
Amazon Simple Queue Service

API Reference

API Version 2012-11-05



Amazon Simple Queue Service: API Reference

Copyright © 2015 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

The following are trademarks of Amazon Web Services, Inc.: Amazon, Amazon Web Services Design, AWS, Amazon CloudFront, AWS CloudTrail, AWS CodeDeploy, Amazon Cognito, Amazon DevPay, DynamoDB, ElastiCache, Amazon EC2, Amazon Elastic Compute Cloud, Amazon Glacier, Amazon Kinesis, Kindle, Kindle Fire, AWS Marketplace Design, Mechanical Turk, Amazon Redshift, Amazon Route 53, Amazon S3, Amazon VPC, and Amazon WorkDocs. In addition, Amazon.com graphics, logos, page headers, button icons, scripts, and service names are trademarks, or trade dress of Amazon in the U.S. and/or other countries. Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon.

All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
Actions	2
AddPermission	3
Request Parameters	3
Errors	4
Examples	4
ChangeMessageVisibility	5
Request Parameters	5
Errors	6
Examples	6
ChangeMessageVisibilityBatch	7
Request Parameters	7
Response Elements	7
Errors	7
Examples	8
CreateQueue	10
Request Parameters	10
Response Elements	11
Errors	11
Examples	11
DeleteMessage	13
Request Parameters	13
Errors	13
Examples	14
DeleteMessageBatch	15
Request Parameters	15
Response Elements	15
Errors	15
Examples	16
DeleteQueue	18
Request Parameters	18
Errors	18
Examples	18
GetQueueAttributes	20
Request Parameters	20
Response Elements	21
Errors	21
Examples	21
GetQueueUrl	24
Request Parameters	24
Response Elements	24
Errors	24
Examples	24
ListDeadLetterSourceQueues	26
Request Parameters	26
Response Elements	26
Errors	26
Examples	26
ListQueues	28
Request Parameters	28
Response Elements	28
Errors	28
Examples	28
PurgeQueue	30
Request Parameters	30

Errors	30
Examples	30
ReceiveMessage	32
Request Parameters	32
Response Elements	33
Errors	34
Examples	34
RemovePermission	36
Request Parameters	36
Errors	36
Examples	36
SendMessage	38
Request Parameters	38
Response Elements	38
Errors	39
Examples	39
SendMessageBatch	41
Request Parameters	41
Response Elements	41
Errors	42
Examples	42
SetQueueAttributes	44
Request Parameters	44
Errors	45
Examples	45
Data Types	47
BatchResultErrorEntry	47
Description	47
Contents	47
ChangeMessageVisibilityBatchRequestEntry	48
Description	48
Contents	48
ChangeMessageVisibilityBatchResultEntry	49
Description	49
Contents	49
DeleteMessageBatchRequestEntry	49
Description	49
Contents	49
DeleteMessageBatchResultEntry	49
Description	49
Contents	50
Message	50
Description	50
Contents	50
MessageAttributeValue	51
Description	51
Contents	51
SendMessageBatchRequestEntry	52
Description	52
Contents	52
SendMessageBatchResultEntry	53
Description	53
Contents	53
Common Parameters	54
.....	54
Common Errors	56
.....	56

Welcome

Welcome to the *Amazon Simple Queue Service API Reference*. This section describes who should read this guide, how the guide is organized, and other resources related to the Amazon Simple Queue Service (Amazon SQS).

Amazon SQS offers reliable and scalable hosted queues for storing messages as they travel between computers. By using Amazon SQS, you can move data between distributed components of your applications that perform different tasks without losing messages or requiring each component to be always available.

Helpful Links:

- [Current WSDL \(2012-11-05\)](#)
- [Making API Requests](#)
- [Amazon SQS product page](#)
- [Using Amazon SQS Message Attributes](#)
- [Using Amazon SQS Dead Letter Queues](#)
- [Regions and Endpoints](#)

We also provide SDKs that enable you to access Amazon SQS from your preferred programming language. The SDKs contain functionality that automatically takes care of tasks such as:

- Cryptographically signing your service requests
- Retrying requests
- Handling error responses

For a list of available SDKs, go to [Tools for Amazon Web Services](#).

This document was last updated on September 3, 2015.

Actions

The following actions are supported:

- [AddPermission](#) (p. 3)
- [ChangeMessageVisibility](#) (p. 5)
- [ChangeMessageVisibilityBatch](#) (p. 7)
- [CreateQueue](#) (p. 10)
- [DeleteMessage](#) (p. 13)
- [DeleteMessageBatch](#) (p. 15)
- [DeleteQueue](#) (p. 18)
- [GetQueueAttributes](#) (p. 20)
- [GetQueueUrl](#) (p. 24)
- [ListDeadLetterSourceQueues](#) (p. 26)
- [ListQueues](#) (p. 28)
- [PurgeQueue](#) (p. 30)
- [ReceiveMessage](#) (p. 32)
- [RemovePermission](#) (p. 36)
- [SendMessage](#) (p. 38)
- [SendMessageBatch](#) (p. 41)
- [SetQueueAttributes](#) (p. 44)

AddPermission

Adds a permission to a queue for a specific [principal](#). This allows for sharing access to the queue.

When you create a queue, you have full control access rights for the queue. Only you (as owner of the queue) can grant or deny permissions to the queue. For more information about these permissions, see [Shared Queues](#) in the *Amazon SQS Developer Guide*.

Note

AddPermission writes an Amazon SQS-generated policy. If you want to write your own policy, use SetQueueAttributes to upload your policy. For more information about writing your own policy, see Using The Access Policy Language in the Amazon SQS Developer Guide.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

ActionName.member.N

The action the client wants to allow for the specified principal. The following are valid values: `*` | `SendMessage` | `ReceiveMessage` | `DeleteMessage` | `ChangeMessageVisibility` | `GetQueueAttributes` | `GetQueueUrl`. For more information about these actions, see [Understanding Permissions](#) in the *Amazon SQS Developer Guide*.

Specifying `SendMessage`, `DeleteMessage`, or `ChangeMessageVisibility` for the `ActionName.n` also grants permissions for the corresponding batch versions of those actions: `SendMessageBatch`, `DeleteMessageBatch`, and `ChangeMessageVisibilityBatch`.

Type: String list

Required: Yes

AWSAccountId.member.N

The AWS account number of the [principal](#) who will be given permission. The principal must have an AWS account, but does not need to be signed up for Amazon SQS. For information about locating the AWS account identification, see [Your AWS Identifiers](#) in the *Amazon SQS Developer Guide*.

