

ListObjectsV2 - Amazon Simple Storage Service

Archived: 2026-04-05 16:09:16 UTC

Returns some or all (up to 1,000) of the objects in a bucket with each request. You can use the request parameters as selection criteria to return a subset of the objects in a bucket. A `200 OK` response can contain valid or invalid XML. Make sure to design your application to parse the contents of the response and handle it appropriately. For more information about listing objects, see [Listing object keys programmatically](#) in the *Amazon S3 User Guide*. To get a list of your buckets, see [ListBuckets](#).

Note

- **General purpose bucket** - For general purpose buckets, `ListObjectsV2` doesn't return prefixes that are related only to in-progress multipart uploads.
- **Directory buckets** - For directory buckets, `ListObjectsV2` response includes the prefixes that are related only to in-progress multipart uploads.
- **Directory buckets** - For directory buckets, you must make requests for this API operation to the Zonal endpoint. These endpoints support virtual-hosted-style requests in the format `https://amzn-s3-demo-bucket.s3express-zone-id.region-code.amazonaws.com/key-name` . Path-style requests are not supported. For more information about endpoints in Availability Zones, see [Regional and Zonal endpoints for directory buckets in Availability Zones](#) in the *Amazon S3 User Guide*. For more information about endpoints in Local Zones, see [Concepts for directory buckets in Local Zones](#) in the *Amazon S3 User Guide*.

Permissions

- **General purpose bucket permissions** - To use this operation, you must have READ access to the bucket. You must have permission to perform the `s3:ListBucket` action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see [Permissions Related to Bucket Subresource Operations](#) and [Managing Access Permissions to Your Amazon S3 Resources](#) in the *Amazon S3 User Guide*.
- **Directory bucket permissions** - To grant access to this API operation on a directory bucket, we recommend that you use the [CreateSession](#) API operation for session-based authorization. Specifically, you grant the `s3express:CreateSession` permission to the directory bucket in a bucket policy or an IAM identity-based policy. Then, you make the `CreateSession` API call on the bucket to obtain a session token. With the session token in your request header, you can make API requests to this operation. After the session token expires, you make another `CreateSession` API call to generate a new session token for use. AWS CLI or SDKs create session and refresh the session token automatically to avoid service interruptions when a session expires. For more information about authorization, see [CreateSession](#) .

Sorting order of returned objects

- **General purpose bucket** - For general purpose buckets, `ListObjectsV2` returns objects in lexicographical order based on their key names.
- **Directory bucket** - For directory buckets, `ListObjectsV2` does not return objects in lexicographical order.

HTTP Host header syntax

Directory buckets - The HTTP Host header syntax is `Bucket-name.s3express-zone-id.region-code.amazonaws.com`.

Important

This section describes the latest revision of this action. We recommend that you use this revised API operation for application development. For backward compatibility, Amazon S3 continues to support the prior version of this API operation, [ListObjects](#).

The following operations are related to `ListObjectsV2`:

- [GetObject](#)
- [PutObject](#)
- [CreateBucket](#)

Important

You must URL encode any signed header values that contain spaces. For example, if your header value is `my file.txt`, containing two spaces after `my`, you must URL encode this value to `my%20%20file.txt`.

Request Syntax

```
GET /?list-type=2&continuation-token= ContinuationToken &delimiter= Delimiter &encoding-type= EncodingType &fetch-
Host: Bucket .s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-optional-object-attributes: OptionalObjectAttributes
```

URI Request Parameters

The request uses the following URI parameters.

[Bucket](#)

Directory buckets - When you use this operation with a directory bucket, you must use virtual-hosted-style requests in the format `Bucket-name.s3express-zone-id.region-code.amazonaws.com`. Path-style requests are not supported. Directory bucket names must be unique in the chosen Zone (Availability Zone or Local Zone). Bucket names must follow the format `bucket-base-name--zone-id--x-s3` (for example, `amzn-s3-demo-bucket--usw2-az1--x-s3`). For information about bucket naming restrictions, see [Directory bucket naming rules](#) in the *Amazon S3 User Guide*.

