Reports Creation

This repository contains a suite of report creation tools intended as a replacement for the previous process that utilized COGNOS. These tools were created using python and several additional libraries such as pandas and reportlab. Reports are generated by Federal Electoral District (FED) and describe various aspects of the given FED. These tools will *NOT* work as intended on networks that do not have access to the Elections Canada Corporate Database.

This document has several sections that include an installation and setup guide, an introduction to the workflow files and their formatting, as well as a description of the tools themselves including their outputs. For information on the individual files that make up the tools please see the readme files contained in the components directory and its subdirectories.

Report Types and Descriptions

Using the tools in this repository the following reports can be created:

Report Name	Abbreviation	Description
Polling District Profile	PDP	Lists all PDs in a given FED. Shows PD number, PD name, electors listed and void status. Vois status indicates whether a given PD is active or not. Totals are included at end of report.
Advance Polling Districts	APD	Lists all APDs in a given FED. Shows APD #, APD name, PDs served, total # of PDs in each APD (includes MOBs and SBPDs). Total number of APDs included at end of report
Mobile Polls Summary	MPS	Lists all MOBs in a given FED. Shows PD #s, # of institutions, electors listed, and APD # for each MOB.
Descriptions	PDD	Lists the PD street segments for each Ordinary Poll (ORD PD), lists each Single Building Poll (SBPD) and Mobile Poll (MOB) in a given FED. Shows PD #, PD name, and CSD name for every PD. ORD PD: Street names, FROM-TO features, FROM-TO civic # ranges, and sides. TRMs are added at the end of affected ORD PDs in the Prairies only. SBPDs: Building name and civic address associated with it. MOBs: Institution names, institution addresses, and electors listed. There is a sub-total at the end of each MOB.
Electoral District Poll Key	DPK	Lists every PD street segment for each ORD, SBPD and MOB in a given FED. Each PDSS shows the CSD, street name, the FROM-TO features, the FROM-TO civic # range, side, PD #, and APD #. The PDSS are grouped and ordered by street name, type, direction, and address range, and are sub-grouped by CSD name and type.

Report Name	Abbreviation	Description
Communities with Indigenous Peoples	IDR	Lists Communities containing Indigenous Peoples in the FED

Please note that going forward the abbreviations in the above table are what will be used to refer to each report. This will be maintained throughout the rest of this documentation and within the tools themselves. For example, all files related to the Polling District Profile will have PDP in their file name.

Environment Setup

Requirements

This project will require the installation python 3.9 or newer from the software center. Package requirements can be found below and can be installed using pip.

The required additional python packages for this tool are as follows:

- pandas
- reportlab
- oracledb
- click
- openpyxl

Python Installation / Setup

Before the additional packages can be installed, python needs to be checked to ensure it is properly setup. To verify this please follow the steps below:

1.) Open the command line (cmd) ⊞ Win then type cmd to bring up. the cmd window. If successful your cmd should look something similar to the below image:
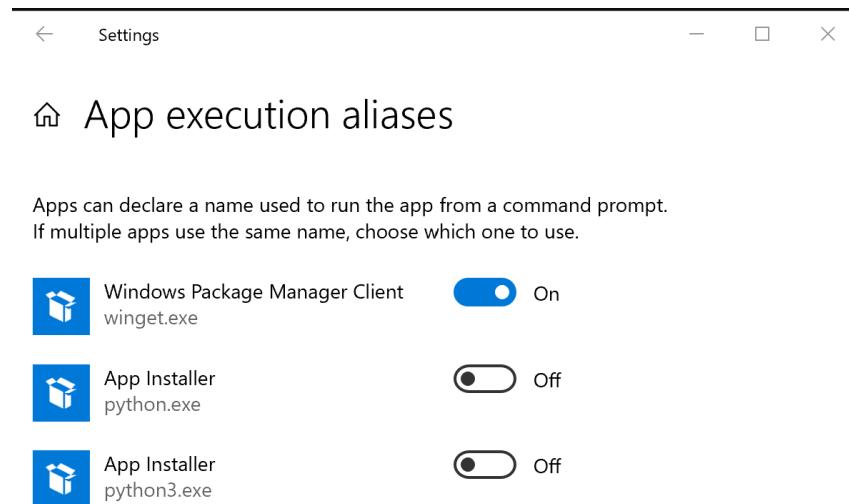
```
U:\>python
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

2.) To ensure that python is set up correctly type python on the command line and hit enter. If successful your cmd should look something similar to the below image:

```
U:\>python
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

3.) If one of the following errors occurs please attempt to resolve them using the instructions below: - If the command is not recognized add the folder containing your python.exe file to the
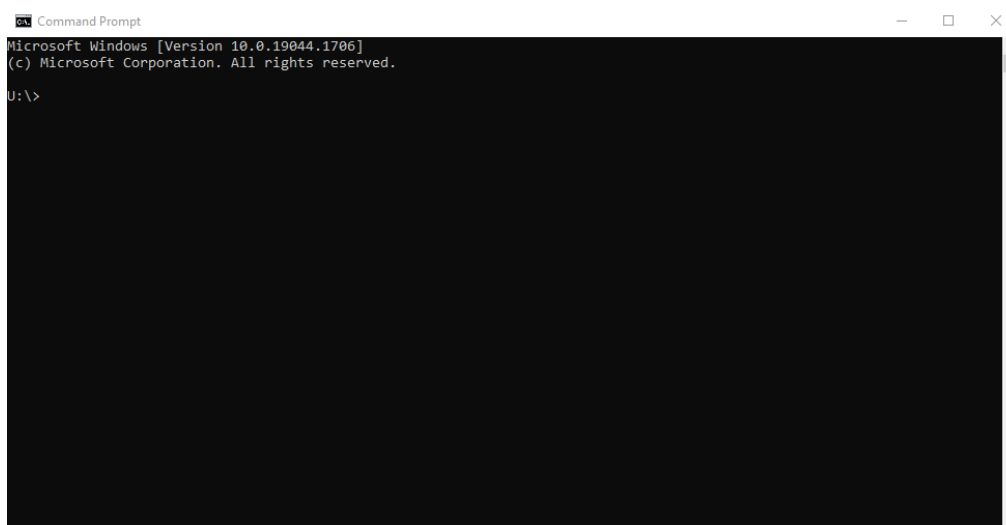
PATH environment variable for your account. - If the python command opens the Windows store type the following into the search bar: "Manage app execution aliases" and turn off the two python app installers that are listed.

4.) Once the error has been corrected run the python command again to ensure that it is now working as expected.

Once python has been confirmed to be working the required packages can be installed using pip. Pip is a tool that can be accessed from the command line to install and manage python packages. It comes preinstalled with python so no additional actions are required to set it up.