Type: String list

Required: Yes

Label

The unique identification of the permission you're setting (e.g., `AliceSendMessage`). Constraints: Maximum 80 characters; alphanumeric characters, hyphens (-), and underscores (_) are allowed.

Type: String

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

OverLimit

The action that you requested would violate a limit. For example, `ReceiveMessage` returns this error if the maximum number of messages in flight has already been reached. [AddPermission \(p. 3\)](#) returns this error if the maximum number of permissions for the queue has already been reached.

HTTP Status Code: 403

Examples

The following example Query request grants a `SendMessage` permission to the principal whose AWS account number is 125074342641. How you structure the `AUTHPARAMS` depends on how you are signing your API request. For information on `AUTHPARAMS` in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=AddPermission  
&Label=testLabel  
&AWSAccountId.1=125074342641  
&ActionName.1=SendMessage  
&AWSAccountId.2=125074342642  
&ActionName.2=ReceiveMessage  
&Version=2012-11-05  
&Expires=2012-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<AddPermissionResponse>  
  <ResponseMetadata>  
    <RequestId>  
      9a285199-c8d6-47c2-bdb2-314cb47d599d  
    </RequestId>  
  </ResponseMetadata>  
</AddPermissionResponse>
```


ChangeMessageVisibility

Changes the visibility timeout of a specified message in a queue to a new value. The maximum allowed timeout value you can set the value to is 12 hours. This means you can't extend the timeout of a message in an existing queue to more than a total visibility timeout of 12 hours. (For more information visibility timeout, see [Visibility Timeout](#) in the *Amazon SQS Developer Guide*.)

For example, let's say you have a message and its default message visibility timeout is 5 minutes. After 3 minutes, you call `ChangeMessageVisibility` with a timeout of 10 minutes. At that time, the timeout for the message would be extended by 10 minutes beyond the time of the `ChangeMessageVisibility` call. This results in a total visibility timeout of 13 minutes. You can continue to call `ChangeMessageVisibility` to extend the visibility timeout to a maximum of 12 hours. If you try to extend beyond 12 hours, the request will be rejected.

Note

There is a 120,000 limit for the number of inflight messages per queue. Messages are inflight after they have been received from the queue by a consuming component, but have not yet been deleted from the queue. If you reach the 120,000 limit, you will receive an `OverLimit` error message from Amazon SQS. To help avoid reaching the limit, you should delete the messages from the queue after they have been processed. You can also increase the number of queues you use to process the messages.

Important

If you attempt to set the `VisibilityTimeout` to an amount more than the maximum time left, Amazon SQS returns an error. It will not automatically recalculate and increase the timeout to the maximum time remaining.

Important

Unlike with a queue, when you change the visibility timeout for a specific message, that timeout value is applied immediately but is not saved in memory for that message. If you don't delete a message after it is received, the visibility timeout for the message the next time it is received reverts to the original timeout value, not the value you set with the `ChangeMessageVisibility` action.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

ReceiptHandle

The receipt handle associated with the message whose visibility timeout should be changed. This parameter is returned by the [ReceiveMessage \(p. 32\)](#) action.

Type: String

Required: Yes

VisibilityTimeout

The new value (in seconds - from 0 to 43200 - maximum 12 hours) for the message's visibility timeout.

Type: Integer

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

MessageNotInflight

The message referred to is not in flight.

HTTP Status Code: 400

ReceiptHandleIsInvalid

The receipt handle provided is not valid.

HTTP Status Code: 400

Examples

The following example Query request changes the visibility timeout for a message to 60 seconds. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=ChangeMessageVisibility  
&VisibilityTimeout=60  
&ReceiptHandle=MbZj6wDwli%2BJvwwJaBV%2B3dcjk2YW2vA3%2BSTFF1jT  
M8tJJg6HRG6PYSasuWXPJB%2BCwLj1FjgXUv1uSj1gUPAWV66FU/WeR4mq2OKpEGY  
WbnImpRCJVAyeMjeU5ZBdteQ%2BQEauMZc8ZRv37sIW2iJKq3M9MFx1YvV11A2x/K  
SbkJ0=  
&Version=2012-11-05  
&Expires=2012-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<ChangeMessageVisibilityResponse>  
  <ResponseMetadata>  
    <RequestId>  
      6a7a282a-d013-4a59-aba9-335b0fa48bed  
    </RequestId>  
  </ResponseMetadata>  
</ChangeMessageVisibilityResponse>
```

ChangeMessageVisibilityBatch

Changes the visibility timeout of multiple messages. This is a batch version of [ChangeMessageVisibility \(p. 5\)](#). The result of the action on each message is reported individually in the response. You can send up to 10 [ChangeMessageVisibility \(p. 5\)](#) requests with each `ChangeMessageVisibilityBatch` action.

Important

Because the batch request can result in a combination of successful and unsuccessful actions, you should check for batch errors even when the call returns an HTTP status code of 200.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

ChangeMessageVisibilityBatchRequestEntry.member.N

A list of receipt handles of the messages for which the visibility timeout must be changed.

Type: [ChangeMessageVisibilityBatchRequestEntry \(p. 48\)](#) list

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Response Elements

The following elements are returned.

Failed

A list of [BatchResultErrorEntry \(p. 47\)](#) items.

Type: [BatchResultErrorEntry \(p. 47\)](#) list

Successful

A list of [ChangeMessageVisibilityBatchResultEntry \(p. 49\)](#) items.

Type: [ChangeMessageVisibilityBatchResultEntry \(p. 49\)](#) list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

BatchEntryIdsNotDistinct

Two or more batch entries have the same `Id` in the request.

HTTP Status Code: 400

EmptyBatchRequest

Batch request does not contain an entry.

HTTP Status Code: 400

InvalidBatchEntryId

The `Id` of a batch entry in a batch request does not abide by the specification.

HTTP Status Code: 400

TooManyEntriesInBatchRequest

Batch request contains more number of entries than permissible.

HTTP Status Code: 400

Examples

`ChangeMessageVisibilityBatch` request changes the visibility timeout settings for two messages. You must URL encode the entire URL; however, we've URL encoded only the message body to make the example easier for you to read.