Access points - When you use this action with an access point for general purpose buckets, you must provide the alias of the access point in place of the bucket name or specify the access point ARN. When you use this action with an access point for directory buckets, you must provide the access point name in place of the bucket name. When using the access point ARN, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this action with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see [Using access points](#) in the *Amazon S3 User Guide*.

Note

Object Lambda access points are not supported by directory buckets.

S3 on Outposts - When you use this action with S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostID.s3-outposts.Region.amazonaws.com`. When you use this action with S3 on Outposts, the destination bucket must be the Outposts access point ARN or the access point alias. For more information about S3 on Outposts, see [What is S3 on Outposts?](#) in the *Amazon S3 User Guide*.

Required: Yes

[continuation-token](#)

`ContinuationToken` indicates to Amazon S3 that the list is being continued on this bucket with a token. `ContinuationToken` is obfuscated and is not a real key. You can use this `ContinuationToken` for pagination of the list results.

[delimiter](#)

A delimiter is a character that you use to group keys.

`CommonPrefixes` is filtered out from results if it is not lexicographically greater than the `StartAfter` value.

Note

- **Directory buckets** - For directory buckets, `/` is the only supported delimiter.

- **Directory buckets** - When you query `ListObjectsV2` with a delimiter during in-progress multipart uploads, the `CommonPrefixes` response parameter contains the prefixes that are associated with the in-progress multipart uploads. For more information about multipart uploads, see [Multipart Upload Overview](#) in the *Amazon S3 User Guide*.

[encoding-type](#)

Encoding type used by Amazon S3 to encode the [object keys](#) in the response. Responses are encoded only in UTF-8. An object key can contain any Unicode character. However, the XML 1.0 parser can't parse certain characters, such as characters with an ASCII value from 0 to 10. For characters that aren't supported in XML 1.0, you can add this parameter to request that Amazon S3 encode the keys in the response. For more information about characters to avoid in object key names, see [Object key naming guidelines](#).

Note

When using the URL encoding type, non-ASCII characters that are used in an object's key name will be percent-encoded according to UTF-8 code values. For example, the object `test_file(3).png` will appear as `test_file%283%29.png`.

Valid Values: `url`

[fetch-owner](#)

The owner field is not present in `ListObjectsV2` by default. If you want to return the owner field with each key in the result, then set the `FetchOwner` field to `true`.

Note

Directory buckets - For directory buckets, the bucket owner is returned as the object owner for all objects.

[max-keys](#)

Sets the maximum number of keys returned in the response. By default, the action returns up to 1,000 key names. The response might contain fewer keys but will never contain more.

[prefix](#)

Limits the response to keys that begin with the specified prefix.

Note

Directory buckets - For directory buckets, only prefixes that end in a delimiter (`/`) are supported.

[start-after](#)

StartAfter is where you want Amazon S3 to start listing from. Amazon S3 starts listing after this specified key. StartAfter can be any key in the bucket.

Note

This functionality is not supported for directory buckets.

x-amz-expected-bucket-owner

The account ID of the expected bucket owner. If the account ID that you provide does not match the actual owner of the bucket, the request fails with the HTTP status code `403 Forbidden` (access denied).

x-amz-optional-object-attributes

Specifies the optional fields that you want returned in the response. Fields that you do not specify are not returned.

Note

This functionality is not supported for directory buckets.

Valid Values: `RestoreStatus`

x-amz-request-payer

Confirms that the requester knows that she or he will be charged for the list objects request in V2 style. Bucket owners need not specify this parameter in their requests.

Note

This functionality is not supported for directory buckets.

