

---

# **salesforce-reporting-chunked**

***Release 0.1.2***

**Jun 05, 2019**



---

## Contents:

---

<b>1</b>	<b>Chunky</b>	<b>1</b>
<b>2</b>	<b>report_chunker</b>	<b>5</b>
<b>3</b>	<b>Indices and tables</b>	<b>7</b>
	<b>Python Module Index</b>	<b>9</b>
	<b>Index</b>	<b>11</b>



# CHAPTER 1

---

## Chunky

---

Uses salesforce\_reporting module to extract chunked data by use of a “Time Frame” column in sales force.

```
class salesforce_reporting_chunked.chunky.Chunky(username=None, password=None,
                                                security_token=None, sandbox=False, api_version='v29.0')
```

Uses salesforce\_reporting module to extract chunked data by use of a “Time Frame” column in sales force.  
Requires a Salesforce account and security token.

### Parameters

- **username** (*str*) – Salesforce username
- **password** (*str*) – Salesforce password
- **security\_token** (*str*) – Salesforce security token
- **sandbox** (*bool*) – Run report in Salesforce sandbox (default False)
- **api\_version** (*str*) – Salesforce reporting API version (default v29.0)

### Example

```
>>> from salesforce_reporting_chunked import chunk_report_by_date
>>> CONFIG = {
...     "security_token": "REPLACE WITH YOUR TOKEN",
...     "username": "REPLACE WITH YOUR USERNAME",
...     "password": "REPLACE WITH YOUR PASSWORD",
...     "api_version": "v38.0",
... }
>>> FIELDNAMES = [
...     "First Name",
...     "Last Name",
...     "Date Column", # this is the magic column used for chunking.
...     "Corhuscorrated Plethanth",
...     "Other Column",
```

(continues on next page)

(continued from previous page)

```

... ]
>>> REPORT_ID = "YOURREPORTID"
>>> data = chunk_report_by_date(
...     CONFIG,
...     REPORT_ID,
...     FIELDNAMES,
...     date_fieldname="Date Column",
...     start_date="2018-01-01",
...     start_date="2019-01-31",
... )
>>> next(data)
OrderedDict([('First Name', 'Fred'), ('Last Name', 'Garvin'), ('DATE_COLUMN_NAME',
↪ '2018-01-01'), ('Corhuscorrated Plethanth', True), ('Other Column': 'Yep. Another
↪ ')])

```

**`_get_report_filtered`** (*url*, *filters=None*, *standard\_date\_filter=None*)

Filter report on filters and/or standard\_date\_filter.

#### Parameters

- **`url`** (*str*) –
- **`filters`** (*list*) –
- **`standard_date_filter`** (*dict*) –

**Returns** requests.post().json() (dict) Salesforce reports object.

#### Example

```

>>> # standard_date_filter JSON object as described in https://developer.
↪ salesforce.com/docs/atlas.en-us.api_analytics.meta/api_analytics/sforce_
↪ analytics_rest_api_getbasic_reportmetadata.htm
>>> {
...     'column': 'foo.TheDate',
...     'durationValue': 'CUSTOM',
...     'endDate': '2019-01-01',
...     'startDate': '2019-01-01',
... }

```

**`get_daterange_chunked_report`** (*report\_id*, *filters=None*, *details=True*, *date\_fieldname=None*,  
*start\_date=None*, *end\_date=None*, *day\_increment=1*)

Get chunked report by daterange. Anything more than 1 may result in unforeseen results, so think it through.

#### Parameters

- **`report_id`** (*str*) – Final portion of Salesforce API endpoint for report.
- **`filters`** (*list*) – List of dictionaries in Salesforce “reportFilters” format. {field: filter}, optional.
- **`details`** (*bool*) – Whether or not detail rows are included in report output, default True
- **`date_fieldname`** (*str*) – Column name of sortable date field from Salesforce report page.
- **`start_date`** (*str*) – iso-formatted date string. ex: “2019-01-01”.
- **`end_date`** (*str*) – iso-formatted date string. ex: “2019-01-01”.

- **day\_increment** (*int*) – Number of days to “chunk” report by. Default 1.