1.) Open the command line (cmd) ⊞ Win then type cmd to bring up the cmd window

2.) Install required packages using the provided requirements.txt using pip:

pip install --trusted-host pypi.org --trusted-host pypi.python.org --trusted-host files.pythonhosted.org -r requirements.txt

Should that command fail, each package can also be installed individually using the following command:

pip install --trusted-host pypi.org --trusted-host pypi.python.org --trusted-host files.pythonhosted.org <library name>

In this case library name stands for the name of the library you want to install. For example, if we wanted to install a library called pandas the command would look something like this:

pip install --trusted-host pypi.org --trusted-host pypi.python.org --trusted-host files.pythonhosted.org pandas

Should an error appear stating that pip needs to be updated navigate to the folder containing your python.exe. For example my python.exe was found at this location:

C:\Program Files\Python311

Once you've found the correct path run the following command to update pip:

python.exe pip install --trusted-host pypi.org --trusted-host pypi.python.org --trusted-host files.pythonhosted.org --upgrade pip

Other Considerations

If downloading the data for this project (such as running the data download tool found in this repo, or running the SQL) access to the corporate database as well as additional schema(s) is required. Please ensure that you have permission to access the corporate database (CDB). At minimum read access to the SITES_ADMIN schema should be included as well.

Using the Tools

The tools in this repository are designed to be run from a command line interface (CLI) and running the tool outside of this type of interface is not recommended or supported. Please note that when running the tools it will overwrite any preexisting files in the output directories. If you need to retain any files for archival purposes please make a copy in a separate directory. If exporting files to a directory on a shared drive ensure that none of the files you are replacing are currently open by you or by other users as this will cause the tool to skip the open document(s). Any files skipped because of this reason will be noted as a message on the command line as well as recorded in the log file produced by the tool. Once the problem files are closed the tool can be rerun to replace just those records by making alterations to the workflow file.

At this time there are three tools in available for use through the CLI:

- *Data Download:* Downloads the data for the reports from the CDB.
- *Report Creation:* Creates the requested reports using the data from the data download tool.
- *Output Zipper:* Creates a FED level .zip file containing all reports and maps associated with that FED.

In order to run the tools in this repository you will need to create a workflow file. These workflow files are structured using Javascript Object Notation (JSON). This file will contain key parameters the tool needs in order to create the reports. Workflow files follow a consistent schema which is outlined and described in the workflow section of each tool. Example files for each tool(except for the data download tool for privacy reasons) can also be found in the workflows directory at the root of this repository. It is recommended that users alter these files as needed using an integrated development environment (IDE) such as VS Code or a text editor like Notepad++. A correctly formatted workflow in an IDE should look like the image below:

```json
{} zip_ex.json  ×
workflows > {} zip_ex.json > ...
  1  {
  2      "map_dir": "J:\\MapSeries\\Dump",
  3      "report_dir": "J:\\EMRP\\Work\\GAM_Reports",
  4      "out_dir": "J:\\GAM_Products",
  5      "feds": [10001, 10002, 10003, 10004, 10005, 10006, 10007, 11001, 11002, 11003, 11004, 12001, 12002, 12003, 12004, 12005, 12006, 12007, 12008, 12009, 12010, 12011,
  6  }
  7  |
```

The benefits of using an IDE over a text editor for this purpose this that the IDE comes with built in tools that will highlight syntax errors and autocomplete partial lines of code. It is recommended that a user take advantage of these functions. An IDE that is available in the Elections Canada production environment from the software centre is Visual Studio Code. This IDE will allow you to easily manage your workflows and code while minimizing errors.

Data Download

This tool is responsible for downloading the data from the databases and into local csv files for the reports creation tool. This tool must be run on a production machine, so it can access the database. Ensure that you have been granted all necessary permissions to databases and schemas before running the tool. Make sure that you are downloading the data for all FED's you require as the tool will only download data for the FED's specified in the workflow file.

There are a number of SQL queries available for the tool to run. They can be found in the folder called sql at the root of this repository. Each SQL query downloads the data needed for certain reports. To check what data is required for each report refer to the table in the report creation tool instructions. The table below specifies the csv that is associated with the data from each sql

SQL Query Name	CSV File Name
pd_nums.sql	pd_nums.csv

SQL Query Name	CSV File Name
pd_desc.sql	pd_desc.csv
strm.sql	strm.csv
ps_add.sql	ps_add.csv

The csv files produced above contain all the information necessary to create every report except for the Communities with Indigenous Peoples report which requires the PDs and Indigenous Communities.xlsx file which is not one of the files produced by this tool and must be retrieved from a location TBD.

Workflow File Creation

For the data download tool the format for the workflow JSON can be seen below:

```
{ "data": [
    {
      "username": your oracle db username,
      "password": your oracle db password,
      "database": connection string for the CDB,
      "sql_path": Path to the required sql file,
      "ed_list": an array of fed numbers to download the data for stored as integers
    }]
}
```

Using the above guide a complete workflow for this tool for three feds would look as follows:

```
{"data": [
    {
      "username": "usernme",
      "password": "pwd",
      "database": "db.connect.string",
      "sql_path": "C:\\reports_creation\\sql\\pd_desc.sql",
      "ed_list": [47001, 48001, 24001]
    },{
      "username": "usernme",
      "password": "pwd",
      "database": "db.connect.string",
      "sql_path": "C:\\reports_creation\\sql\\pd_nums.sql",
      "ed_list": [47001, 48001, 24001]
    },{
      "username": "usernme",
      "password": "pwd",
      "database": "db.connect.string",
      "sql_path": "C:\\reports_creation-Build-PDF-Report\\sql\\ps_add.sql",
      "ed_list": [47001, 48001, 24001]
    },{
      "username": "usernme",
```

```
        "password": "pwd",
        "database": "db.connect.string",
        "sql_path": "C:\\reports_creation\\sql\\strm.sql",
        "ed_list": [47001, 48001, 24001]
    }
]}
```