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/
&Action=ChangeMessageVisibilityBatch
&Version=2012-11-05
&ChangeMessageVisibilityBatchRequestEntry.1.Id=change_visibility_msg_2
&ChangeMessageVisibilityBatchRequestEntry.1.Re
ceiptHandle=gfk0T0R0waama4fVFffkjKznhMCymjQvfTFk2LxT33G4ms5subrE0deLK
WSscPU1oD3J9zgeS4PQQ3U30qOumIE6AdAv3w%2F%2FalIXW6AqaWhGsEPaLm3Vf6Ii
WqdM8u5imB%2BNTwj3tQRzOWdTOePjOjPcTpRxBtXix%2BEvwJOZUma9wabv%2BSw6ZHjwm
NcVDx8dZXJhVp16Bksiox%2FGrUvrVTCJRTWTLc59oHLLF8sEkKzRmGNzTDGTiV%2BYjH
fQj60FD3rVaXmzTsonNxRhKJ72uIHVMGVQiAGgBX6HGv9LDmYhPXw4hy%2FNgIg%3D%3D
&ChangeMessageVisibilityBatchRequestEntry.1.VisibilityTimeout=45
&ChangeMessageVisibilityBatchRequestEntry.2.Id=change_visibility_msg_3
&ChangeMessageVisibilityBatchRequestEntry.2.Re
ceiptHandle=gfk0T0R0waama4fVFffkjKznhMCymjQvfTFk2LxT33FUgBz3%2BnougdeLK
WSscPU1%2FXgx%2BxcNnJnQQ3U30qOumIE6AdAv3w%2F%2FalIXW6AqaWhGsEPaLm3Vf6Ii
WqdM8u5imB%2BNTwj3tQRzOWdTOePjOsogjZM%2F7kzn4Ew27XLU9I%2FYaWYmKvDbq%2Fk3HK
VB9HfB43ke49atP2aWrzNL4yunG41Q4cfRRtfJdcGQGNHQ2%2Byd0U5f5qRldZr1iDo5xk946eQat83Ax
TRP%2BY4Qi0V7FAeSLH9su9xpX6HGv9LDmYhPXw4hy%2FNgIg%3D%3D
&ChangeMessageVisibilityBatchRequestEntry.2.VisibilityTimeout=45
&Expires=2012-10-18T22%3A52%3A43PST
```

Sample Response

Amazon Simple Queue Service API Reference Examples

```
<ChangeMessageVisibilityBatchResponse>
  <ChangeMessageVisibilityBatchResult>
    <ChangeMessageVisibilityBatchResultEntry>
      <Id>change_visibility_msg_2</Id>
    </ChangeMessageVisibilityBatchResultEntry>
    <ChangeMessageVisibilityBatchResultEntry>
      <Id>change_visibility_msg_3</Id>
    </ChangeMessageVisibilityBatchResultEntry>
  </ChangeMessageVisibilityBatchResult>
  <ResponseMetadata>
    <RequestId>ca9668f7-ab1b-4f7a-8859-f15747ab17a7</RequestId>
  </ResponseMetadata>
</ChangeMessageVisibilityBatchResponse>
```

CreateQueue

Creates a new queue, or returns the URL of an existing one. When you request `CreateQueue`, you provide a name for the queue. To successfully create a new queue, you must provide a name that is unique within the scope of your own queues.

Note

If you delete a queue, you must wait at least 60 seconds before creating a queue with the same name.

You may pass one or more attributes in the request. If you do not provide a value for any attribute, the queue will have the default value for that attribute. Permitted attributes are the same that can be set using [SetQueueAttributes](#) (p. 44).

Note

Use `GetQueueUrl` to get a queue's URL. `GetQueueUrl` requires only the `QueueName` parameter.

If you provide the name of an existing queue, along with the exact names and values of all the queue's attributes, `CreateQueue` returns the queue URL for the existing queue. If the queue name, attribute names, or attribute values do not match an existing queue, `CreateQueue` returns an error.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters](#) (p. 54).

Attribute.entry.N

A map of attributes with their corresponding values.

The following lists the names, descriptions, and values of the special request parameters the `CreateQueue` action uses:

- `DelaySeconds` - The time in seconds that the delivery of all messages in the queue will be delayed. An integer from 0 to 900 (15 minutes). The default for this attribute is 0 (zero).
- `MaximumMessageSize` - The limit of how many bytes a message can contain before Amazon SQS rejects it. An integer from 1024 bytes (1 KiB) up to 262144 bytes (256 KiB). The default for this attribute is 262144 (256 KiB).
- `MessageRetentionPeriod` - The number of seconds Amazon SQS retains a message. Integer representing seconds, from 60 (1 minute) to 1209600 (14 days). The default for this attribute is 345600 (4 days).
- `Policy` - The queue's policy. A valid AWS policy. For more information about policy structure, see [Overview of AWS IAM Policies](#) in the *Amazon IAM User Guide*.
- `ReceiveMessageWaitTimeSeconds` - The time for which a [ReceiveMessage](#) (p. 32) call will wait for a message to arrive. An integer from 0 to 20 (seconds). The default for this attribute is 0.
- `VisibilityTimeout` - The visibility timeout for the queue. An integer from 0 to 43200 (12 hours). The default for this attribute is 30. For more information about visibility timeout, see [Visibility Timeout](#) in the *Amazon SQS Developer Guide*.

Type: String to String map

Valid Map Keys: Policy | VisibilityTimeout | MaximumMessageSize | MessageRetentionPeriod | ApproximateNumberOfMessages | ApproximateNumberOfMessagesNotVisible | CreatedTimestamp | LastModifiedTimestamp | QueueArn | ApproximateNumberOfMessagesDelayed | DelaySeconds | ReceiveMessageWaitTimeSeconds | RedrivePolicy

Required: No

QueueName

The name for the queue to be created.

Type: String

Required: Yes

Response Elements

The following element is returned.

QueueUrl

The URL for the created Amazon SQS queue.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

QueueDeletedRecently

You must wait 60 seconds after deleting a queue before you can create another with the same name.

HTTP Status Code: 400

QueueNameExists

A queue already exists with this name. Amazon SQS returns this error only if the request includes attributes whose values differ from those of the existing queue.

HTTP Status Code: 400

Examples

The following example Query request creates a new queue named testQueue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/  
?Action=CreateQueue  
&QueueName=testQueue  
&Attribute.1.Name=VisibilityTimeout  
&Attribute.1.Value=40  
&Version=2012-11-05
```

```
&Expires=2012-10-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<CreateQueueResponse>  
  <CreateQueueResult>  
    <QueueUrl>  
      http://&useast1-query;/123456789012/testQueue  
    </QueueUrl>  
  </CreateQueueResult>  
  <ResponseMetadata>  
    <RequestId>  
      7a62c49f-347e-4fc4-9331-6e8e7a96aa73  
    </RequestId>  
  </ResponseMetadata>  
</CreateQueueResponse>
```


DeleteMessage

Deletes the specified message from the specified queue. You specify the message by using the message's `receipt handle` and not the `message ID` you received when you sent the message. Even if the message is locked by another reader due to the visibility timeout setting, it is still deleted from the queue. If you leave a message in the queue for longer than the queue's configured retention period, Amazon SQS automatically deletes it.

