
ProKnow DS - Python SDK Documentation

Kyle Burnett

Jan 21, 2019

Contents:

1	Usage	3
1.1	Installation	3
1.2	Basic Usage	3
2	API Reference	5
2.1	<code>proknow.ProKnow</code> – ProKnow	5
2.2	<code>proknow.Requestor</code> – Requestor	6
2.3	<code>proknow.Exceptions</code> – Exceptions	8
2.4	<code>proknow.CustomMetrics</code> – Custom Metrics	9
2.5	<code>proknow.Workspaces</code> – Workspaces	11
2.6	<code>proknow.Roles</code> – Roles	14
2.7	<code>proknow.Users</code> – Users	16
2.8	<code>proknow.Patients</code> – Patients	19
3	Contributing Guide	25
3.1	Requirements	25
3.2	Accepted Contributions	25
4	Indices and tables	27
	Python Module Index	29

Welcome to the ProKnow SDK documentation. Use the links below to access the usage page and the complete api documentation.

CHAPTER 1

Usage

Read this guide to learn how to install ProKnow SDK and get started with your first script.

1.1 Installation

Use pip to install:

```
pip install proknow
```

1.2 Basic Usage

To use the SDK, you'll need to generate an [API token](#) in the ProKnow DS user interface. Once you have a credentials file, use it in your script to create an instance of the main ProKnow class:

```
from proknow import ProKnow

pk = ProKnow('https://example.proknow.com', credentials_file="./credentials.json")
workspaces = pk.workspaces.query()
print("The workspaces are as follows:")
for workspace in workspaces:
    print(workspace.name)
```


This guide includes a complete API reference.

2.1 proknow.ProKnow – ProKnow

class proknow.ProKnow.**ProKnow**(*base_url*, *credentials_file*=None, *credentials_id*=None, *credentials_secret*=None)

This is the main class that should be instantiated at the beginning of your Python program with your base URL (which should include your account subdomain) and your API token credentials.

requestor

proknow.Requestor.Requestor – An instance of the Requestor class.

custom_metrics

ProKnow.CustomMetrics.CustomMetrics – An instance of the Custom Metrics class.

workspaces

proknow.Workspaces.Workspaces – An instance of the Workspaces class.

roles

proknow.Roles.Roles – An instance of the Roles class.

users

proknow.Users.Users – An instance of the Users class.

patients

proknow.Patients.Patients – An instance of the Patients class.

__init__(*base_url*, *credentials_file*=None, *credentials_id*=None, *credentials_secret*=None)

Initializes the ProKnow class.

The *base_url* must be provided as should either the *credentials_file* or both the *credentials_id* and *credentials_secret*.

Parameters

- **base_url** (*str*) – The base URL to use when making request to the ProKnow API.

- **credentials_file** (*str*) – The path to the credentials file obtained by creating an API key in your ProKnow account. The file should be a JSON file containing an object with the fields *id* and *secret*.
- **credentials_id** (*str*) – An API key id.
- **credentials_secret** (*str*) – An API key secret.

Raises `AssertionError` – If the input parameters are invalid.

2.2 proknow.Requestor – Requestor

class proknow.Requestor.**Requestor** (*base_url, username, password*)

A class used for issuing requests for the ProKnow API

__init__ (*base_url, username, password*)

Initializes the Requestor class.

Parameters

- **base_url** (*str*) – The base URL to use when making request to the ProKnow API.
- **username** (*str*) – The string used in Basic Authentication as the user name.
- **password** (*str*) – The string used in Basic Authentication as the user password.

delete (*route, query=None, body=None*)

Issues an HTTP DELETE request.

Parameters

- **route** (*str*) – The API route to use in the request.
- **query** (*dict, optional*) – An optional dictionary of query parameters to use in the request.
- **body** (*dict, optional*) – An optional dictionary to be used as a JSON request body for the request.

Returns

A tuple (status_code, msg).

1. status_code (int): the response code
2. msg (str, dict): the text response or, if the response was JSON, the decoded JSON dictionary.

Return type

 tuple

get (*route, query=None*)

Issues an HTTP GET request.

Parameters

- **route** (*str*) – The API route to use in the request.
- **query** (*dict, optional*) – An optional dictionary of query parameters to use in the request.

Returns

A tuple (status_code, msg).

1. status_code (int): the response code

2. `msg` (`str`, `dict`): the text response or, if the response was JSON, the decoded JSON dictionary.

Return type `tuple`

patch (`route`, `query=None`, `body=None`)

Issues an HTTP PATCH request.

Parameters

- **route** (`str`) – The API route to use in the request.
- **query** (`dict`, `optional`) – An optional dictionary of query parameters to use in the request.
- **body** (`dict`, `optional`) – An optional dictionary to be used as a JSON request body for the request.

Returns

A tuple (`status_code`, `msg`).

1. `status_code` (`int`): the response code
2. `msg` (`str`, `dict`): the text response or, if the response was JSON, the decoded JSON dictionary.

Return type `tuple`

post (`route`, `query=None`, `body=None`)

Issues an HTTP POST request.

Parameters

- **route** (`str`) – The API route to use in the request.
- **query** (`dict`, `optional`) – An optional dictionary of query parameters to use in the request.
- **body** (`dict`, `optional`) – An optional dictionary to be used as a JSON request body for the request.

Returns

A tuple (`status_code`, `msg`).

