

# DemystData PYTHON CHEAT SHEET

```
from demyst.analytics import Analytics
```

## AUTHORIZATION

### 1. Username and password

```
analytics_up =  
Analytics(username="EMAIL ADDRESS",  
password="DEMYST PASSWORD")
```

### 2. API Key

```
analytics_key =  
Analytics(key="REPLACE W/ API KEY")
```

### 3. Single Sign on/SAML

```
analytics_sso =  
Analytics(username="EMAIL ADDRESS",  
jwt="REPLACE W/ TOKEN")
```

## SEARCH

### 1. Product search using input

```
analytics.search(input_dataframe)
```

### 2. Product search w/exact match for inputs

```
analytics.search(input_dataframe,  
strict=True)
```

### 3. Response Attribute search

```
analytics.attribute_search("ATTRIBUTE  
_NAME")
```

## CATALOG

### 1. List all data products and details

```
analytics.product()
```

### 2. See catalog for a product

```
analytics.product_catalog(["PRODUCT/S  
NAME"])
```

### 3. See catalog for all products

```
analytics.product_catalog(all_products  
=TRUE])
```

### 4. View Inputs of a product

```
analytics.product_inputs(["PRODUCT/S  
NAME"])
```

### 5. View Outputs of a product

```
analytics.product_outputs(["PRODUCT/S  
NAME"])
```

## INPUTS

### 1. Validate input dataframe

```
analytics.validate(input_dataframe)
```

### 2. Validate input dataframe against product input

```
analytics.validate(input_dataframe,  
providers=["PRODUCT/S NAME"])
```

### 3. Load input (Beta Version for automated type guessing)

```
analytics.load_input(input_dataframe)
```

## ENRICH

### 1. Get enriched data for your input dataframe

```
analytics.enrich_and_download(["PRODU  
CT/S NAME"], input_dataframe)
```

### 2. Run enrichment asynchronously

```
analytics.enrich(["PRODUCT/S NAME"],  
input_dataframe)
```

### 3. Download the enrichment result from .enrich()

```
analytics.enrich_download(enrich_job_  
id)
```

### 4. Check if enrichment is complete

```
analytics.enrich_status(enrich_job_id)
```

### 5. Wait for enrichment completion

```
analytics.enrich_wait(enrich_job_id)
```

### 6. Cost of an enrichment

```
analytics.enrich_credits(["PRODUCT/S  
NAME"], input_dataframe)
```

## STATS

### 1. Statistics on enriched data

```
from demyst.analytics.report import *  
report(input_dataframe, enrich_result)
```

### 2. Statistics on product and it's attributes

```
analytics.product_stats(["PRODUCT/S  
NAME"])
```