An important note for file paths: please ensure that the double slash syntax is maintained as single slashes will result in an error and cause the tool to fail
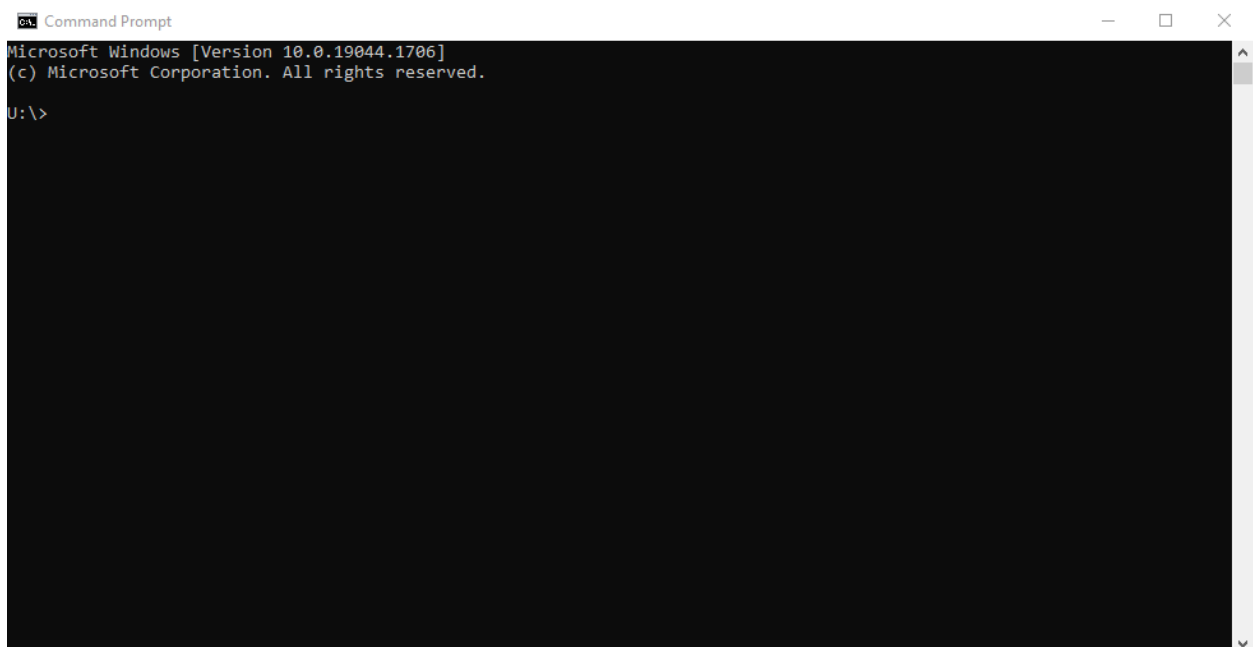
\\ instead of \

The above JSON would download all data for the three listed FEDs. A copy of this JSON can be found in the workflows for the folder but not in a working form as that would contain sensitive information. Care should be taken to protect this file once created, and it should not be shared or placed on a shared drive.

Running the tool

To run the tool please follow these steps:

1.) Create your workflow as described in the workflows section of this documentation for this tool. Ensure there are no syntax errors before running and that you know the path to the file.

2.) Open the cmd window (⊞ Win then type cmd)

3.) Navigate to the root directory of this repository. The recommended path for this is **c:\reports_creation** to keep the path as short as possible. To navigate to the directory use the following commands
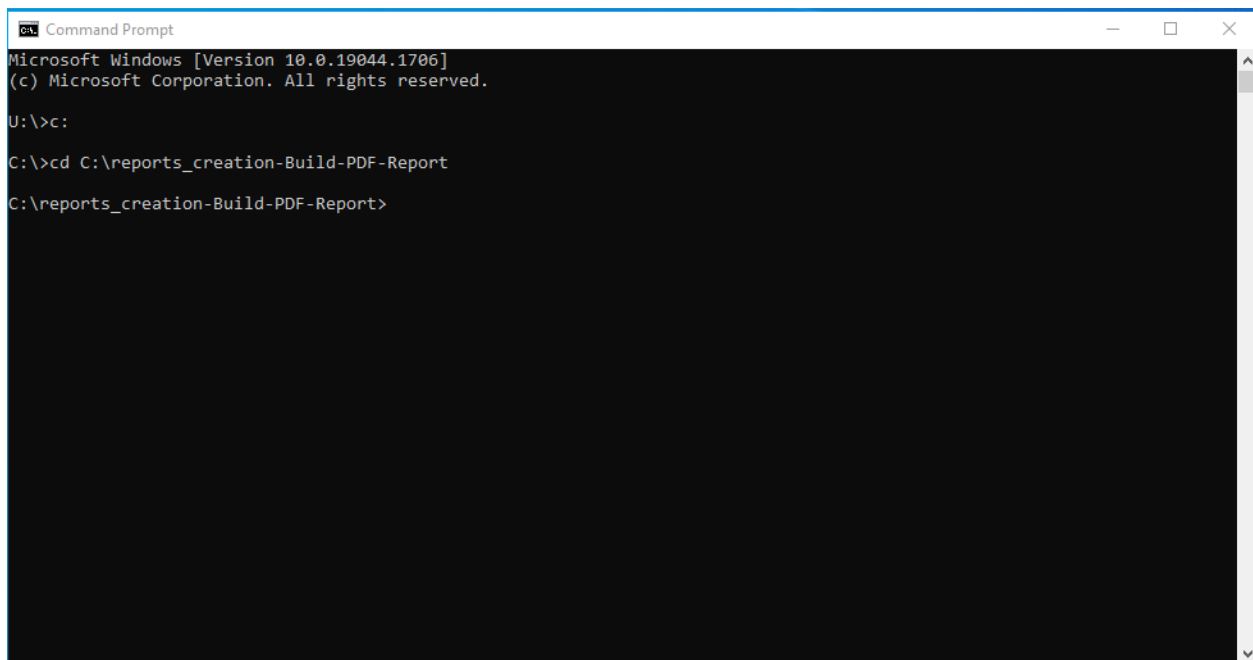
Use the below command if the selected drive does not match the drive containing the files from this repository.

C:

Change to whatever the drive you stored the files. To navigate to the correct directory use the cd command. For example:

