
azureenergylabelerlib Documentation

Release 3.2.1

Sayantana Khanra

Jun 07, 2023

CONTENTS

1	azureenergylabelerlib	3
1.1	Development Workflow	3
1.2	Important Information	4
1.3	Project Features	4
2	Installation	5
3	Usage	7
4	Contributing	9
4.1	Submit Feedback	9
5	azureenergylabelerlib	11
5.1	azureenergylabelerlib package	11
6	Credits	25
6.1	Development Lead	25
6.2	Contributors	25
7	History	27
8	0.0.1 (22-04-2022)	29
9	0.1.0 (22-06-2022)	31
10	0.2.0 (23-06-2022)	33
11	0.2.1 (23-06-2022)	35
12	1.0.0 (15-09-2022)	37
13	1.1.0 (21-09-2022)	39
14	1.1.1 (22-09-2022)	41
15	2.0.0 (04-10-2022)	43
16	3.0.0 (18-10-2022)	45
17	3.1.0 (07-03-2023)	47
18	3.1.1 (21-03-2023)	49

19	3.2.0 (11-05-2023)	51
20	3.2.1 (07-06-2023)	53
21	Indices and tables	55
	Python Module Index	57
	Index	59

Contents:

AZUREENERGYLABELERLIB

A python library that generates energy labels based on findings in Azure subscriptions

- Documentation: <https://azureenergylabelerlib.readthedocs.org/en/latest>

1.1 Development Workflow

The workflow supports the following steps

- lint
- test
- build
- document
- upload
- graph

These actions are supported out of the box by the corresponding scripts under `_CI/scripts` directory with sane defaults based on best practices. Sourcing `setup_aliases.ps1` for windows powershell or `setup_aliases.sh` in bash on Mac or Linux will provide with handy aliases for the shell of all those commands prepended with an underscore.

The bootstrap script creates a `.venv` directory inside the project directory hosting the virtual environment. It uses `pipenv` for that. It is called by all other scripts before they do anything. So one could simple start by calling `_lint` and that would set up everything before it tried to actually lint the project

Once the code is ready to be delivered the `_tag` script should be called accepting one of three arguments, `patch`, `minor`, `major` following the semantic versioning scheme. So for the initial delivery one would call

```
$ _tag -minor
```

which would bump the version of the project to 0.1.0 tag it in git and do a push and also ask for the change and automatically update `HISTORY.rst` with the version and the change provided.

So the full workflow after git is initialized is:

- repeat as necessary (of course it could be `test - code - lint` :))
 - code
 - lint
 - test
- commit and push
- develop more through the code-lint-test cycle

- tag (with the appropriate argument)
- build
- upload (if you want to host your package in pypi)
- document (of course this could be run at any point)

1.2 Important Information

This template is based on pipenv. In order to be compatible with requirements.txt so the actual created package can be used by any part of the existing python ecosystem some hacks were needed. So when building a package out of this **do not** simple call

```
$ python setup.py sdist bdist_egg
```

as this will produce an unusable artifact with files missing. Instead use the provided build and upload scripts that create all the necessary files in the artifact.

1.3 Project Features

- TODO

INSTALLATION

At the command line:

```
$ pip install azureenergylabelerlib
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv azureenergylabelerlib  
$ pip install azureenergylabelerlib
```

Or, if you are using pipenv:

```
$ pipenv install azureenergylabelerlib
```


USAGE

To develop on azureenergylabelerlib:

```
# The following commands require pipenv as a dependency

# To lint the project
_CI/scripts/lint.py

# To execute the testing
_CI/scripts/test.py

# To create a graph of the package and dependency tree
_CI/scripts/graph.py

# To build a package of the project under the directory "dist/"
_CI/scripts/build.py

# To see the package version
_CI/scripts/tag.py

# To bump semantic versioning [--major|--minor|--patch]
_CI/scripts/tag.py --major|--minor|--patch

# To upload the project to a pypi repo if user and password are properly provided
_CI/scripts/upload.py

# To build the documentation of the project
_CI/scripts/document.py
```

To use azureenergylabelerlib in a project:

```
from azureenergylabelerlib import Azureenergylabelerlib
azureenergylabelerlib = Azureenergylabelerlib()
```


CONTRIBUTING

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

4.1 Submit Feedback

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.

4.1.1 Get Started!

Ready to contribute? Here's how to set up *azureenergylabelerlib* for local development. Using of pipenv is highly recommended.