Note

The receipt handle is associated with a specific instance of receiving the message. If you receive a message more than once, the receipt handle you get each time you receive the message is different. When you request `DeleteMessage`, if you don't provide the most recently received receipt handle for the message, the request will still succeed, but the message might not be deleted.

Important

It is possible you will receive a message even after you have deleted it. This might happen on rare occasions if one of the servers storing a copy of the message is unavailable when you request to delete the message. The copy remains on the server and might be returned to you again on a subsequent receive request. You should create your system to be idempotent so that receiving a particular message more than once is not a problem.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

ReceiptHandle

The receipt handle associated with the message to delete.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

InvalidIdFormat

The receipt handle is not valid for the current version.

HTTP Status Code: 400

ReceiptHandleIsInvalid

The receipt handle provided is not valid.

HTTP Status Code: 400

Examples

The following example Query request deletes a message from the queue named testQueue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to Examples of Signed Signature Version 4 Requests.

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=DeleteMessage  
&ReceiptHandle=MbZj6wDWli%2BJvwwJaBV%2B3dcjk2YW2vA3%2BSTFFljT  
M8tJJg6HRG6PYSasuWXPJB%2BCwLj1FjgXUv1uSjl9UPAWV66FU/WeR4mq2OKpEGY  
WbnLmpRCJVAyeMjeU5ZBdtcQ%2BQEauMZc8ZRv37sIW2iJKq3M9MFx1YvV11A2x/K  
SbkJ0=  
&Version=2012-11-05  
&Expires=2012-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<DeleteMessageResponse>  
  <ResponseMetadata>  
    <RequestId>  
      b5293cb5-d306-4a17-9048-b263635abe42  
    </RequestId>  
  </ResponseMetadata>  
</DeleteMessageResponse>
```

DeleteMessageBatch

Deletes up to ten messages from the specified queue. This is a batch version of [DeleteMessage \(p. 13\)](#). The result of the delete action on each message is reported individually in the response.

Important

Because the batch request can result in a combination of successful and unsuccessful actions, you should check for batch errors even when the call returns an HTTP status code of 200.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

DeleteMessageBatchRequestEntry.member.N

A list of receipt handles for the messages to be deleted.

Type: [DeleteMessageBatchRequestEntry \(p. 49\)](#) list

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Response Elements

The following elements are returned.

Failed

A list of [BatchResultErrorEntry \(p. 47\)](#) items.

Type: [BatchResultErrorEntry \(p. 47\)](#) list

Successful

A list of [DeleteMessageBatchResultEntry \(p. 49\)](#) items.

Type: [DeleteMessageBatchResultEntry \(p. 49\)](#) list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

BatchEntryIdsNotDistinct

Two or more batch entries have the same `Id` in the request.

HTTP Status Code: 400

EmptyBatchRequest

Batch request does not contain an entry.

HTTP Status Code: 400

InvalidBatchEntryId

The `id` of a batch entry in a batch request does not abide by the specification.

HTTP Status Code: 400

TooManyEntriesInBatchRequest

Batch request contains more number of entries than permissible.

HTTP Status Code: 400

Examples

The following example `DeleteMessageBatch` request deletes two messages. You must URL encode the entire URL; however, we've URL encoded only the message body to make the example easier for you to read. How you structure the `AUTHPARAMS` depends on how you are signing your API request. For information on `AUTHPARAMS` in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/
&Action=DeleteMessageBatch
&Version=2012-11-05
&DeleteMessageBatchRequestEntry.1.Id=msg1
&DeleteMessageBatchRequestEntry.1.ReceiptHandle=gfk0T0R0waama4fVffkjPQrrvzM
rOg0fTFk2LxT33EuB8wR0ZCFgKWYXGWFOqqpCIiprQUEhir%2F5LeGPyTLzjqLQxyQYaQALeS
NHb0us3uE84uujxpBhsDkZUQkjFFkNqBXn48x1McVhTcI3YLH%2Bd%2BIqetIOHgBCZAPx6r%2B09dW
aBXei6nbK5Ygih21DCDdAwFV68Jo8DXhb3ErEfoDqx7vyvC5nCp
dwqv%2BJhU%2FTNGjNN8t51v5c%2FAXvQsAzyZVNapxUrHit4NxRhKJ72uICcxruyE8eRXlxIVN
geNP8ZEDcw7zZU1Zw%3D%3D
&DeleteMessageBatchRequestEntry.2.Id=msg2
&DeleteMessageBatchRequestEntry.2.ReceiptHandle=gfk0T0R0waama4fVffkjKzmmH
CymjQvfTFk2LxT33G4ms5subrE0deLKWSScPU1oD3J9zgeS4PQQ3U30qOu
mIE6AdAv3w%2F%2Fa1IXW6AqaWhGsEPaLm3Vf6IiWqdM8u5imB%2BNTwj3tQRzOWdTOepjOjPcTpRxB
tXix%2BEvwJOZUma9wabv%2BSw6ZHjwmNcVDx8dZXJhVp16Bksi
ox%2FGrUvrVTCJRTWTLc59oHLLF8sEkKzRmGNzTDGTiV%2BYjHfQj60FD3rVaXmzT
soNxRhKJ72uIHVMGVQIAGgB%2BqAbSqfKHDQtVomJJgkHug%3D%3D
&Expires=2014-10-18T22%3A52%3A43PST
&AUTHPARAMS
```

Sample Response

```
<DeleteMessageBatchResponse>
  <DeleteMessageBatchResult>
```

Amazon Simple Queue Service API Reference Examples

```
<DeleteMessageBatchResultEntry>
  <Id>msg1</Id>
</DeleteMessageBatchResultEntry>
<DeleteMessageBatchResultEntry>
  <Id>msg2</Id>
</DeleteMessageBatchResultEntry>
</DeleteMessageBatchResult>
<ResponseMetadata>
  <RequestId>d6f86b7a-74d1-4439-b43f-196a1e29cd85</RequestId>
</ResponseMetadata>
</DeleteMessageBatchResponse>
```

DeleteQueue

Deletes the queue specified by the **queue URL**, regardless of whether the queue is empty. If the specified queue does not exist, Amazon SQS returns a successful response.