Valid Values: `requester`

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <IsTruncated>boolean</IsTruncated>
  <Contents>
    <ChecksumAlgorithm>string</ChecksumAlgorithm>
    ...
    <ChecksumType>string</ChecksumType>
    <ETag>string</ETag>
    <Key>string</Key>
```

```

<LastModified>timestamp</LastModified>
<Owner>
  <DisplayName>string</DisplayName>
  <ID>string</ID>
</Owner>
<RestoreStatus>
  <IsRestoreInProgress>boolean</IsRestoreInProgress>
  <RestoreExpiryDate>timestamp</RestoreExpiryDate>
</RestoreStatus>
<Size>long</Size>
<StorageClass>string</StorageClass>
</Contents>
...
<Name>string</Name>
<Prefix>string</Prefix>
<Delimiter>string</Delimiter>
<MaxKeys>integer</MaxKeys>
<CommonPrefixes>
  <Prefix>string</Prefix>
</CommonPrefixes>
...
<EncodingType>string</EncodingType>
<KeyCount>integer</KeyCount>
<ContinuationToken>string</ContinuationToken>
<NextContinuationToken>string</NextContinuationToken>
<StartAfter>string</StartAfter>
</ListBucketResult>

```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

The following data is returned in XML format by the service.

ListBucketResult

Root level tag for the ListBucketResult parameters.

Required: Yes

CommonPrefixes

All of the keys (up to 1,000) that share the same prefix are grouped together. When counting the total numbers of returns by this API operation, this group of keys is considered as one item.

A response can contain `CommonPrefixes` only if you specify a delimiter.

`CommonPrefixes` contains all (if there are any) keys between `Prefix` and the next occurrence of the string specified by a delimiter.

`CommonPrefixes` lists keys that act like subdirectories in the directory specified by `Prefix`.

For example, if the prefix is `notes/` and the delimiter is a slash (`/`) as in `notes/summer/july`, the common prefix is `notes/summer/`. All of the keys that roll up into a common prefix count as a single return when calculating the number of returns.

Note

- **Directory buckets** - For directory buckets, only prefixes that end in a delimiter (`/`) are supported.
- **Directory buckets** - When you query `ListObjectsV2` with a delimiter during in-progress multipart uploads, the `CommonPrefixes` response parameter contains the prefixes that are associated with the in-progress multipart uploads. For more information about multipart uploads, see [Multipart Upload Overview](#) in the *Amazon S3 User Guide*.

Type: Array of [CommonPrefix](#) data types

[Contents](#)

Metadata about each object returned.

Type: Array of [Object](#) data types

[ContinuationToken](#)

If `ContinuationToken` was sent with the request, it is included in the response. You can use the returned `ContinuationToken` for pagination of the list response.

Type: String

[Delimiter](#)

Causes keys that contain the same string between the `prefix` and the first occurrence of the delimiter to be rolled up into a single result element in the `CommonPrefixes` collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the `MaxKeys` value.

Note

Directory buckets - For directory buckets, `/` is the only supported delimiter.

Type: String

[EncodingType](#)

Encoding type used by Amazon S3 to encode object key names in the XML response.

If you specify the `encoding-type` request parameter, Amazon S3 includes this element in the response, and returns encoded key name values in the following response elements:

`Delimiter`, `Prefix`, `Key`, and `StartAfter` .

Type: String

Valid Values: `url`

IsTruncated

Set to `false` if all of the results were returned. Set to `true` if more keys are available to return. If the number of results exceeds that specified by `MaxKeys` , all of the results might not be returned.

Type: Boolean

KeyCount

`KeyCount` is the number of keys returned with this request. `KeyCount` will always be less than or equal to the `MaxKeys` field. For example, if you ask for 50 keys, your result will include 50 keys or fewer.

Type: Integer

MaxKeys

Sets the maximum number of keys returned in the response. By default, the action returns up to 1,000 key names. The response might contain fewer keys but will never contain more.

Type: Integer

Name

The bucket name.

Type: String

NextContinuationToken

`NextContinuationToken` is sent when `isTruncated` is true, which means there are more keys in the bucket that can be listed. The next list requests to Amazon S3 can be continued with this `NextContinuationToken` . `NextContinuationToken` is obfuscated and is not a real key

Type: String

Prefix

Keys that begin with the indicated prefix.

Note

Directory buckets - For directory buckets, only prefixes that end in a delimiter (/) are supported.

Type: String

StartAfter

If StartAfter was sent with the request, it is included in the response.

Note

This functionality is not supported for directory buckets.

Type: String

Errors

NoSuchBucket

The specified bucket does not exist.

HTTP Status Code: 404

Examples

Sample Request for general purpose buckets: Listing keys

This request returns the objects in `bucket` . The request specifies the `list-type` parameter, which indicates version 2 of the API operation.

