

Tech Stack

Function App Description

**Title:** Invoiced Data Export  
**Author:** Dominic Vicchello  
**Date:** 2025-05-05  
**Schedule:** Daily @ 5am

**Description:** Nightly data export process that triggers at 4am on daily basis.

Function Apps Directory

- export\_invoice\_json\_staging** - Timer function to export raw json data from invoice api endpoint and send to datalake in staging container
  - fetch\_json\_invoices.py - runs raw export for all invoices and sends to staging in json file
  - fetch\_json\_credits.py - runs raw export for all credits and sends to staging in json file
- process\_invoices** - Event Grid function which fires upon file creation to the staging invoice location in datalake
  - datalake\_read\_process\_json\_invoices.py - processes raw json file into header and line data files and sends to raw storage as parquet format
- process\_credits** - Event Grid function which fires upon file creation to the staging credits location in datalake
  - datalake\_read\_process\_json\_credits.py - processes raw json file into header and line data files and sends to raw storage as parquet format
- customers** - Timer function scheduled to export customer file from Invoiced API
  - export\_customers.py - processes raw json file and converts to parquet format for datalake storage
- csv\_validation** - Timer function scheduled to export customer file from Invoiced API
  - check\_invoices\_for\_csv.py - using the sql data created from process\_invoices, run query to get distinct list of invoices and create message of reason code based on rules
  - send\_email.py - if record is flagged as missing CSV, send email notification with PBI link of invoice details

Azure Function App

Utility Function

Script Function

Function App

Terminator

Version 1.0.1 - 5/25/2025

```
graph TD
    export_invoice_json_staging[export_invoice_json_staging] --> TimeSchedule4am[Time Schedule 4am]
    TimeSchedule4am --> fetch_json_invoices[fetch_json_invoices]
    TimeSchedule4am --> fetch_json_credits[fetch_json_credits]
    fetch_json_invoices --> def_invoice_api_invoices[def invoice api]
    fetch_json_credits --> def_invoice_api_credits[def invoice api]
    def_invoice_api_invoices --> define_page_loop_parameters_invoices[define page loop parameters]
    def_invoice_api_credits --> define_page_loop_parameters_credits[define page loop parameters]
    define_page_loop_parameters_invoices --> upload_invoice_blob_invoices[upload invoice blob]
    define_page_loop_parameters_credits --> upload_invoice_blob_credits[upload invoice blob]
    process_invoices[process_invoices] --> InvoiceEventGrid[Invoice Event Grid]
    InvoiceEventGrid --> datalake_read_process_json_invoices[datalake_read_process_json_invoices]
    datalake_read_process_json_invoices --> read_invoice_json[read_invoice_json]
    read_invoice_json --> define_staging_endpoint_invoices[define staging endpoint]
    define_staging_endpoint_invoices --> read_json_to_dataframe_invoices[read json to dataframe]
    read_json_to_dataframe_invoices --> process_header_invoices[process_header]
    read_json_to_dataframe_invoices --> process_line_invoices[process_line]
    process_header_invoices --> normalize_and_transform_header_dataframe_invoices[normalize and transform header dataframe]
    process_line_invoices --> normalize_and_transform_line_dataframe_invoices[normalize and transform line dataframe]
    normalize_and_transform_header_dataframe_invoices --> convert_to_parquet_invoices[convert to parquet]
    normalize_and_transform_line_dataframe_invoices --> convert_to_parquet_invoices
    convert_to_parquet_invoices --> upload_to_raw_header_container_invoices[upload to raw header container]
    convert_to_parquet_invoices --> upload_to_raw_line_container_invoices[upload to raw line container]
    process_credits[process_credits] --> CreditEventGrid[Credit Event Grid]
    CreditEventGrid --> datalake_read_process_json_credits[datalake_read_process_json_credits]
    datalake_read_process_json_credits --> read_credits_json[read_credits_json]
    read_credits_json --> define_staging_endpoint_credits[define staging endpoint]
    define_staging_endpoint_credits --> read_json_to_dataframe_credits[read json to dataframe]
    read_json_to_dataframe_credits --> process_header_credits[process_header]
    read_json_to_dataframe_credits --> process_line_credits[process_line]
    process_header_credits --> normalize_and_transform_header_dataframe_credits[normalize and transform header dataframe]
    process_line_credits --> normalize_and_transform_line_dataframe_credits[normalize and transform line dataframe]
    normalize_and_transform_header_dataframe_credits --> convert_to_parquet_credits[convert to parquet]
    normalize_and_transform_line_dataframe_credits --> convert_to_parquet_credits
    convert_to_parquet_credits --> upload_to_raw_header_container_credits[upload to raw header container]
    convert_to_parquet_credits --> upload_to_raw_line_container_credits[upload to raw line container]
    export_customers[export_customers] --> TimeSchedule4am_customers[Time Schedule 4am]
    TimeSchedule4am_customers --> process_customers[process_customers]
    process_customers --> def_customer_api[def customer api]
    def_customer_api --> define_column_headers[define column headers]
    define_column_headers --> define_page_loop_parameters_customers[define page loop parameters]
    define_page_loop_parameters_customers --> normalize_json_customers[normalize json]
    normalize_json_customers --> rename_columns_customers[rename columns]
    rename_columns_customers --> upload_customers_blob[upload customers blob]
    csv_validation[csv_validation] --> TimeSchedule6am[Time Schedule 6am]
    TimeSchedule6am --> check_invoices_for_csv[check_invoices_for_csv]
    check_invoices_for_csv --> check_invoices_for_csv_db[check_invoices_for_csv]
    check_invoices_for_csv_db --> define_sql_query[define SQL query]
    define_sql_query --> cursor_fetchall[cursor fetchall()]
    cursor_fetchall --> sql_connection[sql connection]
    sql_connection --> for_invoices_loop[for invoices loop]
    for_invoices_loop --> define_file_attachment_url[define file attachment url]
    define_file_attachment_url --> read_api_response[read api response]
    read_api_response --> filter_only_invoices_missing_csv[filter only invoices missing CSV]
    filter_only_invoices_missing_csv --> create_dataframe_with_results[create dataframe with results]
    create_dataframe_with_results --> add_lastmodified_date[add lastmodified date]
    add_lastmodified_date --> append_data_list[append data[] list]
    append_data_list --> define_blob_storage[define blob storage]
    define_blob_storage --> convert_to_parquet_csv[convert to parquet]
    convert_to_parquet_csv --> send_results_to_datalake[send results to datalake]
    add_lastmodified_date --> send_single_result_in_email[send single result in email]
    send_single_result_in_email --> send_email[send_email]
    send_email --> define_PBI_link[define PBI link]
    define_PBI_link --> define_recipients_list[define recipients list]
    define_recipients_list --> define_email_parameters[define email parameters]
    define_email_parameters --> define_email_body[define email body]
    define_email_body --> add_logo_to_email_body[add logo to email body]
    add_logo_to_email_body --> define_boto3_client[define boto3 client]
    define_boto3_client --> SendEmail[Send Email]
```