1. Clone your fork locally:

```
$ git clone https://github.com/schubergphilis/azureenergylabelerlib
```

2. Install your local copy into a virtualenv. Assuming you have pipenv installed, this is how you set up your clone for local development:

```
$ cd azureenergylabelerlib/  
$ pipenv install --ignore-pipfile
```

3. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally. Do your development while using the CI capabilities and making sure the code passes lint, test, build and document stages.

4. Commit your changes and push your branch to the server:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

5. Submit a merge request

AZUREENERGYLABELERLIB

5.1 azureenergylabellerlib package

5.1.1 Submodules

5.1.2 azureenergylabellerlib.azureenergylabellerlib module

Main code for azureenergylabellerlib.

```
class azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler(tenant_id,
                                                                    frameworks={'Azure CIS
1.1.0', 'Microsoft cloud
security benchmark'}, tenant_thresholds=[{'label':
'A', 'percentage': 90},
{'label': 'B', 'percentage':
70}, {'label': 'C',
'percentage': 50}, {'label':
'D', 'percentage': 30},
{'label': 'E', 'percentage':
20}], re-
source_group_thresholds=[{'label':
'A', 'high': 0, 'medium':
10, 'low': 20,
'days_open_less_than':
999}, {'label': 'B', 'high':
10, 'medium': 20, 'low':
40,
'days_open_less_than':
999}, {'label': 'C', 'high':
15, 'medium': 30, 'low':
60,
'days_open_less_than':
999}, {'label': 'D', 'high':
20, 'medium': 40, 'low':
80,
'days_open_less_than':
999}, {'label': 'E', 'high':
25, 'medium': 50, 'low':
100,
'days_open_less_than':
999}], subscription_thresholds=[{'label':
'A', 'high': 0, 'medium':
10, 'low': 20,
'days_open_less_than':
999}, {'label': 'B', 'high':
10, 'medium': 20, 'low':
40,
'days_open_less_than':
999}, {'label': 'C', 'high':
15, 'medium': 30, 'low':
60,
'days_open_less_than':
999}, {'label': 'D', 'high':
20, 'medium': 40, 'low':
80,
'days_open_less_than':
999}, {'label': 'E', 'high':
25, 'medium': 50, 'low':
100,
'days_open_less_than':
999}], credentials=None,
al-
lowed_subscription_ids=None,
de-
nied_subscription_ids=None)
```

Bases: object

Labeling subscriptions based on findings and label configurations.

Parameters

- **tenant_id** (*str*) – Azure Tenant ID to collect energy label, for example: *18d9dec0-d762-11ec-9cb5-00155da09878*.
- **frameworks** (*set[str]*) – Frameworks taken into account when generating the energy label. Defaults to `DEFAULT_DEFENDER_FOR_CLOUD_FRAMEWORKS`
- **tenant_thresholds** (*list[dict[str, Any]]*) – Defines percentage thresholds mapping to energy labels for the tenant. Defaults to `TENANT_THRESHOLDS`
- **resource_group_thresholds** (*list[dict[str, Any]]*) – Defines percentage thresholds mapping to energy labels for resource groups. Defaults to `RESOURCE_GROUP_THRESHOLDS`
- **subscription_thresholds** (*list[dict[str, Any]]*) – Defines percentage thresholds mapping to energy labels for resource groups. Defaults to `SUBSCRIPTION_THRESHOLDS`
- **credentials** (*Any*) – One of `identity` Credential object containing the credentials used to access the Azure API. If not supplied, the library will create a `DefaultAzureCredential` and attempt to authenticate in the following order:
 1. A service principal configured by environment variables. See `EnvironmentCredential` for more details.
 2. An Azure managed identity. See `ManagedIdentityCredential` for more details.
 3. **On Windows only: a user who has signed in with a Microsoft application, such as Visual Studio. If multiple** identities are in the cache, then the value of the environment variable `AZURE_USERNAME` is used to select which identity to use. See `SharedTokenCacheCredential` for more details.
 4. The user currently signed in to Visual Studio Code.
 5. The identity currently logged in to the Azure CLI.
 6. The identity currently logged in to Azure PowerShell.
- **allowed_subscription_ids** (*Any*) – Inclusion list of subscriptions to be evaluated
- **denied_subscription_ids** (*Any*) – Exclude list of subscriptions to be evaluated

property defender_for_cloud

Defender for cloud.

property defender_for_cloud_findings

Defender for cloud findings.

property filtered_defender_for_cloud_findings

Filtered defender for cloud findings.

property labeled_subscriptions_energy_label

Energy label of the labeled subscriptions.

property matching_frameworks

The frameworks provided to match the findings of.

property tenant

Tenant.

property tenant_energy_label

Energy label of the Azure Tenant.

property tenant_labeled_subscriptions

The tenant labeled subscription objects.