```
GET /?list-type=2 HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T233541Z
Authorization: authorization string
Content-Type: text/plain
```

Sample Response for general purpose buckets

This example illustrates one usage of ListObjectsV2.

```
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>bucket</Name>
  <Prefix/>
  <KeyCount>205</KeyCount>
```

```
<MaxKeys>1000</MaxKeys>
<IsTruncated>>false</IsTruncated>
<Contents>
  <Key>my-image.jpg</Key>
  <LastModified>2009-10-12T17:50:30.000Z</LastModified>
  <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
  <Size>434234</Size>
  <StorageClass>STANDARD</StorageClass>
</Contents>
<Contents>
  ...
</Contents>
...
</ListBucketResult>
```

Sample Request for general purpose buckets: Listing keys using the max-keys, prefix, and start-after parameters

In addition to the `list-type` parameter that indicates version 2 of the API operation, the request also specifies additional parameters to retrieve up to three keys in the `quotes` bucket that start with `E` and occur lexicographically after `ExampleGuide.pdf`.

```
GET /?list-type=2&max-keys=3&prefix=E&start-after=ExampleGuide.pdf HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
x-amz-date: 20160430T232933Z
Authorization: authorization string
```

Sample Response for general purpose buckets

This example illustrates one usage of `ListObjectsV2`.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXr3u7EFM67bNgSAxexeEHndCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>quotes</Name>
  <Prefix>E</Prefix>
  <StartAfter>ExampleGuide.pdf</StartAfter>
  <KeyCount>1</KeyCount>
  <MaxKeys>3</MaxKeys>
  <IsTruncated>>false</IsTruncated>
  <Contents>
    <Key>ExampleObject.txt</Key>
    <LastModified>2013-09-17T18:07:53.000Z</LastModified>
    <ETag>"599bab3ed2c697f1d26842727561fd94"</ETag>
    <Size>857</Size>
    <StorageClass>REDUCED_REDUNDANCY</StorageClass>
  </Contents>
</ListBucketResult>
```

Sample Request for general purpose buckets: Listing keys by using the prefix and delimiter parameters

This example illustrates the use of the `prefix` and the `delimiter` parameters in the request. For this example, we assume that you have the following keys in your bucket:

- `sample.jpg`
- `photos/2006/January/sample.jpg`
- `photos/2006/February/sample2.jpg`
- `photos/2006/February/sample3.jpg`
- `photos/2006/February/sample4.jpg`

The following `GET` request specifies the `delimiter` parameter with a value of `/`.

```
GET /?list-type=2&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T235931Z
Authorization: authorization string
```

Sample Response for general purpose buckets

The key `sample.jpg` does not contain the delimiter character, and Amazon S3 returns it in the `Contents` element in the response. However, all of the other keys contain the delimiter character. Amazon S3 groups these keys and returns a single `CommonPrefixes` element with the `Prefix` value `photos/`. The `Prefix` element is a substring that starts at the beginning of these keys and ends at the first occurrence of the specified delimiter.

```
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>example-bucket</Name>
  <Prefix></Prefix>
  <KeyCount>2</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <Delimiter></Delimiter>
  <IsTruncated>>false</IsTruncated>
  <Contents>
    <Key>sample.jpg</Key>
    <LastModified>2011-02-26T01:56:20.000Z</LastModified>
    <ETag>"bf1d737a4d46a19f3bcd6905cc8b902"</ETag>
    <Size>142863</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <CommonPrefixes>
    <Prefix>photos/</Prefix>
  </CommonPrefixes>
</ListBucketResult>
```