**Yields** *row* (*OrderedDict*) – report row

### Example

```
>>> REPORT_ID = "abc123youandmegirl"
>>> data = get_daterange_chunked_report(REPORT_ID, date_fieldname="The_Date",
↳ start_date="2019-06-01", end_date="2019-06-30")
>>> next(data)
```

**get\_report** (*report\_id*, *filters=None*, *standard\_date\_filter=None*, *details=True*)

Return the full JSON content of a Salesforce report, with or without filters.

#### Parameters

- **report\_id** (*str*) – Final portion of Salesforce API endpoint for report.
- **filters** (*list*) – List of dictionaries in Salesforce “reportFilters” format. {field: filter}, optional.
- **details** (*bool*) – Whether or not detail rows are included in report output, default True

**Returns** Salesforce report

**Return type** report (json)

**salesforce\_reporting\_chunked.chunky.\_sdf\_fieldname\_from\_label** (*metadata*, *standard\_date\_filter*)

Update the “column” value of standard\_date\_filter dictionary with internal date-sortable fieldname.

#### Parameters

- **metadata** (*dict*) –
- **standard\_date\_filter** (*dict*) –

**Returns** standard\_date\_filter (dict)

### Example

```
>>> standard_date_filter = {
...     "column": "CREATED_DATE",
...     "durationValue": "CUSTOM",
...     "endDate": "2019-01-01",
...     "startDate": "2019-06-30",
... }
>>> metadata = {
...     "reportExtendedMetadata": {
...         "detailColumnInfo": {
...             "weird_internal_name__c": {
...                 "label": "CREATED_DATE",
...                 "dataType": "string",
...             }
...         }
...     }
... }
>>> _sdf_fieldname_from_label(metadata, standard_date_filter)
{'column': 'weird_internal_name__c', 'durationValue': 'CUSTOM', 'endDate': '2019-
↳ 01-01', 'startDate': '2019-06-30'}
```





## CHAPTER 2

---

### report\_chunker

---

Contains wrapper function *chunk\_report\_by\_date*. Allows one to get report data with > 2000 rows.

```
salesforce_reporting_chunked.report_chunker.chunk_report_by_date(config, re-  
                                                                port_id,  
                                                                fieldnames,  
                                                                date_fieldname,  
                                                                start_date,  
                                                                end_date,  
                                                                day_increment=1)
```

#### Parameters

- **config** (*dict*) – Dictionary containing username, password, security\_token and api\_version.
- **report\_id** (*str*) – Salesforce report id.
- **fieldnames** (*list*) – Columns from Salesforce report.
- **date\_fieldname** (*str*) – Name of sortable date fieldname used to get incremental chunks of report.
- **start\_date** (*str*) – iso-formatted date string
- **end\_date** (*str*) – iso-formatted date string
- **day\_increment** (*int*) – Number of days in an incremental chunk.

**Yields** *row* (*OrderedDict*) – Report row value as a python OrderedDict object.

#### Example

```
>>> CONFIG = {"security_token": "br549", "username": "your@example.com", "password"  
→": "ultrasecret", "api_version": "v42.0" }  
>>> REPORT_ID = "abc123xyz789"  
>>> FIELDNAMES = ["Foo", "The_Date", "Bar"]
```

(continues on next page)

(continued from previous page)

```
>>> data = chunk_report_by_date(CONFIG, REPORT_ID, FIELDNAMES, date_fieldname=
↳ "The_Date", start_date="2019-06-01", end_date="2019-07-01")
>>> print(next(data))
OrderedDict([('Foo', 'The thing'), ('The_Date', '6/1/2019') ('Bar', 'The other_
↳ thing')])
```

## CHAPTER 3

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### S

`salesforce_reporting_chunked.chunky`, [1](#)  
`salesforce_reporting_chunked.report_chunker`,  
[3](#)



## Symbols

`_get_report_filtered()` (*salesforce\_reporting\_chunked.chunky.Chunky* method), 2

`_sdf_fieldname_from_label()` (in module *salesforce\_reporting\_chunked.chunky*), 3

## C

`chunk_report_by_date()` (in module *salesforce\_reporting\_chunked.report\_chunker*), 5

`Chunky` (class in *salesforce\_reporting\_chunked.chunky*), 1

## G

`get_daterange_chunked_report()` (*salesforce\_reporting\_chunked.chunky.Chunky* method), 2

`get_report()` (*salesforce\_reporting\_chunked.chunky.Chunky* method), 3

## S

`salesforce_reporting_chunked.chunky` (module), 1

`salesforce_reporting_chunked.report_chunker` (module), 3