5.1.3 azureenergylabelerlib.azureenergylabelerlibexceptions module

Custom exception code for azureenergylabelerlib.

exception azureenergylabelerlib.azureenergylabelerlibexceptions.InvalidCredentials

Bases: Exception

Credentials provided are not valid.

exception azureenergylabelerlib.azureenergylabelerlibexceptions.InvalidFrameworks

Bases: Exception

The frameworks provided are not valid.

exception azureenergylabelerlib.azureenergylabelerlibexceptions.InvalidPath

Bases: Exception

The path provided is not valid.

exception

azureenergylabelerlib.azureenergylabelerlibexceptions.InvalidSubscriptionListProvided

Bases: Exception

The list of subscriptions provided are not valid Azure subscriptions.

exception

azureenergylabelerlib.azureenergylabelerlibexceptions.MutuallyExclusiveArguments

Bases: Exception

The arguments provided are mutually exclusive and only one of the should be provided.

exception

azureenergylabelerlib.azureenergylabelerlibexceptions.SubscriptionNotPartOfTenant

Bases: Exception

If subscription ids are provided but are not part of the tenant.

5.1.4 azureenergylabelerlib.configuration module

configuration package.

Import all parts from configuration here .. _Google Python Style Guide:

<https://google.github.io/styleguide/pyguide.html>

5.1.5 azureenergylabelerlib.datamodels module

Main code for datamodels.

```
class azureenergylabelerlib.datamodels.DefenderForCloudFindingsData(filename, de-  
fender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.LabeledResourceGroupData(filename,  
labeled_resource_group_data,  
defender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property data

Data of an subscription to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.LabeledResourceGroupsData(filename,  
labeled_subscriptions,  
defender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.LabeledSubscriptionData(filename, labeled_subscription,  
defender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property data

Data of an subscription to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.LabeledSubscriptionsData(filename, labeled_subscriptions,  
defender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.SubscriptionExemptedPolicies(filename,  
                                                                    labeled_subscriptions)
```

Bases: object

Models the data for exempted policies to export.

property data

Data of an subscription exempted policies to export.

property json

Data to json.

```
class azureenergylabelerlib.datamodels.TenantEnergyLabelingData(filename, id, energy_label,  
                                                                    labeled_subscriptions,  
                                                                    defender_for_cloud_findings)
```

Bases: object

Models the data for energy labeling to export.

property json

Data to json.

5.1.6 azureenergylabelerlib.entities module

Main code for entities.

```
class azureenergylabelerlib.entities.DataExporter(export_types, id, energy_label,  
                                                                    defender_for_cloud_findings,  
                                                                    labeled_subscriptions, credentials=None)
```

Bases: object

Export Azure security data.

export(*path*)

Exports the data to the provided path.

```
class azureenergylabelerlib.entities.DataFileFactory(export_type, id, energy_label,  
                                                                    defender_for_cloud_findings,  
                                                                    labeled_subscriptions)
```

Bases: object

Data export factory to handle the different data types returned.

```
class azureenergylabelerlib.entities.DefenderForCloud(credential, subscription_list)
```

Bases: object

Models the Defender for Cloud and retrieves findings.

frameworks = {'Azure CIS 1.1.0', 'Microsoft cloud security benchmark'}

get_findings(*frameworks*)

Filters provided findings by the provided frameworks.

Parameters

frameworks – The frameworks to filter for

Returns

A list of findings matching the provided frameworks

Return type

findings (list(Findings))

static validate_frameworks(*frameworks*)

Validates provided frameworks.

Parameters

frameworks – One or more of the frameworks to validate according to an accepted list.