1. `status_code` (`int`): the response code
2. `msg` (`str`, `dict`): the text response or, if the response was JSON, the decoded JSON dictionary.

Return type `tuple`

put (`route`, `query=None`, `body=None`)

Issues an HTTP PUT request.

Parameters

- **route** (`str`) – The API route to use in the request.
- **query** (`dict`, `optional`) – An optional dictionary of query parameters to use in the request.
- **body** (`dict`, `optional`) – An optional dictionary to be used as a JSON request body for the request.

Returns

A tuple (status_code, msg).

1. status_code (int): the response code
2. msg (str, dict): the text response or, if the response was JSON, the decoded JSON dictionary.

Return type tuple

stream (route, path)

Issues an HTTP PUT request.

Parameters

- **route** (str) – The API route to use in the request.
- **path** (str) – The file path to stream the request response
- **query** (dict, optional) – An optional dictionary of query parameters to use in the request.
- **body** (dict, optional) – An optional dictionary to be used as a JSON request body for the request.

2.3 proknow.Exceptions – Exceptions

exception proknow.Exceptions.ProKnowError

All errors involved in interacting with the ProKnow DS - Python SDK modules extend this.

exception proknow.Exceptions.HttpError (status_code, body)

Errors produced at the HTTP layer.

status_code

int – The status code.

body

str – The response body.

__init__ (status_code, body)

Initializes the WorkspaceLookupError class.

Parameters

- **status_code** (int) – The status code.
- **body** (str) – The response body.

exception proknow.Exceptions.WorkspaceLookupError (message)

Indicates that there was a problem looking up the workspace by the given identifier.

message

str – An explanation of the error.

__init__ (message)

Initializes the WorkspaceLookupError class.

Parameters **message** (str) – An explanation of the error.

exception proknow.Exceptions.CustomMetricLookupError (message)

Indicates that there was a problem looking up the custom metric by the given identifier.

message*str* – An explanation of the error.**__init__** (*message*)

Initializes the CustomMetricLookupError class.

Parameters **message** (*str*) – An explanation of the error.

2.4 proknow.CustomMetrics – Custom Metrics

class proknow.CustomMetrics.CustomMetrics (*proknow, requestor*)

This class should be used to interact with the custom metrics in a ProKnow organization. It is instantiated for you as an attribute of the `proknow.ProKnow.ProKnow` class.

__init__ (*proknow, requestor*)

Initializes the CustomMetrics class.

Parameters

- **proknow** (`proknow.ProKnow.ProKnow`) – The ProKnow instance that is instantiating the object.
- **requestor** (`proknow.Requestor.Requestor`) – An object used to make API requests.

create (*name, context, metric_type*)

Creates a new custom metric.

Parameters

- **name** (*str*) – The custom metric name.
- **context** (*str*) – The custom metric context.
- **metric_type** (*dict*) – The custom metric type.

Returns A representation of the created custom metric.**Return type** `proknow.CustomMetrics.CustomMetricItem`**Raises** `AssertionError` – If the input parameters are invalid.**delete** (*custom_metric_id*)

Deletes a custom metric by id.

Parameters **custom_metric_id** (*str*) – The id of the custom metric to delete.**Raises** `AssertionError` – If the input parameters are invalid.**find** (*predicate=None, **props*)

Finds the first custom metric that matches the input paramters.

Note: This method utilizes a cache of custom metrics. Once it has a cache of custom metrics, it will use that cache until the `proknow.CustomMetrics.CustomMetrics.query()` method is called to refresh the cache. If you wish to make your code resilient to custom metric changes (i.e., new custom metrics, renamed custom metrics, deleted custom metrics, etc.) while your script is running, you should call the `proknow.CustomMetrics.CustomMetrics.query()` method before this method. In most use cases, this is not necessary.

Parameters

- **predicate** (*func*) – A function that is passed a metric as input and which should return a bool indicating whether the metric is a match.
- ****props** – A dictionary of keyword arguments that may include any custom metric attribute. These arguments are considered in turn to find matching custom metrics.

Returns A representation of the matching custom metric.

Return type `proknow.CustomMetrics.CustomMetricItem`

query ()

Queries for custom metrics.

Note: This method refreshes the custom metric cache.

Returns A list of `proknow.CustomMetrics.CustomMetricItem` objects, each representing a custom metric in the organization.

Return type list

resolve (*custom_metric*)

Resolves a metric by id or name.

Parameters **custom_metric** (*str*) – The metric id or name.

Returns A representation of the resolved custom metric.

Return type `proknow.CustomMetrics.CustomMetricItem`

Raises

- `AssertionError` – If the input parameters are invalid.
- `proknow.Exceptions.CustomMetricLookupError` – If the custom metric with the given id or name could not be found.

resolveById (*custom_metric_id*)

Resolves a custom metric id to a custom metric.

Parameters **custom_metric_id** (*str*) – The custom metric id.

Returns A representation of the resolved custom metric.

Return type `proknow.CustomMetrics.CustomMetricItem`

Raises

- `AssertionError` – If the input parameters are invalid.
- `proknow.Exceptions.CustomMetricLookupError` – If the custom metric with the given id could not be found.

resolveByName (*name*)

Resolves a custom metric name to a custom metric.

Parameters **name** (*str*) – The custom metric name.

Returns A representation of the resolved custom metric.

Return type `proknow.CustomMetrics.CustomMetricItem`

Raises

- `AssertionError` – If the input parameters are invalid.