cd C:\reports_creation

the above example would change the active directory to the reports_creation directory. With the correct directory assigned we can now run the tools. The below image shows the complete process that is broken down above.
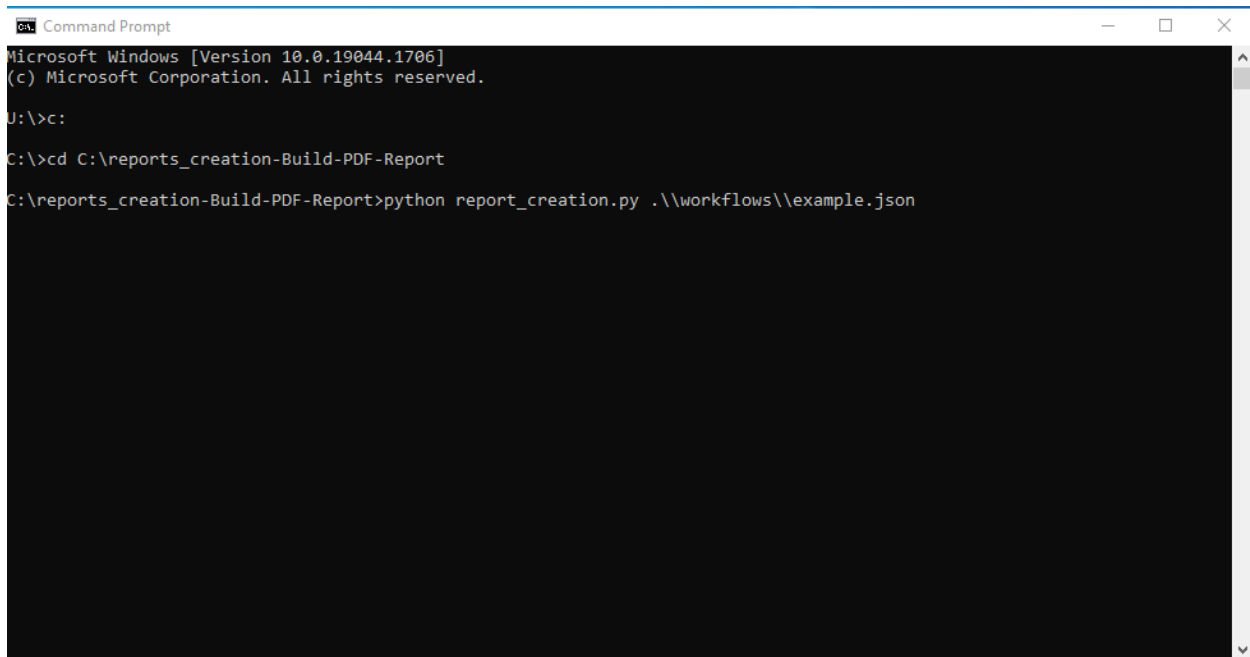

4.) Activate the tool by creating a command using the following formula:

python <tool name .py> <path to workflow>

An example of a valid command for the data download tool using a workflow file called download_workflow.json which is located in the workflows folder of this repository would look as follows:

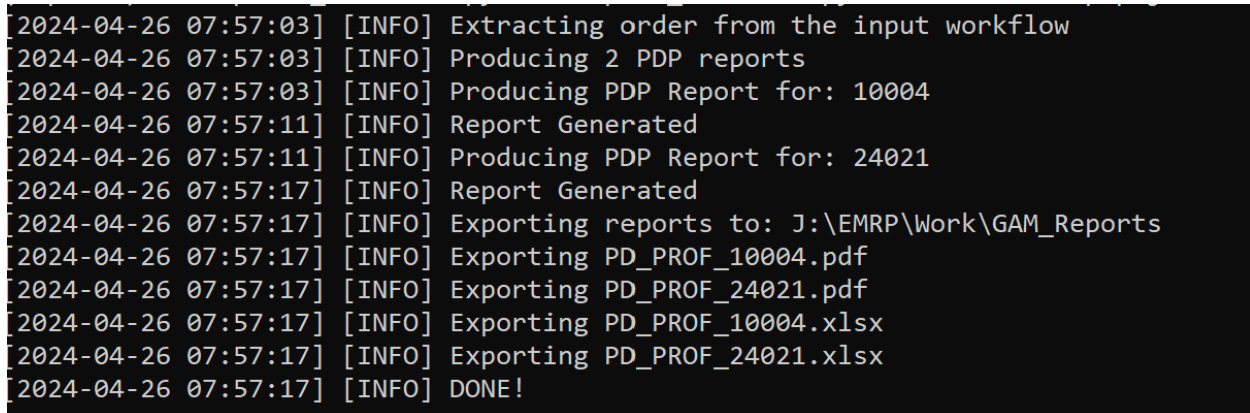python data_download.py .\\workflows\\download_workflow.json

5.) Once the command is constructed hit ⏎ Enter to run it.

6.) If the command was valid the tool will begin to process the workflow file. While running the tool it will produce a series of messages complete with time stamps to give the user updates on the tools progress and any significant events that might have occurred.

```
[2024-04-26 07:57:03] [INFO] Extracting order from the input workflow
[2024-04-26 07:57:03] [INFO] Producing 2 PDP reports
[2024-04-26 07:57:03] [INFO] Producing PDP Report for: 10004
[2024-04-26 07:57:11] [INFO] Report Generated
[2024-04-26 07:57:11] [INFO] Producing PDP Report for: 24021
[2024-04-26 07:57:17] [INFO] Report Generated
[2024-04-26 07:57:17] [INFO] Exporting reports to: J:\EMRP\Work\GAM_Reports
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.xlsx
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.xlsx
[2024-04-26 07:57:17] [INFO] DONE!
```

There are three types of messages that can appear in the console:

- INFO: Informational messages on the current action the tool is processing.
- WARNING: Something occurred that was outside the normal parameters of the tool but did not inhibit processing. An example of the common warning for this tool is No data available for the specified FED.
- ERROR: Something occurred that was significant enough to inhibit processing.

Outputs

This tool creates the following outputs:

1.) CSV file(s) containing data for the feds specified in the workflow. These are located in the data folder at the root of this directory (this location cannot be changed)

2.) A log file containing all the messages that were printed to the cmd window. This serves as a record of the process and allows the user to check for errors after processing. This file will be located in the logs folder at the root of this repository and all log files will use this naming convention: .log with the date following the YYYY-MM-DD convention.

Report Creation

This tool is responsible for creating the reports and exporting them to a given directory. This tool should always be run after the data download tool as without data no reports will be generated. Please note that there are no template files used by this tool to generate the reports. Instead, each report is built from the ground up using the specifications found in the builder class for the given report type. If a user wants to alter the reports from its default settings then they will need to edit the associated generator and builder scripts for that report. Further information on these scripts can be found in the readme file in the *components* directory of this repository.

In order to produce each report certain datasets are required to be present in the *data* directory. The table below shows the report and the required datasets (as a csv). Ensure that the data is saved locally in a subdirectory of the root of this repository for the tools to run correctly. Ensure that all data for the desired FED's is present as the data can be downloaded only for specific FEDs when using the data download tool.