Returns

True if frameworks are valid False otherwise.

class azureenergylabelerlib.entities.**EnergyLabeler**(*object_type, name, findings, threshold*)

Bases: object

Generic EnergyLabel factory to return energy label for resource groups and subscriptions.

property energy_label

Energy Label for the subscription or resource group.

class azureenergylabelerlib.entities.**Finding**(*data*)

Bases: object

Models a finding.

property azure_portal_recommendation_link

Azure portal recommendation link Steps.

property compliance_control_id

Compliance control id.

property compliance_standard_id

Compliance standard id.

property compliance_state

Compliance state.

property control_name

Control Name.

property days_open

Days open.

property description

Finding Description.

property first_evaluation_date

First Evaluation Date.

property is_skipped

The finding is skipped or not.

property not_applicable_reason

Control Name.

property recommendation_display_name

Recommendation Display Name.

property recommendation_id

Recommendation Id.

property recommendation_name

Recommendation Name.

property remediation_steps

Remediation Steps.

property resource_group

Resource group name.

property resource_id

Resource name.

property resource_name

Resource name.

property resource_type

Resource type.

property severity

Severity.

property state

Title.

property status_change_date

Status Change Date.

property subscription_id

Subscription id.

class azureenergylabelerlib.entities.FindingParserLabeler

Bases: object

static exclude_findings_by_state(*findings, states*)

Returns findings excluding those with specific states.

static get_not_skipped_findings(*findings*)

Not skipped findings for the subscription.

class azureenergylabelerlib.entities.ResourceGroup(*data*)

Bases: *FindingParserLabeler*

Models the Azure subscription's resource group that can label itself.

get_energy_label(*findings, states=('notapplicable', 'healthy')*)

Calculates the energy label for the resource group.

Parameters

- **findings** – Either a list of defender for cloud findings.
- **states** – The states to filter findings out for.

Returns

The energy label of the resource group based on the provided configuration.

get_open_findings(*findings*)

Findings for the resource group.

property location

location.

property name

name.

class azureenergylabelerlib.entities.**Subscription**(*credential, data*)

Bases: [*FindingParserLabeler*](#)

Models the Azure subscription that can label itself.

property display_name

display_name.

property exempted_policies

Policies exempted for this subscription.

get_energy_label(*findings, states=('notapplicable', 'healthy')*)

Calculates the energy label for the Subscription.

Parameters

- **findings** – Either a list of defender for cloud findings.
- **states** – The states to filter findings out for.

Returns

The energy label of the resource group based on the provided configuration.

get_open_findings(*findings*)

Findings for the resource group.

property resource_groups

Resource groups of this subscription.

property state

State of the subscription.

property subscription_id

Subscription id.

property tenant_id

Tenant id.

```
class azureenergylabelerlib.entities.Tenant(credential, tenant_id, thresholds=[{'label': 'A',
                                     'percentage': 90}, {'label': 'B', 'percentage': 70}, {'label':
                                     'C', 'percentage': 50}, {'label': 'D', 'percentage': 30},
                                     {'label': 'E', 'percentage': 20}],
                                     subscription_thresholds=[{'label': 'A', 'high': 0, 'medium':
                                     10, 'low': 20, 'days_open_less_than': 999}, {'label': 'B',
                                     'high': 10, 'medium': 20, 'low': 40, 'days_open_less_than':
                                     999}, {'label': 'C', 'high': 15, 'medium': 30, 'low': 60,
                                     'days_open_less_than': 999}, {'label': 'D', 'high': 20,
                                     'medium': 40, 'low': 80, 'days_open_less_than': 999},
                                     {'label': 'E', 'high': 25, 'medium': 50, 'low': 100,
                                     'days_open_less_than': 999}],
                                     resource_group_thresholds=[{'label': 'A', 'high': 0,
                                     'medium': 10, 'low': 20, 'days_open_less_than': 999},
                                     {'label': 'B', 'high': 10, 'medium': 20, 'low': 40,
                                     'days_open_less_than': 999}, {'label': 'C', 'high': 15,
                                     'medium': 30, 'low': 60, 'days_open_less_than': 999},
                                     {'label': 'D', 'high': 20, 'medium': 40, 'low': 80,
                                     'days_open_less_than': 999}, {'label': 'E', 'high': 25,
                                     'medium': 50, 'low': 100, 'days_open_less_than': 999}],
                                     allowed_subscription_ids=None,
                                     denied_subscription_ids=None)
```

Bases: object

Models the Azure tenant and retrieves subscriptions from it.

get_allowed_subscriptions()

Retrieves allowed subscriptions based on an allow list.

Returns

The list of subscriptions based on the allowed list.

get_energy_label(defender_for_cloud_findings)

Calculates and returns the energy label of the Tenant.

Parameters

defender_for_cloud_findings – The measurement data of all the findings for a tenant.

Returns

The labeling object of the Tenant.