- `proknow.Exceptions.CustomMetricLookupError` – If the custom metric with the given name could not be found.

class `proknow.CustomMetrics.CustomMetricItem` (*custom_metrics, custom_metric*)
 This class represents a custom metric. It's instantiated by the `proknow.CustomMetrics.CustomMetrics` class to represent each of the custom metrics in a query result and a created custom metric.

id

str – The id of the custom metric (readonly).

data

dict – The complete representation of the custom metric as returned from the API (readonly).

name

str – The name of the custom metric.

context

str – The context of the custom metric.

type

dict – The type of the custom metric (readonly).

__init__ (*custom_metrics, custom_metric*)

Initializes the CustomMetricItem class.

Parameters

- **custom_metrics** (`proknow.CustomMetrics.CustomMetrics`) – The CustomMetrics instance that is instantiating the object.
- **custom_metric** (*dict*) – A dictionary of custom metric attributes.

delete ()

Deletes the custom metric.

save ()

Saves the changes made to a custom metric.

Example

The following example illustrates how to find a custom metric by its name, change the name, and save it:

```
pk = ProKnow('https://example.proknow.com', credentials_file="./credentials.
↪json")
metric = pk.custom_metric.find(name='Type')
metric.name = "Genetic Type"
metric.save()
```

2.5 proknow.Workspaces – Workspaces

class `proknow.Workspaces.Workspaces` (*proknow, requestor*)

This class should be used to interact with the workspaces in a ProKnow organization. It is instantiated for you as an attribute of the `proknow.ProKnow.ProKnow` class.

__init__ (*proknow, requestor*)

Initializes the Workspaces class.

Parameters

- **proknow** (`proknow.ProKnow.ProKnow`) – The ProKnow instance that is instantiating the object.
- **requestor** (`proknow.Requestor.Requestor`) – An object used to make API requests.

create (*slug, name, protected=True*)

Creates a new workspace.

Parameters

- **slug** (*str*) – The workspace slug.
- **name** (*str*) – The workspace name.
- **protected** (*bool, optional*) – Indicates whether the workspace should be protected from accidental deletion.

Returns A representation of the created workspace.**Return type** `proknow.Workspaces.WorkspaceItem`**Raises** `AssertionError` – If the input parameters are invalid.**delete** (*workspace_id*)

Deletes a workspace.

Parameters **workspace_id** (*str*) – The id of the workspace to delete.**Raises** `AssertionError` – If the input parameters are invalid.**find** (*predicate=None, **props*)

Finds the first workspace that matches the input paramters.

Note: This method utilizes a cache of workspaces. Once it has a cache of workspaces, it will use that cache until the `proknow.Workspaces.Workspaces.query()` method is called to refresh the cache. If you wish to make your code resilient to workspaces changes (i.e., new workspaces, renamed workspaces, deleted workspaces, etc.) while your script is running, you should call the `proknow.Workspaces.Workspaces.query()` method before this method. In most use cases, this is not necessary.

Parameters

- **predicate** (*func*) – A function that is passed a workspace as input and which should return a bool indicating whether the workspace is a match.
- ****props** – A dictionary of keyword arguments that may include any workspace attribute. These arguments are considered in turn to find matching workspaces.

Returns A representation of the matching workspace.**Return type** `proknow.Workspaces.WorkspaceItem`**query** ()

Queries for workspaces.

Note: This method refreshes the workspaces cache.

Returns A list of `proknow.Workspaces.WorkspaceItem` objects, each representing a workspace in the organization.

Return type list

resolve (*workspace*)

Resolves a workspace by id or name.

Parameters **workspace** (*str*) – The workspace id or name.

Returns A representation of the resolved workspace.

Return type `proknow.Workspaces.WorkspaceItem`

Raises

- `AssertionError` – If the input parameters are invalid.
- `proknow.Exceptions.WorkspaceLookupError` – If the workspace with the given name could not be found.

resolveById (*workspace_id*)

Resolves a workspace id to a workspace.

Parameters **workspace_id** (*str*) – The workspace id.

Returns A representation of the resolved workspace.

Return type `proknow.Workspaces.WorkspaceItem`

Raises

- `AssertionError` – If the input parameters are invalid.
- `proknow.Exceptions.WorkspaceLookupError` – If the workspace with the given name could not be found.

resolveByName (*name*)

Resolves a workspace name to a workspace.

Parameters **name** (*str*) – The workspace name.

Returns A representation of the resolved workspace.

Return type `proknow.Workspaces.WorkspaceItem`

Raises

- `AssertionError` – If the input parameters are invalid.
- `proknow.Exceptions.WorkspaceLookupError` – If the workspace with the given name could not be found.

class `proknow.Workspaces.WorkspaceItem` (*workspaces, workspace*)

This class represents a workspace. It's instantiated by the `proknow.Workspaces.Workspaces` class to represent each of the workspaces in a query result and a created workspace.

id

str – The id of the workspace (readonly).

data

dict – The complete representation of the workspace as returned from the API (readonly).

slug

str – A string used in the URL that uniquely identifies the workspace.

name

str – The name of the workspace.

protected

bool – Indicates whether the workspace should be protected from accidental deletion.

__init__ (*workspaces, workspace*)

Initializes the WorkspaceItem class.

Parameters

- **workspaces** (*proknow.Workspaces.Workspaces*) – The Workspaces instance that is instantiating the object.
- **workspace** (*dict*) – A dictionary of workspace attributes.

delete ()

Deletes the workspace.

save ()

Saves the changes made to a workspace.