Report	Abbreviation	Required Datasets
Polling District Profile	PDP	pd_nums.csv
Advance Polling Districts	APD	pd_nums.csv
Mobile Polls Summary	MPS	pd_nums.csv
Descriptions	PDD	pd_desc.csv, strm.csv, ps_add.csv
Electoral District Poll Key	DPK	pd_desc.csv
Communities with Indigenous Peoples	IDR	PDs and Indigenous Communities.xlsx

Workflow File Creation

The JSON for the reports creation tool should be formatted as follows:

```
{
  "reports": [{
    "type": report abbreviation,
    "feds": an array of all fed numbers
    "data": the path to the local csv or xlsx containing the data
  },{
    "type": report abbreviation,
```

"feds": an array of all fed numbers to create reports for
"data": the path to the local csv or xlsx containing the data
}],
   "export_directory": path to export directory
}

Using the above guide an example of a valid workflow creating all reports for three feds would look as follows:

```
{"reports":[{
    "type": "PDP",
    "feds": [47001, 48001, 24001],
    "data":".\\data\\pd_nums.csv"
    },{
    "type": "APD",
    "feds": [47001, 48001, 24001],
    "data": ".\\data\\pd_nums.csv"
    },{
    "type": "PDD",
    "feds": [47001, 48001, 24001],
    "data": ".\\data\\pd_desc.csv"
    }, {
    "type": "DPK",
    "feds": [47001, 48001, 24001],
    "data": ".\\data\\pd_desc.csv"
    }, {
    "type": "MPS",
    "feds": [47001, 48001, 24001],
    "data": ".\\data\\pd_nums.csv"
    }, {
    "type": "IDR",
    "feds": [47001, 48001, 24001],
    "data": ".\\data\\PDs and Indigenous Communities.xlsx"
 }],
   "export_directory": "J:\\EMRP\\Work\\GAM_Reports"
}
```

An important note for file paths: please ensure that the double slash syntax is maintained as single slashes will result in an error and cause the tool to fail
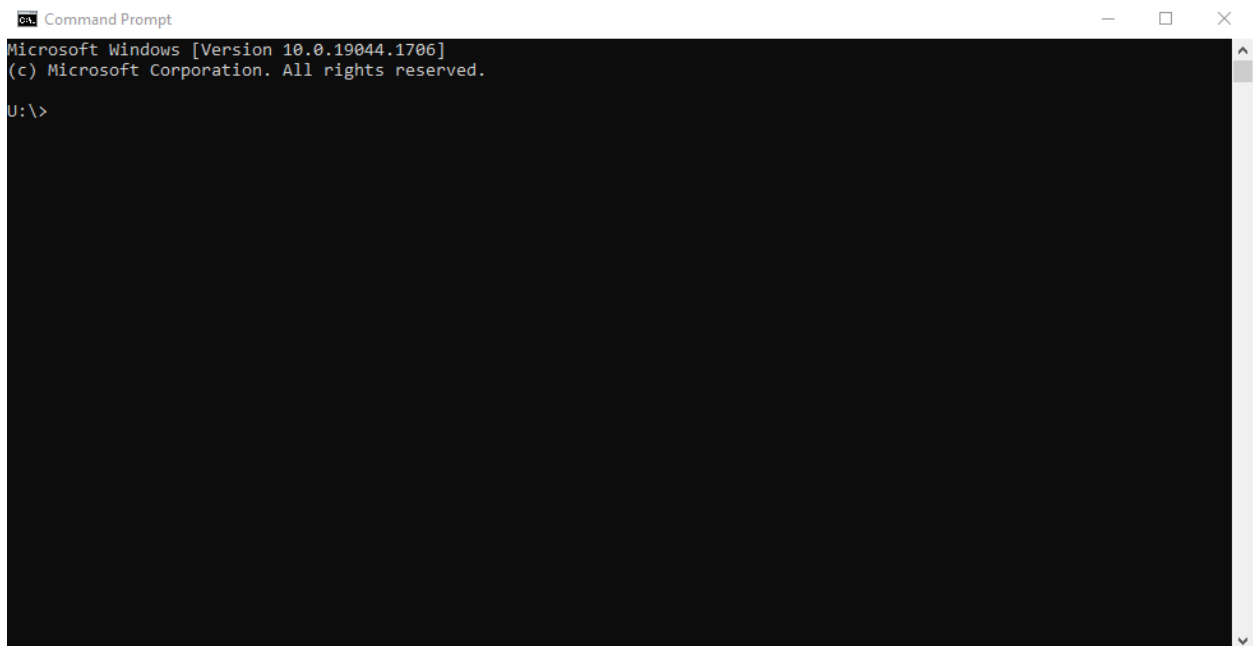
\\ instead of \

The above file can be found in workflows folder at the root of this repository and is meant to serve as a reference when creating other workflows. The above JSON would create reports of every type for each of the three FED's listed in the array (47001, 48001, and 24001). If wanting to run the tool on all FED's then an array of all FED's would need to be added to the workflow. Only the report types being generated need an array unneeded report types can be removed from the workflow.

Running the Tool

To run the reports creation tool follow the following steps:

1.) Create your workflow as described in the workflows section of this documentation for this tool. Ensure there are no syntax errors before running and that you know the path to the file.

2.) Open the cmd window (⊞ Win then type cmd)


```
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

U:\>
```

3.) Navigate to the root directory of this repository. The recommended path for this is **c:\reports_creation** to keep the path as short as possible. To navigate to the directory use the following commands
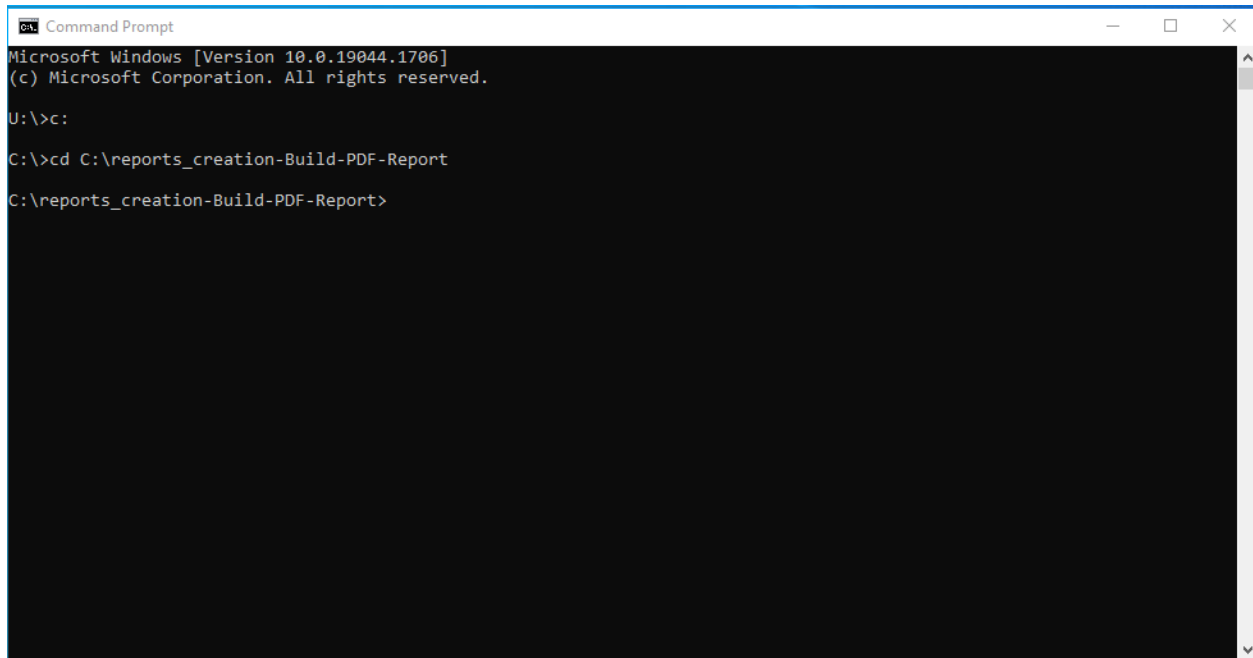
Use the below command if the selected drive does not match the drive containing the files from thi repository

