Python SDK

Contents:

The Trifacta® Python SDK enables you to integrate the Designer Cloud application into your Python pipelines. When your Python environment has been integrated with the Designer Cloud application , you can leverage the visual tools in the application to rapidly construct your transformation steps on exampled data that you upload. When you have finished building your recipe, you can invoke a function in your Python environment to download the recipe as Python Pandas code for use in your data pipelines.

Basic workflow:

1. Through your Python notebook:
   a. Upload example data to your Trifacta workspace.
   b. Launch the Designer Cloud application .
2. In the Designer Cloud application :
   a. Use the transformation tools in the application to transform your example data using a series of recipe steps.
   b. Iterate on your recipe. Generate results through the Designer Cloud application to verify that you have transformed your data correctly.
3. In your Python notebook:
   a. Invoke a function to translate the recipe into Python Pandas and download it to your local Python environment.
   b. Deploy this recipe into other Python pipelines to transform other datasets as needed.

Prerequisites

Trifacta prerequisites

- A workspace administrator must enable the Python to Wrangle feature in your workspace. For more information, see *Workspace Settings Page*.
- You must have a valid API access token. For more information, see *Manage API Access Tokens*.

Python prerequisites

Please see the installation instructions available at the download URL listed below.

Limitations

> **NOTE:** This is an Alpha release. Do not use the Python SDK in a production environment.

- Some Wrangle functions and transformations are not supported by Python Pandas. Known limitations:

    - NUMFORMAT function
    - String comparison functions
- Transformations that use Array or Map data types are not supported for Python Pandas generation.
- Uploaded files must be in CSV file format.

Download and Install

For more information on downloading and installing the Python SDK, see *https://pypi.org/project/trifacta/*.

Examples

For a basic example, please see *https://pypi.org/project/trifacta/*.

Wrangle function reference

The following wrangling functions are available through the SDK.

Trifacta module functions

`tf` is an alias to the Trifacta module.

Function Name	Description	Arguments
`tf.wrangle(*datasets)`	Upload one ore more datasets to the Designer Cloud application and create a flow for it. This flow is then available through the Designer Cloud application , where you can transform the dataset through the user interface. See *https://pypi.org/project/trifacta/*.	***datasets**: Pandas DataFrames to be wrangled. It could also be a tuple, where the first element in the tuple is a Pandas DataFrame, and second element is the reference name (string) for the DataFrame.

WrangleFlow module functions

All the below functions are available for the `WrangleFlow` object in your Python environment. So, you must call them using a `WrangleFlow` object.

`wf` is a reference to the WrangleFlow object.

Function Name	Description	Arguments
`wf.add_datasets(*datasets)`	Add Pandas DataFrames to a flow, where `datasets` is a list of DataFrames.	***datasets**: Pandas DataFrames to be added to a flow. It could also be a tuple, where the first element in the tuple is a Pandas DataFrame, and second element is the reference name (string) for the DataFrame.
`get_pandas(add_to_next_cell=False, recipe_name="<my_recipe>")`	Generates Python Pandas code for your Wrangle recipe.	**add_to_next_cell**: Set it to True, if you're using Jupyter Notebook and would like to add the generated Pandas code to be added to next cell. If False, the Pandas code is returned as string. **recipe_name**: Recipe for which you want to get the Pandas code. If not specified, the default recipe is used. Use `wf.recipe_names()` to retrieve available recipes.
`wf.run_job(pbar=None, execution='photon', recipe_name=None)`	Run a job for a specified recipe.	**pbar:** can be ignored. **execution**: Running environment in Designer Cloud powered by Trifacta platform where you want to execute the job. Possible values: `photon` or `emrSpark`. **recipe_name**: Recipe for which you want to execute the job. If set to `None`, input is the default recipe.

`wf.profile (recipe_name=None)`	Generate a profile for a specified recipe.	**recipe_name**: Recipe for which you want to generate profile. If set to `None`, input is the default recipe.
`wf.recipe_names()`	Lists the recipe names for the recipe present in Trifact a flow.	N/A
`wf.open_profile (recipe_name=None)`	Open a profile that you have previously generated for the specified recipe.	**recipe_name**: Recipe for which you want to open the profile. If set to `None`, input is the default recipe.

Data profiling functions

Function Name	Description	Arguments
`wf.summary (recipe_name=None)`	Returns a table of summary statistics per column	**recipe_name:** Recipe name for which you want to generate the summary. If set to `None`, input is the default recipe.
`wf.dq_bars (show_types=True, recipe_name=None)`	Returns the valid/invalid/missing ratio per column	**show_types**: Show column types information along with data quality bars for the column. **recipe_name**: Recipe name for which you want to generate the data quality bar. If set to `None`, input is the default recipe.
`wf.col_types (recipe_name=None)`	Lists the inferred data type for each column	**recipe_name**: Recipe name for which you want to infer data types for each column. If set to `None`, input is the default recipe.
`wf.bars_df_list (recipe_name=None)`	Returns a list of dataframes, one per column, representing a bar-chart for that column	**recipe_name:** Recipe name for which you want to generate the bar-chart. If set to `None`, input is the default recipe.
`wf.pdf_profile (filename=None, recipe_name=None)`	Returns a snazzy PDF report with all the statistics	**filename**: Name of the file to which PDF profile results are written. If set to `None`, results are returned back from the function. **recipe_name**: Recipe for which you want to generate PDF profile results. If set to `None`, results are generated for the default recipe.