Example

The following example illustrates how to find a workspace by its slug, modify the name, and save it:

```
pk = ProKnow('https://example.proknow.com', credentials_file="./credentials.  
↪json")  
clinical = pk.workspaces.find(slug='clinical')  
clinical.name = "Clinical Patients"  
clinical.save()
```

2.6 proknow.Roles – Roles

class *proknow.Roles.Roles* (*proknow, requestor*)

This class should be used to interact with the roles in a ProKnow organization. It is instantiated for you as an attribute of the *proknow.ProKnow.ProKnow* class.

__init__ (*proknow, requestor*)

Initializes the Roles class.

Parameters

- **proknow** (*proknow.ProKnow.ProKnow*) – The ProKnow instance that is instantiating the object.
- **requestor** (*proknow.Requestor.Requestor*) – An object used to make API requests.

create (*name, permissions*)

Creates a new role.

Parameters

- **name** (*str*) – The name of the role.
- **permissions** (*dict*) – A dictionary of permissions.

Returns A representation of the created item.

Return type `proknow.Roles.RoleItem`

Raises `AssertionError` – If the input parameters are invalid.

delete (*role_id*)

Deletes a role.

Parameters **role_id** (*str*) – The id of the role to delete.

Raises `AssertionError` – If the input parameters are invalid.

find (*predicate=None, **props*)

Finds the first role that matches the input paramters.

Parameters

- **predicate** (*func*) – A function that is passed a role as input and which should return a bool indicating whether the role is a match.
- ****props** – A dictionary of keyword arguments that may include any role attribute. These arguments are considered in turn to find matching roles.

Returns A representation of the matching role.

Return type `proknow.Roles.RoleItem`

get (*role_id*)

Gets a role.

Parameters **role_id** (*str*) – The id of the role to get.

Returns an object representing a role in the organization

Return type `proknow.Roles.RoleItem`

query ()

Queries for roles.

Returns A list of `proknow.Roles.RoleSummary` objects, each representing a summarized role in the organization.

Return type list

class `proknow.Roles.RoleSummary` (*roles, role*)

This class represents a summary view of a role. It's instantiated by the `proknow.Roles.Roles.query()` method to represent each of the roles returned in a query result.

id

str – The id of the role (readonly).

name

str – The name of the role (readonly).

data

dict – The summary representation of the role as returned from the API (readonly).

__init__ (*roles, role*)

Initializes the RoleSummary class.

Parameters

- **roles** (`proknow.Roles.Roles`) – The Roles instance that is instantiating the object.
- **role** (*dict*) – A dictionary of role attributes.

get ()

Gets the complete representation of the role.

Returns an object representing a role in the organization

Return type `proknow.Roles.RoleItem`

class `proknow.Roles.RoleItem` (*roles, role*)

This class represents a role. It's instantiated by the `proknow.Roles.Roles` class as a complete representation of a role.

id

str – The id of the role (readonly).

data

dict – The complete representation of the role as returned from the API (readonly).

name

str – The name of the role.

permissions

dict – The dictionary of role permissions.

__init__ (*roles, role*)

Initializes the RoleItem class.

Parameters

- **roles** (`proknow.Roles.Roles`) – The Role instance that is instantiating the object.
- **role** (*dict*) – A dictionary of role attributes.

delete ()

Deletes the role.

save ()

Saves the changes made to a role.

Example

The following example illustrates how to find a role by its slug, modify a permission, and save it:

```
pk = ProKnow('https://example.proknow.com', credentials_file='./credentials.
↪json')
researchers = pk.roles.find(name='researchers')
researchers.permissions["organization_read"] = True
researchers.save()
```

2.7 proknow.Users – Users

class `proknow.Users.Users` (*proknow, requestor*)

This class should be used to interact with the users in a ProKnow organization. It is instantiated for you as an attribute of the `proknow.ProKnow.ProKnow` class.

__init__ (*proknow, requestor*)

Initializes the Users class.

Parameters

- **proknow** (`proknow.ProKnow.ProKnow`) – The ProKnow instance that is instantiating the object.

- **requestor** (`proknow.Requestor.Requestor`) – An object used to make API requests.

create (*email, name, role_id, password=None*)

Creates a new user.

Parameters

- **email** (*str*) – The email of the user.
- **name** (*str*) – The name of the user.
- **role_id** (*str*) – The id of the role for the user.
- **password** (*str, optional*) – The password of the user.

Returns A representation of the created item.

Return type `proknow.Users.UserItem`

Raises `AssertionError` – If the input parameters are invalid.

delete (*user_id*)

Deletes a user.

Parameters **user_id** (*str*) – The id of the user to delete.

Raises `AssertionError` – If the input parameters are invalid.

find (*predicate=None, **props*)

Finds a user by id, email, or name.

Parameters

- **predicate** (*func*) – A function that is passed a user as input and which should return a bool indicating whether the user is a match.
- ****props** – A dictionary of keyword arguments that may include any user attribute. These arguments are considered in turn to find matching users.

Returns A representation of the matching user.

Return type `proknow.Users.UserItem`

get (*user_id*)

Gets a user.

Parameters **user_id** (*str*) – The id of the user to get.

Returns an object representing a user in the organization

Return type `proknow.Users.UserItem`

query ()

Queries for users.

Returns A list of `proknow.Users.UserSummary` objects, each representing a summarized user in the organization.