C:

Change to whatever the drive you stored the files. To navigate to the correct directory use the cd command. For example:

cd C:\reports_creation

the above example would change the active directory to the reports_creation directory. With the correct directory assigned we can now run the tools. The below image shows the complete process that is broken down above.
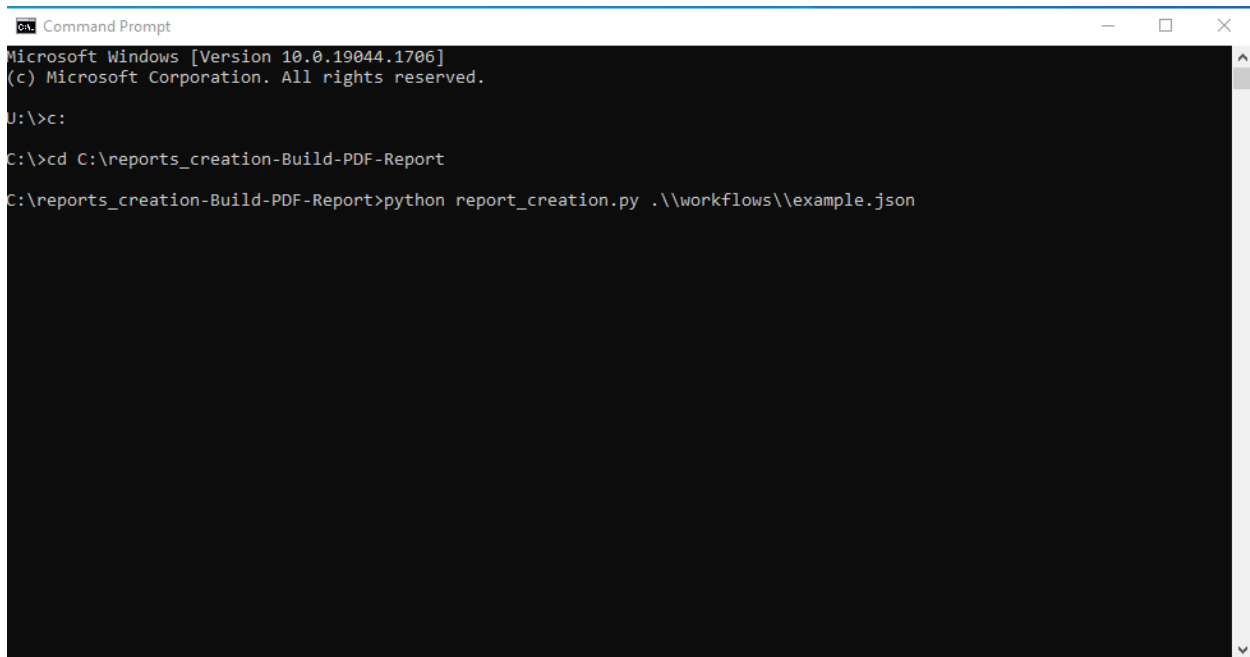
4.) Activate the tool by creating a command using the following formula:

python <tool name .py> <path to workflow>

An example of a valid command for the data download tool using a workflow file called download_workflow.json which is located in the workflows folder of this repository would look as follows:

python data_download.py .\\workflows\\pdd.json

5.) Once the command is constructed hit ⏎ Enter to run it.

6.) If the command was valid the tool will begin to process the workflow file. While running the tool it will produce a series of messages complete with time stamps to give the user updates on the tools progress and any significant events that might have occurred.

```
[2024-04-26 07:57:03] [INFO] Extracting order from the input workflow
[2024-04-26 07:57:03] [INFO] Producing 2 PDP reports
[2024-04-26 07:57:03] [INFO] Producing PDP Report for: 10004
[2024-04-26 07:57:11] [INFO] Report Generated
[2024-04-26 07:57:11] [INFO] Producing PDP Report for: 24021
[2024-04-26 07:57:17] [INFO] Report Generated
[2024-04-26 07:57:17] [INFO] Exporting reports to: J:\EMRP\Work\GAM_Reports
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.xlsx
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.xlsx
[2024-04-26 07:57:17] [INFO] DONE!
```

There are three types of messages that can appear in the console:

- INFO: Informational messages on the current action the tool is processing.
- WARNING: Something occurred that was outside the normal parameters of the tool but did not inhibit processing. An example of the common warning for this tool is No data available for the specified FED.
- ERROR: Something occurred that was significant enough to inhibit processing.

The pdf files produced by this tool will be output in a folder called 'scratch' in the root folder of this repository. From there they are exported to the directory specified by the export_directory parameter in the workflow file. Once production of all reports is complete the script will export

all pdf files in the scratch directory to the directory specified in the 'export_directory' parameter in the workflow file.

Note that the scratch directory gets deleted everytime the script is run. The tool will overwrite existing versions of a report if a new one is generated.

1.) PDF and XLSX files sorted by fed in the given export directory. Standardized base file names for each report can be found in the table below. Cells have been left blank where a file name is not applicable as no file is produced:

Report	English PDF Base Name	French PDF Base Name	English XLSX Base Name	French XLSX Base Name
PDP	PD_PROF	PD_PROF	PD_PROF	PD_PROF
APD	ADVANC	ADVANC	ADVANC	ADVANC
PDD	DESCRIPTIONS	DESCRIPTIONS		
DPK	INDCIR	INDCIR	INDCIR	INDCIR
MPS	SUMINS	SUMINS	SUMINS	SUMINS
IDR	INDIG_AUTOCH	AUTOCH_INDIG		

Note that the above table only contains the base file names of the file. All file names will have the FED number appended after the base in order to signify the FED to which the report belongs.