Important

Use DeleteQueue with care; once you delete your queue, any messages in the queue are no longer available.

When you delete a queue, the deletion process takes up to 60 seconds. Requests you send involving that queue during the 60 seconds might succeed. For example, a [SendMessage \(p. 38\)](#) request might succeed, but after the 60 seconds, the queue and that message you sent no longer exist. Also, when you delete a queue, you must wait at least 60 seconds before creating a queue with the same name.

We reserve the right to delete queues that have had no activity for more than 30 days. For more information, see [How Amazon SQS Queues Work](#) in the *Amazon SQS Developer Guide*.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

Examples

The following example Query request deletes the specified queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=DeleteQueue  
&Version=2012-11-05  
&Expires=2012-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<DeleteQueueResponse>
  <ResponseMetadata>
    <RequestId>
      6fde8d1e-52cd-4581-8cd9-c512f4c64223
    </RequestId>
  </ResponseMetadata>
</DeleteQueueResponse>
```

GetQueueAttributes

Gets attributes for the specified queue. The following attributes are supported:

- `All` - returns all values.
- `ApproximateNumberOfMessages` - returns the approximate number of visible messages in a queue. For more information, see [Resources Required to Process Messages](#) in the *Amazon SQS Developer Guide*.
- `ApproximateNumberOfMessagesNotVisible` - returns the approximate number of messages that are not timed-out and not deleted. For more information, see [Resources Required to Process Messages](#) in the *Amazon SQS Developer Guide*.
- `VisibilityTimeout` - returns the visibility timeout for the queue. For more information about visibility timeout, see [Visibility Timeout](#) in the *Amazon SQS Developer Guide*.
- `CreatedTimestamp` - returns the time when the queue was created (epoch time in seconds).
- `LastModifiedTimestamp` - returns the time when the queue was last changed (epoch time in seconds).
- `Policy` - returns the queue's policy.
- `MaximumMessageSize` - returns the limit of how many bytes a message can contain before Amazon SQS rejects it.
- `MessageRetentionPeriod` - returns the number of seconds Amazon SQS retains a message.
- `QueueArn` - returns the queue's Amazon resource name (ARN).
- `ApproximateNumberOfMessagesDelayed` - returns the approximate number of messages that are pending to be added to the queue.
- `DelaySeconds` - returns the default delay on the queue in seconds.
- `ReceiveMessageWaitTimeSeconds` - returns the time for which a `ReceiveMessage` call will wait for a message to arrive.
- `RedrivePolicy` - returns the parameters for dead letter queue functionality of the source queue. For more information about `RedrivePolicy` and dead letter queues, see [Using Amazon SQS Dead Letter Queues](#) in the *Amazon SQS Developer Guide*.

Note

Going forward, new attributes might be added. If you are writing code that calls this action, we recommend that you structure your code so that it can handle new attributes gracefully.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

AttributeName.member.N

A list of attributes to retrieve information for.

Type: String list

Valid Values: `Policy` | `VisibilityTimeout` | `MaximumMessageSize` | `MessageRetentionPeriod` | `ApproximateNumberOfMessages` |

ApproximateNumberOfMessagesNotVisible | CreatedTimestamp |
LastModifiedTimestamp | QueueArn | ApproximateNumberOfMessagesDelayed |
DelaySeconds | ReceiveMessageWaitTimeSeconds | RedrivePolicy

Required: No

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Response Elements

The following element is returned.

Attributes

A map of attributes to the respective values.

Type: String to String map

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

InvalidAttributeName

The attribute referred to does not exist.

HTTP Status Code: 400

Examples

The following example Query requests gets all the attribute values for the specified queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=GetQueueAttributes  
&AttributeName.1=All  
&Version=2012-11-05  
&Expires=2013-10-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<GetQueueAttributesResponse>
  <GetQueueAttributesResult>
    <Attribute>
      <Name>ReceiveMessageWaitTimeSeconds</Name>
      <Value>2</Value>
    </Attribute>
    <Attribute>
      <Name>VisibilityTimeout</Name>
      <Value>30</Value>
    </Attribute>
    <Attribute>
      <Name>ApproximateNumberOfMessages</Name>
      <Value>0</Value>
    </Attribute>
    <Attribute>
      <Name>ApproximateNumberOfMessagesNotVisible</Name>
      <Value>0</Value>
    </Attribute>
    <Attribute>
      <Name>CreatedTimestamp</Name>
      <Value>1286771522</Value>
    </Attribute>
    <Attribute>
      <Name>LastModifiedTimestamp</Name>
      <Value>1286771522</Value>
    </Attribute>
    <Attribute>
      <Name>QueueArn</Name>
      <Value>arn:aws:sqs:us-east-1:123456789012:qfoo</Value>
    </Attribute>
    <Attribute>
      <Name>MaximumMessageSize</Name>
      <Value>8192</Value>
    </Attribute>
    <Attribute>
      <Name>MessageRetentionPeriod</Name>
      <Value>345600</Value>
    </Attribute>
  </GetQueueAttributesResult>
  <ResponseMetadata>
    <RequestId>1ea71be5-b5a2-4f9d-b85a-945d8d08cd0b</RequestId>
  </ResponseMetadata>
</GetQueueAttributesResponse>
```

The following example Query request gets three attribute values for the specified queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to Examples of Signed Signature Version 4 Requests.