Return type list

class `proknow.Users.UserSummary` (*users, user*)

This class represents a summary view of a user. It's instantiated by the `proknow.Users.Users.query()` method to represent each of the users returned in a query result.

id

str – The id of the user (readonly).

email

str – The email of the user (readonly).

name

str – The name of the user (readonly).

data

dict – The summary representation of the user as returned from the API (readonly).

__init__ (*users, user*)

Initializes the UserSummary class.

Parameters

- **users** (`proknow.Users.Users`) – The Users instance that is instantiating the object.
- **user** (*dict*) – A dictionary of user attributes.

get ()

Gets the complete representation of the user.

Returns an object representing a user in the organization

Return type `proknow.Users.UserItem`

class `proknow.Users.UserItem` (*users, user*)

This class represents a user. It's instantiated by the `proknow.Users.Users` class as a complete representation of a user.

id

str – The id of the user (readonly).

data

dict – The complete representation of the user as returned from the API (readonly).

name

str – The name of the user.

email

str – The email of the user.

active

bool – Indicates whether the user is active.

role_id

str – The id of the role for the user.

__init__ (*users, user*)

Initializes the UserItem class.

Parameters

- **users** (`proknow.Users.Users`) – The User instance that is instantiating the object.
- **user** (*dict*) – A dictionary of user attributes.

delete ()

Deletes the user.

save ()

Saves the changes made to a user.

Example

The following example illustrates how to find a user by its email, set it to inactive, and save it:

```
pk = ProKnow('https://example.proknow.com', credentials_file="./credentials.
↪json")
jsmith = pk.users.find(email='jsmith@example.com')
jsmith.active = False
jsmith.save()
```

2.8 proknow.Patients – Patients

class `proknow.Patients.Patients` (*proknow, requestor*)

This class should be used to interact with the patients in a ProKnow organization. It is instantiated for you as an attribute of the `proknow.ProKnow.ProKnow` class.

__init__ (*proknow, requestor*)

Initializes the Patients class.

Parameters

- **proknow** (`proknow.ProKnow.ProKnow`) – The ProKnow instance that is instantiating the object.
- **requestor** (`proknow.Requestor.Requestor`) – An object used to make API requests.

create (*workspace, mrn, name, birth_date=None, birth_time=None, sex=None*)

Creates a new patient.

Parameters

- **workspace** (*str*) – An id or name of the workspace in which to create the patient.
- **mrn** (*str*) – The MRN of the patient.
- **name** (*str*) – The name of the patient.
- **birth_date** (*str, optional*) – The birth date of the patient.
- **birth_time** (*str, optional*) – The birth time of the patient.
- **sex** (*str, optional*) – The sex of the patient.

Returns A representation of the created patient.

Return type `proknow.Patients.PatientItem`

Raises `AssertionError` – If the input parameters are invalid.

delete (*workspace_id, patient_id*)

Deletes a patient.

Parameters

- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to delete.

Raises `AssertionError` – If the input parameters are invalid.

find (*workspace, predicate=None, **props*)

Finds the first patient that matches the input paramters.

Parameters

- **workspace** (*str*) – An id or name of the workspace in which to query for patients.
- **predicate** (*func*) – A function that is passed a metric as input and which should return a bool indicating whether the metric is a match.
- ****props** – A dictionary of keyword arguments that may include any patient attribute. These arguments are considered in turn to find matching patients.

Returns A representation of the matching patient.

Return type `proknow.Patients.PatientSummary`

get (*workspace_id, patient_id*)

Gets a patient from the given workspace.

Parameters

- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to get.

Returns an object representing a patient in the organization

Return type `proknow.Patients.PatientItem`

lookup (*workspace, mrns*)

Looks up a collection of patients matching the given list of MRNs.

Parameters

- **workspace** (*str*) – An id or name of the workspace in which to query for patients.
- **mrns** (*list*) – A list of MRN string values.

Returns A list of `proknow.Patients.PatientSummary` objects that match the given MRNs.

Return type list

query (*workspace*)

Queries for patients.

Parameters **workspace** (*str*) – An id or name of the workspace in which to query for patients.

Returns A list of `proknow.Patients.PatientSummary` objects, each representing a summarized patient in the given workspace.

Return type list

class `proknow.Patients.PatientSummary` (*patients, workspace_id, patient*)

This class represents a summary view of a patient. It's instantiated by the `proknow.Patients.Patients.query()` method to represent each of the patients returned in a query result.

id

str – The id of the patient (readonly).

mrn

str – The patient medical record number or MRN (readonly). In the Proknow interface, this is referred to as the Patient ID.

name

str – The name of the patient (readonly).

birth_date*str* – The birth_date of the patient (readonly).**birth_time***str* – The birth_time of the patient (readonly).**sex***str* – The sex of the patient (readonly).**data***dict* – The summary representation of the patient as returned from the API (readonly).**__init__** (*patients, workspace_id, patient*)

Initializes the PatientSummary class.

Parameters

- **patients** (*proknow.Patients.Patients*) – The Patients instance that is instantiating the object.
- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient** (*dict*) – A dictionary of patient attributes.

get ()

Gets the complete representation of the patient.

