
Azure SDK for Python Documentation

Release 0.37.0

Microsoft

Oct 05, 2017

Contents

1	Installation:	1
2	Documentation:	3
3	Features:	5
4	System Requirements:	7
5	Need Help?:	9
6	Contributing:	11
6.1	Contribute Code or Provide Feedback:	11
7	Learn More	13
8	Indices and tables	15
8.1	Upgrade Guide	15
8.2	General Changes	15
8.3	Blob	16
8.4	Queue	16
8.5	Table	16
8.6	azure.storage.blob package	16
8.7	azure.storage.common package	17
8.8	azure.storage.file package	17
8.9	azure.storage.queue package	17
8.10	azure	17

CHAPTER 1

Installation:

Please refer to the Github page for the installation instructions: <https://github.com/Azure/azure-storage-python>

If you are looking for Azure Service Bus or the Azure management libraries, please visit <https://github.com/Azure/azure-sdk-for-python>

CHAPTER 2

Documentation:

- [Blob Getting Started Doc – \(API\)](#)
- [Queue Getting Started Doc – \(API\)](#)
- [File Getting Started Doc – \(API\)](#)
- [Reference Documentation - All Services](#)

CHAPTER 3

Features:

- Blob
 - Create/Read/Update/Delete Containers
 - Create/Read/Update/Delete Blobs
 - Advanced Blob Operations
- Queue
 - Create/Delete Queues
 - Insert/Peek Queue Messages
 - Advanced Queue Operations
- Files
 - Create/Update/Delete Shares
 - Create/Update/Delete Directories
 - Create/Read/Update/Delete Files
 - Advanced File Operations

CHAPTER 4

System Requirements:

The supported Python versions are 2.7.x, 3.3.x, 3.4.x, and 3.5.x. To download Python, please visit <https://www.python.org/download/>

We recommend Python Tools for Visual Studio as a development environment for developing your applications. Please visit <http://aka.ms/python> for more information.

CHAPTER 5

Need Help?:

Be sure to check out the [Microsoft Azure Developer Forums on Stack Overflow](#) if you have trouble with the provided code.

Contributing:

Contribute Code or Provide Feedback:

If you would like to become an active contributor to this project, please follow the instructions provided in [Microsoft Azure Projects Contribution Guidelines](#).

If you encounter any bugs with the library, please file an issue in the [Issues](#) section of the project.

CHAPTER 7

[Learn More](#)

Microsoft Azure Python Developer Center

- [genindex](#)
- [modindex](#)
- [search](#)

Upgrade Guide

This guide is intended to help upgrade code written for the Azure Storage Python library before version 0.30.0.

The best way to see how to upgrade a specific API is to take a look at the usage samples in the [Samples](#) directory on GitHub. A very specific set of changes as well as additions can be found in the [ChangeLog](#) and [BreakingChanges](#) documents. The below is a summary of those documents containing the most relevant pieces for the upgrade scenario.

General Changes

In general, we attempted to use more appropriate Python types for parameter and return values rather than always taking and receiving strings. Parameter and return values previously prefixed with `x_ms` were simplified by removing this prefix, and parameter values are divided by `'_'` between words as is idiomatic.

`Listing` returns a generator rather than a result segment. This generator automatically follows continuation tokens as more results are requested.

SAS methods take several individual parameters rather than a single parameter object. Similarly, ACL getters and setters take dictionaries mapping id to `AccessPolicy` rather than a list of `SignedIdentifiers` each holding an id and an `AccessPolicy`.

Blob

The single BlobService object was divided into three subservices for the different blob types (BlockBlobService, PageBlobService, AppendBlobService) with common methods in the abstract BaseBlobService these inherit from. This was done for two reasons. First, to reduce confusion when blob-type specific methods were used on the incorrect blob type. Second, to simplify the BlobService object which had grown to be quite extensive when append blob was added.

ContentSettings objects have replaced all content_* and cache_control parameters and return values for applicable APIs. This is intended to highlight that the semantics of setting content properties is replace rather than merge, meaning that setting any one property will cause any unspecified properties to be cleared on the blob.

On the performance side, single-threaded blob download APIs will now download the blob without chunking to improve perf and not do an initial get to find the size of the blob. However, as a result the progress_callback may receive None for its total parameter when parallelism is off to allow for this optimization.

Queue

The largest change to the QueueService class is that queue messages are both XML encoded and decoded by default. In past versions, either messages were not encoded or decoded by default, or only encoded. Encoding and decoding methods can be modified using the QueueService encode_function and decode_function instance variables.

Methods operating on queue messages and which have return values will return QueueMessage objects. QueueMessages objects contain essentially the same fields as previously, but the times are returned as dates rather than strings and the dequeue count is returned as an int.

Table

Rather than having a boolean switch for turning batching on and off, batches are an object which can be populated and then committed. Entities can be sent as dictionaries or as Entity objects, and returned entities are accessible as either objects or dictionaries. Methods which access and modify entities have been simplified so that if they take an entity object they extract the partition key and row key from that object rather than requiring these be sent separately.

All table entity integer values are stored on the service with type Edm.Int64 unless the type is explicitly overridden as Edm.Int32. Formerly, the type was decided based on the size of the number, but this resulted in hard to predict types on the service. So, the more consistent option was chosen.

Operations no longer echo content from the service and JSON is used instead of AtomPub, improving performance.

azure.storage.blob package

Submodules

azure.storage.blob.appendblobservice module

azure.storage.blob.baseblobservice module

azure.storage.blob.blockblobservice module

`azure.storage.blob.models` module

`azure.storage.blob.pageblob.service` module

Module contents

azure.storage.common package

Submodules

`azure.storage.common.cloudstorageaccount` module

`azure.storage.common.models` module

`azure.storage.common.sharedaccesssignature` module

`azure.storage.common.storageclient` module

Module contents

azure.storage.file package

Submodules

`azure.storage.file.fileservice` module

`azure.storage.file.models` module

Module contents

azure.storage.queue package

Submodules

`azure.storage.queue.models` module

`azure.storage.queue.queue.service` module

Module contents

azure