Return type

energy_label (*TenantEnergyLabel*)

get_energy_label_of_targeted_subscriptions(defender_for_cloud_findings)

Get the energy label of the targeted subscriptions.

Parameters

defender_for_cloud_findings – The findings from defender for cloud.

Returns

The energy label of the targeted subscriptions.

Return type

energy_label (str)

get_labeled_targeted_subscriptions(defender_for_cloud_findings)

Labels the subscriptions based on the allow and deny list provided.

Parameters

defender_for_cloud_findings – The findings for a Tenant.

Returns

A list of Azure Subscriptions objects that have their labels calculated.

Return type

labeled_subscriptions (list)

get_not_denied_subscriptions()

Retrieves denied subscriptions based on an denied list.

Returns

The list of subscriptions based on the denied list.

property subscriptions

Subscriptions of the Tenant.

Returns

List of subscriptions retrieved

property subscriptions_to_be_labeled

Subscriptions to be labeled according to the allow or deny list arguments.

Returns

A list of subscriptions to be labeled.

Return type

subscription (list)

5.1.7 azureenergylabelerlib.labels module

Main code for labels.

```
class azureenergylabelerlib.labels.AggregateEnergyLabel(label: str, best_label: str, worst_label: str)
```

Bases: object

Models an energy label averaging multiple subscription labels.

best_label: str

label: str

worst_label: str

```
class azureenergylabelerlib.labels.AggregateSubscriptionEnergyLabel(label: str, best_label: str,
                                                                    worst_label: str,
                                                                    subscriptions_measured:
                                                                    str)
```

Bases: [*AggregateEnergyLabel*](#)

Models the landing zone energy label.

subscriptions_measured: str

```
class azureenergylabelerlib.labels.ResourceGroupEnergyLabel(label: str = 'F',
                                                            number_of_high_findings: int = 9999,
                                                            number_of_medium_findings: int =
                                                            9999, number_of_low_findings: int =
                                                            9999, max_days_open: int = 9999)
```

Bases: object

Models the resource group energy label.

label: str = 'F'

max_days_open: int = 9999

number_of_high_findings: int = 9999

number_of_low_findings: int = 9999

number_of_medium_findings: int = 9999

```
class azureenergylabelerlib.labels.SubscriptionEnergyLabel(label: str = 'F',  
                                                           number_of_high_findings: int = 9999,  
                                                           number_of_medium_findings: int =  
                                                           9999, number_of_low_findings: int =  
                                                           9999, max_days_open: int = 9999)
```

Bases: object

Models the subscription energy label.

label: str = 'F'

max_days_open: int = 9999

number_of_high_findings: int = 9999

number_of_low_findings: int = 9999

number_of_medium_findings: int = 9999

```
class azureenergylabelerlib.labels.TenantEnergyLabel(label: str, best_label: str, worst_label: str,  
                                                       coverage: str)
```

Bases: *AggregateEnergyLabel*

Models the landing zone energy label.

coverage: str

5.1.8 azureenergylabelerlib.schemas module

schemas package.

Import all parts from schemas here .. *_Google Python Style Guide*:

<https://google.github.io/styleguide/pyguide.html>

5.1.9 azureenergylabelerlib.validations module

Main code for validations.

class azureenergylabelerlib.validations.**DestinationPath**(*location*)

Bases: object

Models a destination path and identifies if it is valid and it's type.

is_valid()

Is the path valid.

property type

The type of the path.

azureenergylabelerlib.validations.**are_valid_subscription_ids**(*subscription_ids*)

Checks whether a provided list of subscription ids contains all valid Azure subscription ids.

Parameters

subscription_ids (*list*) – A list of subscription id strings.

Returns

True if the provided list contains all valid Azure subscription ids, false otherwise.

azureenergylabelerlib.validations.**is_valid_subscription_id**(*subscription_id*)

Checks whether a provided subscription_id is a valid Azure subscription id.

Parameters

subscription_id (*str*) – A subscription id string.

Returns

True if the provided value is a valid Azure subscription id, false otherwise.

azureenergylabelerlib.validations.**validate_allowed_denied_subscription_ids**(*allowed_subscription_ids=None, denied_subscription_ids=None*)

Validates provided allow and deny subscription id lists.

Not both arguments can contain values as they are logically mutually exclusive. The validations process also validates that the arguments contain valid subscription id values if provided.