Returns an object representing a patient in the organization.**Return type** *proknow.Patients.PatientItem***class** *proknow.Patients.PatientItem* (*patients, workspace_id, patient*)This class represents a patient. It's instantiated by the *proknow.Patients.Patients* class as a complete representation of a patient.**id***str* – The id of the patient (readonly).**data***dict* – The complete representation of the patient as returned from the API (readonly).**mrn***str* – The MRN of the patient.**name***str* – The name of the patient.**birth_date***str* – The birth_date of the patient.**birth_time***str* – The birth_time of the patient.**sex***str* – The sex of the patient.**metadata***dict* – The metadata of the patient.**__init__** (*patients, workspace_id, patient*)

Initializes the PatientItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (`str`) – The id of the workspace to which the patient belongs.
- **patient** (`dict`) – A dictionary of patient attributes.

delete()

Deletes the patient.

find_entities (`predicate=None, **props`)

Finds the entities for the patient matching the input paramters.

Parameters

- **predicate** (`func`) – A function that is passed a entity as input and which should return a bool indicating whether the entity is a match.
- ****props** – A dictionary of keyword arguments that may include any entity attribute to match. These arguments are considered in turn to find matching entities.

Returns A list of matching `proknow.Patients.EntitySummary` objects.

Return type list

get_metadata()

Gets the metadata dictionary and decodes the ids into metrics names.

Returns The dictionary of custom metric key-value pairs where the keys are the decoded custom metric names.

Return type dict

save()

Saves the changes made to a patient.

set_metadata (`metadata`)

Sets the metadata dictionary to an encoded version of the given metadata dictionary.

Params: metadata (dict): A dictionary of custom metric key-value pairs where the keys are the names of the custom metric.

class `proknow.Patients.StudySummary` (`patients, workspace_id, patient_id, study`)

This class represents a summary view of a study. It's instantiated by the `proknow.Patients.PatientItem()` class to represent each of the studies that belong to the patient.

id

`str` – The id of the study (readonly).

data

`dict` – The summary representation of the study as returned from the API (readonly).

__init__ (`patients, workspace_id, patient_id, study`)

Initializes the PatientItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (`str`) – The id of the workspace to which the patient belongs.
- **patient_id** (`str`) – The id of the patient to which the study belongs.
- **study** (`dict`) – A dictionary of study attributes.

class `proknow.Patients.EntitySummary` (*patients, workspace_id, patient_id, entity*)

This class represents a summary view of an entity. It's instantiated by the `proknow.Patients.StudySummary()` class to represent each of the entities that belong to the study.

id

str – The id of the entity (readonly).

data

dict – The summary representation of the entity as returned from the API (readonly).

__init__ (*patients, workspace_id, patient_id, entity*)

Initializes the PatientItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to which the entity belongs.
- **entity** (*dict*) – A dictionary of entity attributes.

get ()

Gets the entity item for the entity summary type.

Returns One of (`proknow.Patients.ImageSetItem`, `proknow.Patients.StructureSetItem`, `proknow.Patients.PlanItem`, `proknow.Patients.DoseItem`) based on the entity summary type.

class `proknow.Patients.EntityItem` (*patients, workspace_id, patient_id, entity*)

This class is a base class for specific entity types.

id

str – The id of the entity (readonly).

data

dict – The summary representation of the entity as returned from the API (readonly).

__init__ (*patients, workspace_id, patient_id, entity*)

Initializes the PatientItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to which the entity belongs.
- **entity** (*dict*) – A dictionary of entity attributes.

class `proknow.Patients.ImageSetItem` (*patients, workspace_id, patient_id, entity*)

Not implemented

class `proknow.Patients.StructureSetItem` (*patients, workspace_id, patient_id, entity*)

Not implemented

class `proknow.Patients.PlanItem` (*patients, workspace_id, patient_id, entity*)

This class represents a plan. It's instantiated by the `proknow.Patients.EntitySummary` class as a complete representation of a plan entity.

id

str – The id of the entity (readonly).

data

dict – The summary representation of the entity as returned from the API (readonly).

__init__ (*patients, workspace_id, patient_id, entity*)

Initializes the PlanItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to which the entity belongs.
- **entity** (*dict*) – A dictionary of entity attributes.

download (*path*)

Download the plan file.

Parameters **path** (*str*) – A path to a directory or file to which the plan file should be streamed.

class `proknow.Patients.DoseItem` (*patients, workspace_id, patient_id, entity*)

This class represents a dose. It's instantiated by the `proknow.Patients.EntitySummary` class as a complete representation of a dose entity.

id

str – The id of the entity (readonly).

data

dict – The summary representation of the entity as returned from the API (readonly).

__init__ (*patients, workspace_id, patient_id, entity*)

Initializes the DoseItem class.

Parameters

- **patients** (`proknow.Patients.Patients`) – The Patients instance that is instantiating the object.
- **workspace_id** (*str*) – The id of the workspace to which the patient belongs.
- **patient_id** (*str*) – The id of the patient to which the entity belongs.
- **entity** (*dict*) – A dictionary of entity attributes.

download (*path*)

Downloads the dose file.

Parameters **path** (*str*) – A path to a directory or file to which the dose file should be streamed.

This is an open source repository, and we appreciate your contributions. Use this guide to learn how you can be involved.

3.1 Requirements

1. You should discuss your change with a ProKnow developer before you begin. We know your time is valuable, and submitting a proposal of your change on our [GitHub issues page](#) is a great way to ensure that you will be working on something that we will likely merge once you submit a pull request.
2. You should follow the existing coding styles. We try to follow the guidelines presented in **PEP 8**. Should you find code that not comply with these guidelines, please bring it to our attention so that we may address it.
3. You should have a ProKnow DS account to run basic tests on your changes. If you do not have an account but still wish to contribute, please contact us at support@proknow.com.

