error
    raise ex(message)
tap_google_sheets.client.GoogleBadRequestError: HTTP-error-code: 400 {'code': 400, 'message': "Unable to parse range: 'olist_customers_dataset'!1:2", 'status': 'INVALID_ARGUMENT'}: Unknown Error

2024-08-12 02:41:57.425  ERROR       dadosfera_logs.dadosfera_logger  Uncaught exception: Command: meltano config tap-google-sheets > config.json && .meltano/extractors/tap-google-sheets/venv/bin/tap-google-sheets --config config.json --discover has returned a code different than 0
Traceback (most recent call last):
> File "/app/main.py", line 233, in <module>
    main(config=config, logger=logger, file_log_path=file_log_path)
  File "/app/main.py", line 133, in main
    df_singer = meltano_extractor.extract()
  File "/app/core/meltano/extractors/base.py", line 292, in extract
    self._validate_configuration()
  File "/app/core/meltano/extractors/base.py", line 190, in _validate_configuration
    execute_subprocess(command, timeout=180, file_log_path=self.file_log_path)
```

```
  File "/app/core/meltano/utils.py", line 90, in execute_subprocess
    raise _exception
Exception: Command: meltano config tap-google-sheets > config.json &&
.meltano/extractors/tap-google-sheets/venv/bin/tap-google-sheets --config config.json --
discover has returned a code different than 0
```