Parameters

- **allowed_subscription_ids** (*str/iterable*) – A single or multiple subscription id to validate, mutually exclusive with the deny list
- **denied_subscription_ids** (*str/iterable*) – A single or multiple subscription id to validate, mutually exclusive with the allow list

Returns

A tuple of list values with valid subscription ids

Return type

allowed_subscription_ids, denied_subscription_ids

Raises

- **MutuallyExclusiveArguments** – If both arguments contain values.
- **InvalidSubscriptionListProvided** – If any of the provided ids in the list is not a valid subscription id.

`azureenergylabelerlib.validations.validate_subscription_ids(subscription_ids)`

Validates a provided string or iterable that it contains valid Azure subscription ids.

Parameters

subscription_ids – A string or iterable of strings with Azure subscription ids.

Returns

A list of valid Azure subscription ids.

Return type

subscription_ids (list)

Raises

InvalidSubscriptionIdProvided – If any of the provided Subscription ids is not a valid Azure subscription id.

5.1.10 Module contents

azureenergylabelerlib package.

Import all parts from azureenergylabelerlib here

CREDITS

6.1 Development Lead

- Sayantan Khanra <skhanra@schubergphilis.com>

6.2 Contributors

None yet. Why not be the first?

CHAPTER
SEVEN

HISTORY

0.0.1 (22-04-2022)

- First code creation

0.1.0 (22-06-2022)

- First release

0.2.0 (23-06-2022)

- First Release

0.2.1 (23-06-2022)

- Changed export all parameter

1.0.0 (15-09-2022)

- – Removed pandas dependency in favor of native python functionality
- – Added support for SAS URLs to export results to a Storage Account
- – Fixed a bug where open days would show as 9999 for subscriptions scoring an A
- – Fixed a typo on the exempted findings json file

1.1.0 (21-09-2022)

- Added more information to the `-export-metrics` option output

1.1.1 (22-09-2022)

- Fixed a bug where Resource Groups lack the `exempted_findings` property

2.0.0 (04-10-2022)

- Removed ExemptedPolicy class

3.0.0 (18-10-2022)

- Microsoft renamed “Azure Security Benchmark” to “Microsoft cloud security benchmark”, changing the interface

3.1.0 (07-03-2023)

- Bump dependencies.

3.1.1 (21-03-2023)

- Check subscription tenant id on Tenant init

3.2.0 (11-05-2023)

- Improved how findings are filtered

3.2.1 (07-06-2023)

- Fixed pagination
- Fixed typos

INDICES AND TABLES

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

PYTHON MODULE INDEX

a

- `azureenergylabelerlib`, [24](#)
- `azureenergylabelerlib.azureenergylabelerlib`,
[11](#)
- `azureenergylabelerlib.azureenergylabelerlibexceptions`,
[14](#)
- `azureenergylabelerlib.configuration`, [14](#)
- `azureenergylabelerlib.datamodels`, [15](#)
- `azureenergylabelerlib.entities`, [16](#)
- `azureenergylabelerlib.labels`, [21](#)
- `azureenergylabelerlib.schemas`, [22](#)
- `azureenergylabelerlib.validations`, [23](#)

A

AggregateEnergyLabel (class in *azureenergylabelerlib.labels*), 21

AggregateSubscriptionEnergyLabel (class in *azureenergylabelerlib.labels*), 21

are_valid_subscription_ids() (in module *azureenergylabelerlib.validations*), 23

azure_portal_recommendation_link (*azureenergylabelerlib.entities.Finding* property), 17

AzureEnergyLabeler (class in *azureenergylabelerlib.azureenergylabelerlib*), 11

azureenergylabelerlib
module, 24

azureenergylabelerlib.azureenergylabelerlib
module, 11

azureenergylabelerlib.azureenergylabelerlibexceptions
module, 14

azureenergylabelerlib.configuration
module, 14

azureenergylabelerlib.datamodels
module, 15

azureenergylabelerlib.entities
module, 16

azureenergylabelerlib.labels
module, 21

azureenergylabelerlib.schemas
module, 22

azureenergylabelerlib.validations
module, 23

B

best_label (*azureenergylabelerlib.labels.AggregateEnergyLabel* attribute), 21

C

compliance_control_id (*azureenergylabelerlib.entities.Finding* property), 17

compliance_standard_id (*azureenergylabelerlib.entities.Finding* property), 17

compliance_state (*azureenergylabelerlib.entities.Finding* property), 17

control_name (*azureenergylabelerlib.entities.Finding* property), 17

coverage (*azureenergylabelerlib.labels.TenantEnergyLabel* attribute), 22

D

data (*azureenergylabelerlib.datamodels.LabeledResourceGroupData* property), 15

data (*azureenergylabelerlib.datamodels.LabeledSubscriptionData* property), 15

data (*azureenergylabelerlib.datamodels.SubscriptionExemptedPolicies* property), 16