2.) A log file containing all the messages that were printed to the cmd window. This serves as a record of the process and allows the user to check for errors after processing. This file will be located in the logs folder at the root of this repository and all log files will use this naming convention: .log with the date following the YYYY-MM-DD convention.

Zip Outputs

This tool is responsible for creating an organized zip file of all the products produced by the map series and report creation projects. The output of this tool is a .zip file per FED that contains both maps and reports for the consumption of both internal and external clients. This tool should only be run after all other tools are finished processing as it requires all products to be available at runtime.

This tool will place the produced .zip files in a directory as specified in the workflow .zip. It is imperative that the user ensure that none of these files are open BEFORE running the tool as a permission error will cause the tool to fail and require processing to start over again.

Workflow File Creation

The JSON for the reports creation tool should be formatted as follows:

```
{
  "map_dir": path to the directory containg unsorted map series PDFs,
  "report_dir": path to the dorectory containing sorted report PDFs from the report creation tool,
  "out_dir": path to the directory that will contain the sorted and zipped outputs,
  "feds": an array of fed numbers as integers to sort and create .zip files for
}
```

An example of what a complete workflow would look like is found below:

```
{
  "map_dir": "J:\\MapSeries\\Dump",
  "report_dir": "J:\\EMRP\\Work\\GAM_Reports",
  "out_dir": ".\\test",
  "feds": [10001, 12002, 24002]
}
```

An important note for file paths: please ensure that the double slash syntax is maintained as single slashes will result in an error and cause the tool to fail
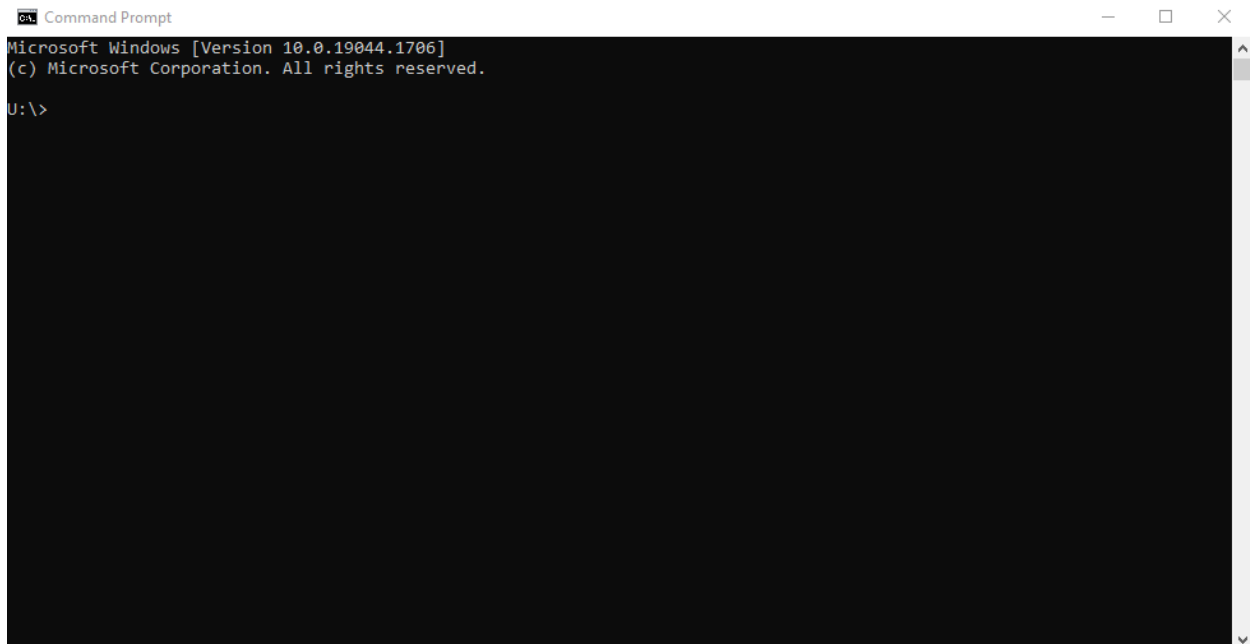
\\ instead of \

The above workflow would sort and zip all the files contained in the both the map and report directories for each of the given FEDs and place zipped versions of those directories in the specified output directory.

Running the Tool

The process for running this tool is similar to the other tools previously described in this section. Please follow the directions below to best use this tool:

1.) Create your workflow as described in the workflows section of this documentation for this tool. Ensure there are no syntax errors before running and that you know the path to the file.

2.) Open the cmd window (⊞ Win then type cmd)

3.) Navigate to the root directory of this repository. The recommended path for this is **c:\reports_creation** to keep the path as short as possible. To navigate to the directory use the following commands
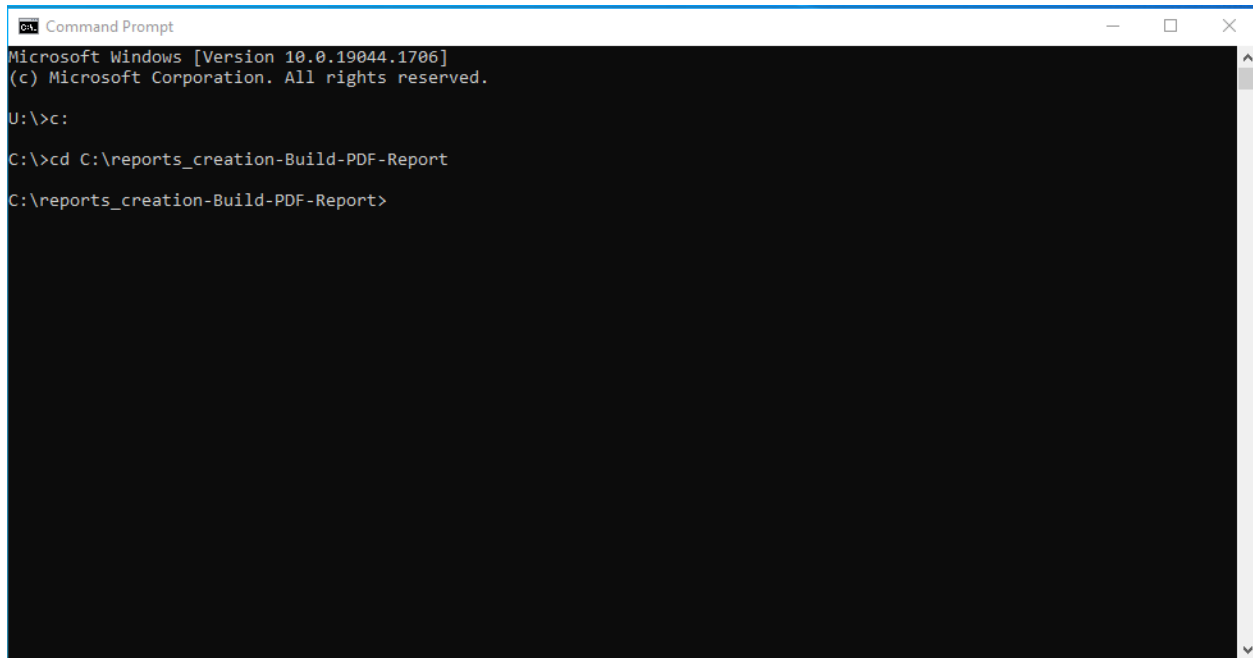
Use the below command if the selected drive does not match the drive containing the files from thi repository