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=GetQueueAttributes  
&Action=GetQueueAttributes  
&Version=2012-11-05  
&AttributeName.1=VisibilityTimeout  
&AttributeName.2=DelaySeconds  
&AttributeName.3=ReceiveMessageWaitTimeSeconds  
&Expires=2014-10-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<GetQueueAttributesResponse>  
  <GetQueueAttributesResult>  
    <Attribute>  
      <Name>VisibilityTimeout</Name>  
      <Value>30</Value>  
    </Attribute>  
    <Attribute>  
      <Name>DelaySeconds</Name>  
      <Value>0</Value>  
    </Attribute>  
    <Attribute>  
      <Name>ReceiveMessageWaitTimeSeconds</Name>  
      <Value>2</Value>  
    </Attribute>  
  </GetQueueAttributesResult>  
</GetQueueAttributesResponse>
```

GetQueueUrl

Returns the URL of an existing queue. This action provides a simple way to retrieve the URL of an Amazon SQS queue.

To access a queue that belongs to another AWS account, use the `QueueOwnerAWSAccountId` parameter to specify the account ID of the queue's owner. The queue's owner must grant you permission to access the queue. For more information about shared queue access, see [AddPermission \(p. 3\)](#) or go to [Shared Queues](#) in the *Amazon SQS Developer Guide*.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueName

The name of the queue whose URL must be fetched. Maximum 80 characters; alphanumeric characters, hyphens (-), and underscores (_) are allowed.

Type: String

Required: Yes

QueueOwnerAWSAccountId

The AWS account ID of the account that created the queue.

Type: String

Required: No

Response Elements

The following element is returned.

QueueUrl

The URL for the queue.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

QueueDoesNotExist

The queue referred to does not exist.

HTTP Status Code: 400

Examples

The following example Query request gets the URL for the specified queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/  
?Action=GetQueueUrl  
&QueueName=testQueue  
&Version=2012-11-05  
&Expires=2011-10-24T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<GetQueueUrlResponse>  
  <GetQueueUrlResult>  
    <QueueUrl>  
      http://sqs.us-east-1.amazonaws.com/123456789012/testQueue  
    </QueueUrl>  
  </GetQueueUrlResult>  
  <ResponseMetadata>  
    <RequestId>470a6f13-2ed9-4181-ad8a-2fdea142988e</RequestId>  
  </ResponseMetadata>  
</GetQueueUrlResponse>
```

ListDeadLetterSourceQueues

Returns a list of your queues that have the RedrivePolicy queue attribute configured with a dead letter queue.

For more information about using dead letter queues, see [Using Amazon SQS Dead Letter Queues](#).

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueUrl

The queue URL of a dead letter queue.

Type: String

Required: Yes

Response Elements

The following element is returned.

QueueUrls

A list of source queue URLs that have the RedrivePolicy queue attribute configured with a dead letter queue.

Type: String list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

QueueDoesNotExist

The queue referred to does not exist.

HTTP Status Code: 400

Examples

The following example Query request returns a list of dead letter source queues. In this example only one source queue, MySourceQueue, was configured with a dead letter queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
?Action=ListDeadLetterSourceQueues
&Version=2012-11-05
&Expires=2014-12-12T22%3A52%3A43PST
&AUTHPARAMS
```

Sample Response

```
<ListDeadLetterSourceQueuesResponse xmlns="http://queue.amazonaws.com/doc/2012-11-05/">
  <ListDeadLetterSourceQueuesResult>
    <QueueUrl>http://sqs.us-east-1.amazonaws.com/123456789012/MySourceQueue</QueueUrl>
  </ListDeadLetterSourceQueuesResult>
  <ResponseMetadata>
    <RequestId>8ffb921f-b85e-53d9-abcf-d8d0057f38fc</RequestId>
  </ResponseMetadata>
</ListDeadLetterSourceQueuesResponse>
```

ListQueues

Returns a list of your queues. The maximum number of queues that can be returned is 1000. If you specify a value for the optional `QueueNamePrefix` parameter, only queues with a name beginning with the specified value are returned.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueNamePrefix

A string to use for filtering the list results. Only those queues whose name begins with the specified string are returned.

Type: String

Required: No

Response Elements

The following element is returned.

QueueUrls

A list of queue URLs, up to 1000 entries.

Type: String list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

Examples

The following example Query request returns the queues whose names begin with the letter "t". How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/  
?Action=ListQueues  
&QueueNamePrefix=t  
&Version=2012-11-05  
&Expires=2014-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```


Sample Response

```
<ListQueuesResponse>
  <ListQueuesResult>
    <QueueUrl>
      http://sqs.us-east-1.amazonaws.com/123456789012/testQueue
    </QueueUrl>
  </ListQueuesResult>
  <ResponseMetadata>
    <RequestId>
      725275ae-0b9b-4762-b238-436d7c65a1ac
    </RequestId>
  </ResponseMetadata>
</ListQueuesResponse>
```

PurgeQueue

Deletes the messages in a queue specified by the **queue URL**.

Important

When you use the `PurgeQueue` API, the deleted messages in the queue cannot be retrieved.

When you purge a queue, the message deletion process takes up to 60 seconds. All messages sent to the queue before calling `PurgeQueue` will be deleted; messages sent to the queue while it is being purged may be deleted. While the queue is being purged, messages sent to the queue before `PurgeQueue` was called may be received, but will be deleted within the next minute.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

QueueUrl

The queue URL of the queue to delete the messages from when using the `PurgeQueue` API.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

PurgeQueueInProgress

Indicates that the specified queue previously received a `PurgeQueue` request within the last 60 seconds, the time it can take to delete the messages in the queue.

HTTP Status Code: 403

QueueDoesNotExist

The queue referred to does not exist.

HTTP Status Code: 400

Examples

The following example Query request purges a queue named `testQueue`. How you structure the `AUTHPARAMS` depends on how you are signing your API request. For information on `AUTHPARAMS` in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=PurgeQueue  
&Version=2012-11-05  
&Expires=2014-12-12T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<PurgeQueueResponse>
  <ResponseMetadata>
    <RequestId>
      6fde8d1e-52cd-4581-8cd9-c512f4c64223
    </RequestId>
  </ResponseMetadata>
</PurgeQueueResponse>
```

ReceiveMessage

Retrieves one or more messages, with a maximum limit of 10 messages, from the specified queue. Long poll support is enabled by using the `WaitTimeSeconds` parameter. For more information, see [Amazon SQS Long Poll](#) in the *Amazon SQS Developer Guide*.

Short poll is the default behavior where a weighted random set of machines is sampled on a `ReceiveMessage` call. This means only the messages on the sampled machines are returned. If the number of messages in the queue is small (less than 1000), it is likely you will get fewer messages than you requested per `ReceiveMessage` call. If the number of messages in the queue is extremely small, you might not receive any messages in a particular `ReceiveMessage` response; in which case you should repeat the request.