DataExporter (class in *azureenergylabelerlib.entities*), 16

DataFileFactory (class in *azureenergylabelerlib.entities*), 16

days_open (*azureenergylabelerlib.entities.Finding* property), 17

defender_for_cloud (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 13

defender_for_cloud_findings (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 13

DefenderForCloud (class in *azureenergylabelerlib.entities*), 16

DefenderForCloudFindingsData (class in *azureenergylabelerlib.datamodels*), 15

description (*azureenergylabelerlib.entities.Finding* property), 17

DestinationPath (class in *azureenergylabelerlib.validations*), 23

display_name (*azureenergylabelerlib.entities.Subscription* property), 19

E

energy_label (*azureenergylabelerlib.entities.EnergyLabeler* property), 17

EnergyLabeler (class in *azureenergylabelerlib.entities*), 17

exclude_findings_by_state() (*azureenergylabelerlib.entities.FindingParserLabeler* static method), 18

exempted_policies (*azureenergylabelerlib.entities.Subscription* property), 19

export() (*azureenergylabelerlib.entities.DataExporter* method), 16

F

filtered_defender_for_cloud_findings (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 13

Finding (class in *azureenergylabelerlib.entities*), 17

FindingParserLabeler (class in *azureenergylabelerlib.entities*), 18

first_evaluation_date (*azureenergylabelerlib.entities.Finding* property), 17

frameworks (*azureenergylabelerlib.entities.DefenderForCloud* attribute), 16

G

get_allowed_subscriptions() (*azureenergylabelerlib.entities.Tenant* method), 20

get_energy_label() (*azureenergylabelerlib.entities.ResourceGroup* method), 18

get_energy_label() (*azureenergylabelerlib.entities.Subscription* method), 19

get_energy_label() (*azureenergylabelerlib.entities.Tenant* method), 20

get_energy_label_of_targeted_subscriptions() (*azureenergylabelerlib.entities.Tenant* method), 20

get_findings() (*azureenergylabelerlib.entities.DefenderForCloud* method), 16

get_labeled_targeted_subscriptions() (*azureenergylabelerlib.entities.Tenant* method), 20

get_not_denied_subscriptions() (*azureenergylabelerlib.entities.Tenant* method), 21

get_not_skipped_findings() (*azureenergylabelerlib.entities.FindingParserLabeler* static method), 18

get_open_findings() (*azureenergylabelerlib.entities.ResourceGroup* method), 18

get_open_findings() (*azureenergylabelerlib.entities.Subscription* method), 19

I

InvalidCredentials, 14

InvalidFrameworks, 14

InvalidPath, 14

InvalidSubscriptionListProvided, 14

is_skipped (*azureenergylabelerlib.entities.Finding* property), 17

is_valid() (*azureenergylabelerlib.validations.DestinationPath* method), 23

is_valid_subscription_id() (in module *azureenergylabelerlib.validations*), 23

J

json (*azureenergylabelerlib.datamodels.DefenderForCloudFindingsData* property), 15

json (*azureenergylabelerlib.datamodels.LabeledResourceGroupData* property), 15

json (*azureenergylabelerlib.datamodels.LabeledResourceGroupsData* property), 15

json (*azureenergylabelerlib.datamodels.LabeledSubscriptionData* property), 15

json (*azureenergylabelerlib.datamodels.LabeledSubscriptionsData* property), 15

json (*azureenergylabelerlib.datamodels.SubscriptionExemptedPolicies* property), 16

json (*azureenergylabelerlib.datamodels.TenantEnergyLabelingData* property), 16

L

label (*azureenergylabelerlib.labels.AggregateEnergyLabel* attribute), 21

label (*azureenergylabelerlib.labels.ResourceGroupEnergyLabel* attribute), 22

label (*azureenergylabelerlib.labels.SubscriptionEnergyLabel* attribute), 22

labeled_subscriptions_energy_label (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 13