3.2 Accepted Contributions

1. New features.
2. Bug fixes.
3. Documentation.
4. Tests.

We are unlikely to accept pull requests that modify the license or change the way we publish and deploy. For example, we would not accept a change that separated the source and build directories because it would change the way we published the documentation. Tests are the exception to this because we do not yet have a testing strategy in place.

CHAPTER 4

Indices and tables

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

p

- `proknow.CustomMetrics`, [9](#)
- `proknow.Exceptions`, [8](#)
- `proknow.Patients`, [19](#)
- `proknow.ProKnow`, [5](#)
- `proknow.Requestor`, [6](#)
- `proknow.Roles`, [14](#)
- `proknow.Users`, [16](#)
- `proknow.Workspaces`, [11](#)

Symbols

__init__() (proknow.CustomMetrics.CustomMetricItem method), 11
__init__() (proknow.CustomMetrics.CustomMetrics method), 9
__init__() (proknow.Exceptions.CustomMetricLookupError method), 9
__init__() (proknow.Exceptions.HttpError method), 8
__init__() (proknow.Exceptions.WorkspaceLookupError method), 8
__init__() (proknow.Patients.DoseItem method), 24
__init__() (proknow.Patients.EntityItem method), 23
__init__() (proknow.Patients.EntitySummary method), 23
__init__() (proknow.Patients.PatientItem method), 21
__init__() (proknow.Patients.PatientSummary method), 21
__init__() (proknow.Patients.Patients method), 19
__init__() (proknow.Patients.PlanItem method), 24
__init__() (proknow.Patients.StudySummary method), 22
__init__() (proknow.ProKnow.ProKnow method), 5
__init__() (proknow.Requestor.Requestor method), 6
__init__() (proknow.Roles.RoleItem method), 16
__init__() (proknow.Roles.RoleSummary method), 15
__init__() (proknow.Roles.Roles method), 14
__init__() (proknow.Users.UserItem method), 18
__init__() (proknow.Users.UserSummary method), 18
__init__() (proknow.Users.Users method), 16
__init__() (proknow.Workspaces.WorkspaceItem method), 14
__init__() (proknow.Workspaces.Workspaces method), 11

A

active (proknow.Users.UserItem attribute), 18

B

birth_date (proknow.Patients.PatientItem attribute), 21
birth_date (proknow.Patients.PatientSummary attribute), 20

birth_time (proknow.Patients.PatientItem attribute), 21
birth_time (proknow.Patients.PatientSummary attribute), 21
body (proknow.Exceptions.HttpError attribute), 8

C

context (proknow.CustomMetrics.CustomMetricItem attribute), 11
create() (proknow.CustomMetrics.CustomMetrics method), 9
create() (proknow.Patients.Patients method), 19
create() (proknow.Roles.Roles method), 14
create() (proknow.Users.Users method), 17
create() (proknow.Workspaces.Workspaces method), 12
custom_metrics (proknow.ProKnow.ProKnow attribute), 5
CustomMetricItem (class in proknow.CustomMetrics), 11
CustomMetricLookupError, 8
CustomMetrics (class in proknow.CustomMetrics), 9

D

data (proknow.CustomMetrics.CustomMetricItem attribute), 11
data (proknow.Patients.DoseItem attribute), 24
data (proknow.Patients.EntityItem attribute), 23
data (proknow.Patients.EntitySummary attribute), 23
data (proknow.Patients.PatientItem attribute), 21
data (proknow.Patients.PatientSummary attribute), 21
data (proknow.Patients.PlanItem attribute), 24
data (proknow.Patients.StudySummary attribute), 22
data (proknow.Roles.RoleItem attribute), 16
data (proknow.Roles.RoleSummary attribute), 15
data (proknow.Users.UserItem attribute), 18
data (proknow.Users.UserSummary attribute), 18
data (proknow.Users.Users attribute), 13
delete() (proknow.CustomMetrics.CustomMetricItem method), 11
delete() (proknow.CustomMetrics.CustomMetrics method), 9

delete() (proknow.Patients.PatientItem method), 22
delete() (proknow.Patients.Patients method), 19
delete() (proknow.Requestor.Requestor method), 6
delete() (proknow.Roles.RoleItem method), 16
delete() (proknow.Roles.Roles method), 15
delete() (proknow.Users.UserItem method), 18
delete() (proknow.Users.Users method), 17
delete() (proknow.Workspaces.WorkspaceItem method), 14
delete() (proknow.Workspaces.Workspaces method), 12
DoseItem (class in proknow.Patients), 24
download() (proknow.Patients.DoseItem method), 24
download() (proknow.Patients.PlanItem method), 24

E

email (proknow.Users.UserItem attribute), 18
email (proknow.Users.UserSummary attribute), 17
EntityItem (class in proknow.Patients), 23
EntitySummary (class in proknow.Patients), 22

F

find() (proknow.CustomMetrics.CustomMetrics method), 9
find() (proknow.Patients.Patients method), 19
find() (proknow.Roles.Roles method), 15
find() (proknow.Users.Users method), 17
find() (proknow.Workspaces.Workspaces method), 12
find_entities() (proknow.Patients.PatientItem method), 22