Sample Request for general purpose buckets

The following request specifies the `delimiter` parameter with the value `/`, and the `prefix` parameter with the value `photos/2006/`.

```
GET /?list-type=2&prefix=photos/2006/&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160501T000433Z
Authorization: authorization string
```

Sample Response for general purpose buckets

In response, Amazon S3 returns only the keys that start with the specified prefix. Further, Amazon S3 uses the delimiter character to group keys that contain the same substring until the first occurrence of the delimiter character after the specified prefix. For each such key group, Amazon S3 returns one `CommonPrefixes` element in the response. The keys grouped under this `CommonPrefixes` element are not returned elsewhere in the response.

The `Prefix` value returned in the `CommonPrefixes` element is a substring that starts at the beginning of the key and ends at the first occurrence of the specified delimiter after the prefix.

Note

If you created folders by using the Amazon S3 console, you will see an additional 0-byte object with a key of `photos/2006/`. This object is created because of the way that the console supports folder structures. For more information, see [Organizing objects in the Amazon S3 console using folders](#) in the *Amazon S3 User Guide*.

```
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>example-bucket</Name>
  <Prefix>photos/2006/</Prefix>
  <KeyCount>2</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <Delimiter></Delimiter>
  <IsTruncated>>false</IsTruncated>

  <CommonPrefixes>
    <Prefix>photos/2006/February/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/January/</Prefix>
  </CommonPrefixes>
</ListBucketResult>
```

Sample Request for general purpose buckets: Using a continuation token

In this example, the initial request returns more than 1,000 keys. In response to this request, Amazon S3 returns the `IsTruncated` element with the value set to `true` and with a `NextContinuationToken` element.

```
GET /?list-type=2 HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 02 May 2016 23:17:07 GMT
Authorization: authorization string
```

Sample Response for general purpose buckets: Using a continuation token

This example illustrates one usage of ListObjectsV2.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXr3u7EFM67bNgSAxexeEHndCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>bucket</Name>
  <Prefix></Prefix>
  <NextContinuationToken>1ueGcxLPRx1Tr/XYExHnhbYLgveDs2J/wm36Hy4vb0wM=</NextContinuationToken>
  <KeyCount>1000</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>true</IsTruncated>
  <Contents>
    <Key>happyface.jpg</Key>
    <LastModified>2014-11-21T19:40:05.000Z</LastModified>
    <ETag>"70ee1738b6b21e2c8a43f3a5ab0eee71"</ETag>
    <Size>11</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  ...
</ListBucketResult>
```

Sample request for general purpose buckets

In the following subsequent request, we include a `continuation-token` query parameter in the request with the value of the `NextContinuationToken` element from the preceding response.

```
GET /?list-type=2 HTTP/1.1
GET /?list-type=2&continuation-token=1ueGcxLPRx1Tr/XYExHnhbYLgveDs2J/wm36Hy4vb0wM= HTTP/1.1

Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 02 May 2016 23:17:07 GMT
Authorization: authorization string
```

Sample response for general purpose buckets:

Amazon S3 returns a list of the next set of keys starting where the previous request ended.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXr3u7EFM67bNgSAxexeEHndCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>bucket</Name>
  <Prefix></Prefix>
  <ContinuationToken>1ueGcxLPRx1Tr/XYExHnhbYLgveDs2J/wm36Hy4vb0wM=</ContinuationToken>
  <KeyCount>112</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>>false</IsTruncated>
  <Contents>
    <Key>happyfacex.jpg</Key>
    <LastModified>2014-11-21T19:40:05.000Z</LastModified>
    <ETag>"70ee1738b6b21e2c8a43f3a5ab0eee71"</ETag>
    <Size>1111</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  ...
</ListBucketResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)

- [AWS SDK for Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Source: https://docs.aws.amazon.com/AmazonS3/latest/API/API_ListObjectsV2.html