LabeledResourceGroupData (class in *azureenergylabelerlib.datamodels*), 15

LabeledResourceGroupsData (class in *azureenergylabelerlib.datamodels*), 15

LabeledSubscriptionData (class in *azureenergylabelerlib.datamodels*), 15

LabeledSubscriptionsData (class in *azureenergylabelerlib.datamodels*), 15

location (azureenergylabelerlib.entities.ResourceGroup property), 18

M

matching_frameworks (azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler property), 13

max_days_open (azureenergylabelerlib.labels.ResourceGroupEnergyLabel attribute), 22

max_days_open (azureenergylabelerlib.labels.SubscriptionEnergyLabel attribute), 22

module

azureenergylabelerlib, 24

azureenergylabelerlib.azureenergylabelerlib, 11

azureenergylabelerlib.azureenergylabelerlib.exceptions, 14

azureenergylabelerlib.configuration, 14

azureenergylabelerlib.datamodels, 15

azureenergylabelerlib.entities, 16

azureenergylabelerlib.labels, 21

azureenergylabelerlib.schemas, 22

azureenergylabelerlib.validations, 23

MutuallyExclusiveArguments, 14

N

name (azureenergylabelerlib.entities.ResourceGroup property), 19

not_applicable_reason (azureenergylabelerlib.entities.Finding property), 17

number_of_high_findings (azureenergylabelerlib.labels.ResourceGroupEnergyLabel attribute), 22

number_of_high_findings (azureenergylabelerlib.labels.SubscriptionEnergyLabel attribute), 22

number_of_low_findings (azureenergylabelerlib.labels.ResourceGroupEnergyLabel attribute), 22

number_of_low_findings (azureenergylabelerlib.labels.SubscriptionEnergyLabel attribute), 22

number_of_medium_findings (azureenergylabelerlib.labels.ResourceGroupEnergyLabel attribute), 22

number_of_medium_findings (azureenergylabelerlib.labels.SubscriptionEnergyLabel attribute), 22

R

recommendation_display_name (azureenergylabelerlib.entities.Finding property), 17

recommendation_id (azureenergylabelerlib.entities.Finding property), 17

recommendation_name (azureenergylabelerlib.entities.Finding property), 17

remediation_steps (azureenergylabelerlib.entities.Finding property), 18

resource_group (azureenergylabelerlib.entities.Finding property), 18

resource_groups (azureenergylabelerlib.entities.Subscription property), 19

resource_id (azureenergylabelerlib.entities.Finding property), 18

resource_name (azureenergylabelerlib.entities.Finding property), 18

resource_type (azureenergylabelerlib.entities.Finding property), 18

ResourceGroup (class in azureenergylabelerlib.entities), 18

ResourceGroupEnergyLabel (class in azureenergylabelerlib.labels), 21

S

severity (azureenergylabelerlib.entities.Finding property), 18

state (azureenergylabelerlib.entities.Finding property), 18

state (azureenergylabelerlib.entities.Subscription property), 19

status_change_date (azureenergylabelerlib.entities.Finding property), 18

Subscription (class in azureenergylabelerlib.entities), 19

subscription_id (azureenergylabelerlib.entities.Finding property), 18

subscription_id (azureenergylabelerlib.entities.Subscription property), 19

SubscriptionEnergyLabel (class in azureenergylabelerlib.labels), 22

SubscriptionExemptedPolicies (class in azureenergylabelerlib.datamodels), 15

SubscriptionNotPartOfTenant, 14

subscriptions (azureenergylabelerlib.entities.Tenant property), 21

subscriptions_measured (azureenergylabelerlib.labels.AggregateSubscriptionEnergyLabel attribute), 21

subscriptions_to_be_labeled (azureenergylabelerlib.entities.Tenant property), 21

T

tenant (azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler property), 13

Tenant (class in azureenergylabelerlib.entities), 19

`tenant_energy_label` (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 14

`tenant_id` (*azureenergylabelerlib.entities.Subscription* property), 19

`tenant_labeled_subscriptions` (*azureenergylabelerlib.azureenergylabelerlib.AzureEnergyLabeler* property), 14

`TenantEnergyLabel` (class in *azureenergylabelerlib.labels*), 22

`TenantEnergyLabelingData` (class in *azureenergylabelerlib.datamodels*), 16

`type` (*azureenergylabelerlib.validations.DestinationPath* property), 23

V

`validate_allowed_denied_subscription_ids()` (in module *azureenergylabelerlib.validations*), 23

`validate_frameworks()` (*azureenergylabelerlib.entities.DefenderForCloud* static method), 17

`validate_subscription_ids()` (in module *azureenergylabelerlib.validations*), 23

W

`worst_label` (*azureenergylabelerlib.labels.AggregateEnergyLabel* attribute), 21