C:

Change to whatever the drive you stored the files. To navigate to the correct directory use the cd command. For example:

cd C:\reports_creation

the above example would change the active directory to the reports_creation directory. With the correct directory assigned we can now run the tools. The below image shows the complete process that is broken down above.

```
Command Prompt                                              —    □    ×

Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

U:\>c:

C:\>cd C:\reports_creation-Build-PDF-Report

C:\reports_creation-Build-PDF-Report>
```
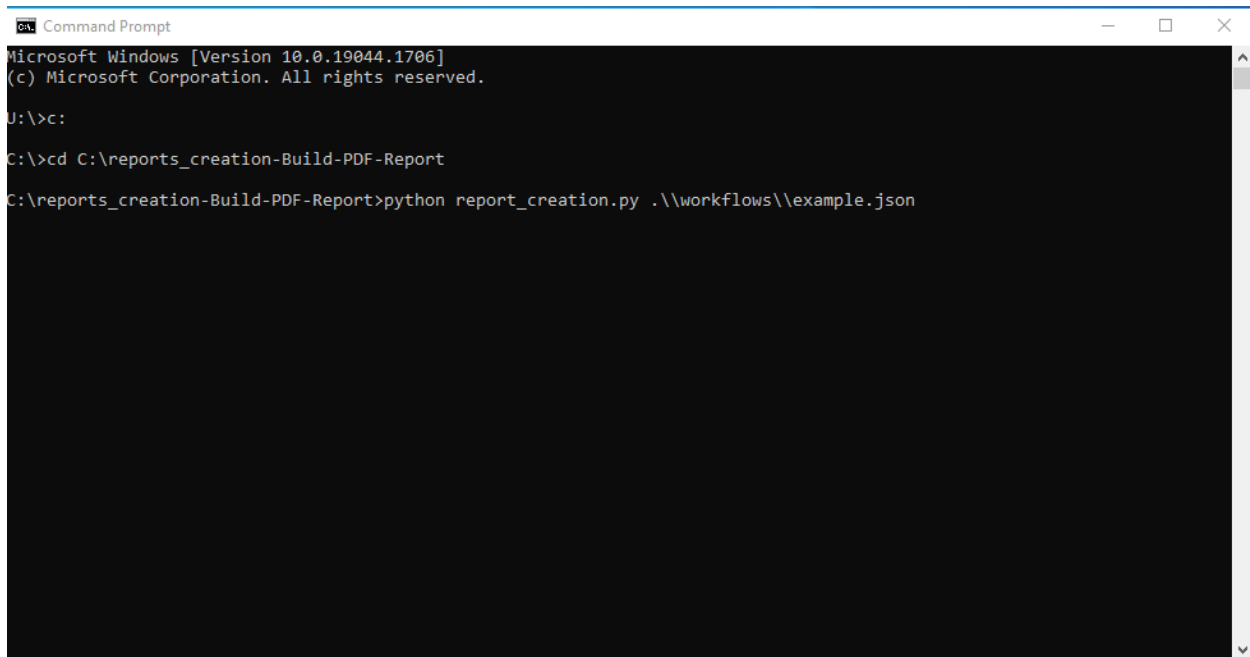
4.Activate the tool by creating a command using the following formula:

python <tool name .py> <path to workflow>

An example of a valid command for the data download tool using a workflow file called download_workflow.json which is located in the workflows folder of this repository would look as follows:

python data_download.py .\\workflows\\download_workflow.json

5.) Once the command is constructed hit ⏎ Enter to run it.

```
Command Prompt                                                                    —    □    ×
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

U:\>c:

C:\>cd C:\reports_creation-Build-PDF-Report

C:\reports_creation-Build-PDF-Report>python report_creation.py .\\workflows\\example.json
```

6.) If the command was valid the tool will begin to process the workflow file. While running the tool it will produce a series of messages complete with time stamps to give the user updates on the tools progress and any significant events that might have occurred.

```
[2024-04-26 07:57:03] [INFO] Extracting order from the input workflow
[2024-04-26 07:57:03] [INFO] Producing 2 PDP reports
[2024-04-26 07:57:03] [INFO] Producing PDP Report for: 10004
[2024-04-26 07:57:11] [INFO] Report Generated
[2024-04-26 07:57:11] [INFO] Producing PDP Report for: 24021
[2024-04-26 07:57:17] [INFO] Report Generated
[2024-04-26 07:57:17] [INFO] Exporting reports to: J:\EMRP\Work\GAM_Reports
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.pdf
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_10004.xlsx
[2024-04-26 07:57:17] [INFO] Exporting PD_PROF_24021.xlsx
[2024-04-26 07:57:17] [INFO] DONE!
```

There are three types of messages that can appear in the console:

- INFO: Informational messages on the current action the tool is processing.
- WARNING: Something occurred that was outside the normal parameters of the tool but did not inhibit processing. An example of the common warning for this tool is No data available for the specified FED.
- ERROR: Something occurred that was significant enough to inhibit processing.

Outputs

On a successful run of the tool the following outputs will be produced:

1.) A .zip file for each FED that contains all map and report files that were available in the provided directories when the tool was run.

2.) A log file containing all the messages that were printed to the cmd window. This serves as a record of the process and allows the user to check for errors after processing. This file will be located in the logs folder at the root of this repository and all log files will use this naming convention: .log with the date following the YYYY-MM-DD convention.

Further Automation

A method that can be used to further automate these tools is to create a .bat file containing the commands from the above section. This can be used to chain several workflows together as needed. An example .bat file called 'example.bat' can be found in the workflows folder as a guide for creating your own custom .bat files.