For each message returned, the response includes the following:

- Message body
- MD5 digest of the message body. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.
- Message ID you received when you sent the message to the queue.
- Receipt handle.
- Message attributes.
- MD5 digest of the message attributes.

The receipt handle is the identifier you must provide when deleting the message. For more information, see [Queue and Message Identifiers](#) in the *Amazon SQS Developer Guide*.

You can provide the `VisibilityTimeout` parameter in your request, which will be applied to the messages that Amazon SQS returns in the response. If you do not include the parameter, the overall visibility timeout for the queue is used for the returned messages. For more information, see [Visibility Timeout](#) in the *Amazon SQS Developer Guide*.

Note

Going forward, new attributes might be added. If you are writing code that calls this action, we recommend that you structure your code so that it can handle new attributes gracefully.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

AttributeName.member.N

A list of attributes that need to be returned along with each message.

The following lists the names and descriptions of the attributes that can be returned:

- `All` - returns all values.
- `ApproximateFirstReceiveTimestamp` - returns the time when the message was first received from the queue (epoch time in milliseconds).
- `ApproximateReceiveCount` - returns the number of times a message has been received from the queue but not deleted.
- `SenderId` - returns the AWS account number (or the IP address, if anonymous access is allowed) of the sender.
- `SentTimestamp` - returns the time when the message was sent to the queue (epoch time in milliseconds).

Type: String list

Valid Values: Policy | VisibilityTimeout | MaximumMessageSize | MessageRetentionPeriod | ApproximateNumberOfMessages | ApproximateNumberOfMessagesNotVisible | CreatedTimestamp | LastModifiedTimestamp | QueueArn | ApproximateNumberOfMessagesDelayed | DelaySeconds | ReceiveMessageWaitTimeSeconds | RedrivePolicy

Required: No

MaxNumberOfMessages

The maximum number of messages to return. Amazon SQS never returns more messages than this value but may return fewer. Values can be from 1 to 10. Default is 1.

All of the messages are not necessarily returned.

Type: Integer

Required: No

MessageAttributeName.member.N

The name of the message attribute, where *N* is the index. The message attribute name can contain the following characters: A-Z, a-z, 0-9, underscore (_), hyphen (-), and period (.). The name must not start or end with a period, and it should not have successive periods. The name is case sensitive and must be unique among all attribute names for the message. The name can be up to 256 characters long. The name cannot start with "AWS." or "Amazon." (or any variations in casing), because these prefixes are reserved for use by Amazon Web Services.

When using `ReceiveMessage`, you can send a list of attribute names to receive, or you can return all of the attributes by specifying "All" or ".*" in your request. You can also use "foo.*" to return all message attributes starting with the "foo" prefix.

Type: String list

Required: No

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

VisibilityTimeout

The duration (in seconds) that the received messages are hidden from subsequent retrieve requests after being retrieved by a `ReceiveMessage` request.

Type: Integer

Required: No

WaitTimeSeconds

The duration (in seconds) for which the call will wait for a message to arrive in the queue before returning. If a message is available, the call will return sooner than `WaitTimeSeconds`.

Type: Integer

Required: No

Response Elements

The following element is returned.

Messages

A list of messages.

Type: [Message \(p. 50\)](#) list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

OverLimit

The action that you requested would violate a limit. For example, `ReceiveMessage` returns this error if the maximum number of messages inflight has already been reached. [AddPermission \(p. 3\)](#) returns this error if the maximum number of permissions for the queue has already been reached.

HTTP Status Code: 403

Examples

The following example Query request receives messages from the specified queue. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=ReceiveMessage  
&MaxNumberOfMessages=5  
&VisibilityTimeout=15  
&AttributeName=All;  
&Version=2012-11-05  
&Expires=2014-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<ReceiveMessageResponse>  
  <ReceiveMessageResult>  
    <Message>  
      <MessageId>  
        5fea7756-0ea4-451a-a703-a558b933e274  
      </MessageId>  
      <ReceiptHandle>  
        MbZj6wDWli+JvwwJaBV+3dcjk2YW2vA3+STFF1jTM8tJJg6HRG6PYSasuWXPJB+Cw  
        Lj1FjgXUv1uSj1gUPAWV66FU/WeR4mq2OKpEGYWbnLmpRCJVAyeMjeU5ZBdtcQ+QE  
        auMZc8ZRv37sIW2iJKq3M9MFx1YvV11A2x/KSbkJ0=  
      </ReceiptHandle>  
      <MD5OfBody>
```

Amazon Simple Queue Service API Reference Examples

```
    fafb00f5732ab283681e124bf8747ed1
  </MD5OfBody>
  <Body>This is a test message</Body>
  <Attribute>
    <Name>SenderId</Name>
    <Value>195004372649</Value>
  </Attribute>
  <Attribute>
    <Name>SentTimestamp</Name>
    <Value>1238099229000</Value>
  </Attribute>
  <Attribute>
    <Name>ApproximateReceiveCount</Name>
    <Value>5</Value>
  </Attribute>
  <Attribute>
    <Name>ApproximateFirstReceiveTimestamp</Name>
    <Value>1250700979248</Value>
  </Attribute>
</Message>
</ReceiveMessageResult>
<ResponseMetadata>
  <RequestId>
    b6633655-283d-45b4-ae4-4e84e0ae6afa
  </RequestId>
</ResponseMetadata>
</ReceiveMessageResponse>
```

RemovePermission

Revokes any permissions in the queue policy that matches the specified `Label` parameter. Only the owner of the queue can remove permissions.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

Label

The identification of the permission to remove. This is the label added with the [AddPermission \(p. 3\)](#) action.

Type: String

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

Examples

The following example Query request removes the `testLabel` permission on the queue named `testQueue`. How you structure the `AUTHPARAMS` depends on how you are signing your API request. For information on `AUTHPARAMS` in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=RemovePermission  
&Label=testLabel  
&Version=2012-11-05  
&Expires=2012-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<RemovePermissionResponse>  
  <ResponseMetadata>
```


Amazon Simple Queue Service API Reference Examples

```
<RequestId>
  f8bdb362-6616-42c0-977a-ce9a8bcce3bb
</RequestId>
</ResponseMetadata>
</RemovePermissionResponse>
```

SendMessage

Delivers a message to the specified queue. With Amazon SQS, you now have the ability to send large payload messages that are up to 256KB (262,144 bytes) in size. To send large payloads, you must use an AWS SDK that supports SigV4 signing. To verify whether SigV4 is supported for an AWS SDK, check the SDK release notes.