G

get() (proknow.Patients.EntitySummary method), 23
get() (proknow.Patients.Patients method), 20
get() (proknow.Patients.PatientSummary method), 21
get() (proknow.Requestor.Requestor method), 6
get() (proknow.Roles.Roles method), 15
get() (proknow.Roles.RoleSummary method), 15
get() (proknow.Users.Users method), 17
get() (proknow.Users.UserSummary method), 18
get_metadata() (proknow.Patients.PatientItem method), 22

H

HttpError, 8

I

id (proknow.CustomMetrics.CustomMetricItem attribute), 11
id (proknow.Patients.DoseItem attribute), 24
id (proknow.Patients.EntityItem attribute), 23
id (proknow.Patients.EntitySummary attribute), 23
id (proknow.Patients.PatientItem attribute), 21
id (proknow.Patients.PatientSummary attribute), 20
id (proknow.Patients.PlanItem attribute), 23

id (proknow.Patients.StudySummary attribute), 22
id (proknow.Roles.RoleItem attribute), 16
id (proknow.Roles.RoleSummary attribute), 15
id (proknow.Users.UserItem attribute), 18
id (proknow.Users.UserSummary attribute), 17
id (proknow.Workspaces.WorkspaceItem attribute), 13
ImageSetItem (class in proknow.Patients), 23

L

lookup() (proknow.Patients.Patients method), 20

M

message (proknow.Exceptions.CustomMetricLookupError attribute), 8
message (proknow.Exceptions.WorkspaceLookupError attribute), 8
metadata (proknow.Patients.PatientItem attribute), 21
mrn (proknow.Patients.PatientItem attribute), 21
mrn (proknow.Patients.PatientSummary attribute), 20

N

name (proknow.CustomMetrics.CustomMetricItem attribute), 11
name (proknow.Patients.PatientItem attribute), 21
name (proknow.Patients.PatientSummary attribute), 20
name (proknow.Roles.RoleItem attribute), 16
name (proknow.Roles.RoleSummary attribute), 15
name (proknow.Users.UserItem attribute), 18
name (proknow.Users.UserSummary attribute), 18
name (proknow.Workspaces.WorkspaceItem attribute), 13

P

patch() (proknow.Requestor.Requestor method), 7
PatientItem (class in proknow.Patients), 21
Patients (class in proknow.Patients), 19
patients (proknow.ProKnow.ProKnow attribute), 5
PatientSummary (class in proknow.Patients), 20
permissions (proknow.Roles.RoleItem attribute), 16
PlanItem (class in proknow.Patients), 23
post() (proknow.Requestor.Requestor method), 7
ProKnow (class in proknow.ProKnow), 5
proknow.CustomMetrics (module), 9
proknow.Exceptions (module), 8
proknow.Patients (module), 19
proknow.ProKnow (module), 5
proknow.Requestor (module), 6
proknow.Roles (module), 14
proknow.Users (module), 16
proknow.Workspaces (module), 11
ProKnowError, 8
protected (proknow.Workspaces.WorkspaceItem attribute), 14
put() (proknow.Requestor.Requestor method), 7

Python Enhancement Proposals

PEP 8, 25

Q

query() (proknow.CustomMetrics.CustomMetrics method), 10

query() (proknow.Patients.Patients method), 20

query() (proknow.Roles.Roles method), 15

query() (proknow.Users.Users method), 17

query() (proknow.Workspaces.Workspaces method), 12

R

Requestor (class in proknow.Requestor), 6

requestor (proknow.ProKnow.ProKnow attribute), 5

resolve() (proknow.CustomMetrics.CustomMetrics method), 10

resolve() (proknow.Workspaces.Workspaces method), 13

resolveById() (proknow.CustomMetrics.CustomMetrics method), 10

resolveById() (proknow.Workspaces.Workspaces method), 13

resolveByName() (proknow.CustomMetrics.CustomMetrics method), 10

resolveByName() (proknow.Workspaces.Workspaces method), 13

role_id (proknow.Users.UserItem attribute), 18

RoleItem (class in proknow.Roles), 16

Roles (class in proknow.Roles), 14

roles (proknow.ProKnow.ProKnow attribute), 5

RoleSummary (class in proknow.Roles), 15

S

save() (proknow.CustomMetrics.CustomMetricItem method), 11

save() (proknow.Patients.PatientItem method), 22

save() (proknow.Roles.RoleItem method), 16

save() (proknow.Users.UserItem method), 18

save() (proknow.Workspaces.WorkspaceItem method), 14

set_metadata() (proknow.Patients.PatientItem method), 22

sex (proknow.Patients.PatientItem attribute), 21

sex (proknow.Patients.PatientSummary attribute), 21

slug (proknow.Workspaces.WorkspaceItem attribute), 13

status_code (proknow.Exceptions.HttpError attribute), 8

stream() (proknow.Requestor.Requestor method), 8

StructureSetItem (class in proknow.Patients), 23

StudySummary (class in proknow.Patients), 22

T

type (proknow.CustomMetrics.CustomMetricItem attribute), 11

U

UserItem (class in proknow.Users), 18

Users (class in proknow.Users), 16

users (proknow.ProKnow.ProKnow attribute), 5

UserSummary (class in proknow.Users), 17

W

WorkspaceItem (class in proknow.Workspaces), 13

WorkspaceLookupError, 8

Workspaces (class in proknow.Workspaces), 11

workspaces (proknow.ProKnow.ProKnow attribute), 5