Important

The following list shows the characters (in Unicode) allowed in your message, according to the W3C XML specification. For more information, go to <http://www.w3.org/TR/REC-xml/#charsets>. If you send any characters not included in the list, your request will be rejected. #x9 | #xA | #xD | [#x20 to #xD7FF] | [#xE000 to #xFFFD] | [#x10000 to #x10FFFF]

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

DelaySeconds

The number of seconds (0 to 900 - 15 minutes) to delay a specific message. Messages with a positive `DelaySeconds` value become available for processing after the delay time is finished. If you don't specify a value, the default value for the queue applies.

Type: Integer

Required: No

MessageAttribute.entry.N

Each message attribute consists of a Name, Type, and Value. For more information, see [Message Attribute Items](#).

Type: String to [MessageAttributeValue \(p. 51\)](#) map

Required: No

MessageBody

The message to send. String maximum 256 KB in size. For a list of allowed characters, see the preceding important note.

Type: String

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Response Elements

The following elements are returned.

MD5OfMessageAttributes

An MD5 digest of the non-URL-encoded message attribute string. This can be used to verify that Amazon SQS received the message correctly. Amazon SQS first URL decodes the message before creating the MD5 digest. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.

Type: String

MD5OfMessageBody

An MD5 digest of the non-URL-encoded message body string. This can be used to verify that Amazon SQS received the message correctly. Amazon SQS first URL decodes the message before creating the MD5 digest. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.

Type: String

MessageId

An element containing the message ID of the message sent to the queue. For more information, see [Queue and Message Identifiers](#) in the *Amazon SQS Developer Guide*.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 56).

InvalidMessageContents

The message contains characters outside the allowed set.

HTTP Status Code: 400

UnsupportedOperation

Error code 400. Unsupported operation.

HTTP Status Code: 400

Examples

The following example `SendMessage` request sends a message containing "This is a test message" to the queue. You must URL encode the entire URL; however, we've URL encoded only the message body to make the example easier for you to read. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=SendMessage  
&MessageBody=This+is+a+test+message  
&MessageAttribute.1.Name=test_attribute_name_1  
&MessageAttribute.1.Value.StringValue=test_attribute_value_1  
&MessageAttribute.1.Value.DataType=String  
&MessageAttribute.2.Name=test_attribute_name_2  
&MessageAttribute.2.Value.StringValue=test_attribute_value_2  
&MessageAttribute.2.Value.DataType=String  
&Version=2012-11-05  
&Expires=2014-05-05T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<SendMessageResponse>
  <SendMessageResult>
    <MD5OfMessageBody>
      fafb00f5732ab283681e124bf8747ed1
    </MD5OfMessageBody>
    <MD5OfMessageAttributes>
      3ae8f24a165a8cedc005670c81a27295
    </MD5OfMessageAttributes>
    <MessageId>
      5fea7756-0ea4-451a-a703-a558b933e274
    </MessageId>
  </SendMessageResult>
  <ResponseMetadata>
    <RequestId>
      27daac76-34dd-47df-bd01-1f6e873584a0
    </RequestId>
  </ResponseMetadata>
</SendMessageResponse>
```

SendMessageBatch

Delivers up to ten messages to the specified queue. This is a batch version of [SendMessage \(p. 38\)](#). The result of the send action on each message is reported individually in the response. The maximum allowed individual message size is 256 KB (262,144 bytes).

The maximum total payload size (i.e., the sum of all a batch's individual message lengths) is also 256 KB (262,144 bytes).

If the `DelaySeconds` parameter is not specified for an entry, the default for the queue is used.

Important

The following list shows the characters (in Unicode) that are allowed in your message, according to the W3C XML specification. For more information, go to <http://www.faqs.org/rfcs/rfc1321.html>. If you send any characters that are not included in the list, your request will be rejected. `#x9 | #xA | #xD | [#x20 to #xD7FF] | [#xE000 to #xFFFF] | [#x10000 to #x10FFFF]`

Important

Because the batch request can result in a combination of successful and unsuccessful actions, you should check for batch errors even when the call returns an HTTP status code of 200.

Note

Some API actions take lists of parameters. These lists are specified using the `param.n` notation. Values of `n` are integers starting from 1. For example, a parameter list with two elements looks like this:

```
&Attribute.1=this
```

```
&Attribute.2=that
```

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

SendMessageBatchRequestEntry.member.N

A list of [SendMessageBatchRequestEntry \(p. 52\)](#) items.

Type: [SendMessageBatchRequestEntry \(p. 52\)](#) list

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Response Elements

The following elements are returned.

Failed

A list of [BatchResultErrorEntry \(p. 47\)](#) items with the error detail about each message that could not be enqueued.

Type: [BatchResultErrorEntry \(p. 47\)](#) list

Successful

A list of [SendMessageBatchResultEntry \(p. 53\)](#) items.

Type: [SendMessageBatchResultEntry \(p. 53\)](#) list

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

BatchEntryIdsNotDistinct

Two or more batch entries have the same `Id` in the request.

HTTP Status Code: 400

BatchRequestTooLong

The length of all the messages put together is more than the limit.

HTTP Status Code: 400

EmptyBatchRequest

Batch request does not contain an entry.

HTTP Status Code: 400

InvalidBatchEntryId

The `Id` of a batch entry in a batch request does not abide by the specification.

HTTP Status Code: 400

TooManyEntriesInBatchRequest

Batch request contains more number of entries than permissible.

HTTP Status Code: 400

UnsupportedOperation

Error code 400. Unsupported operation.

HTTP Status Code: 400

Examples

The following example `SendMessageBatch` request sends two messages to the queue. You must URL encode the entire URL; however, we've URL encoded only the message body to make the example easier for you to read. How you structure the `AUTHPARAMS` depends on how you are signing your API request. For information on `AUTHPARAMS` in Signature Version 4, go to [Examples of Signed Signature Version 4 Requests](#).

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=SendMessageBatch  
&SendMessageBatchRequestEntry.1.Id=test_msg_001  
&SendMessageBatchRequestEntry.1.MessageBody=test%20message%20body%201  
&SendMessageBatchRequestEntry.2.Id=test_msg_002  
&SendMessageBatchRequestEntry.2.MessageBody=test%20message%20body%202  
&SendMessageBatchRequestEntry.2.DelaySeconds=60  
&SendMessageBatchRequestEntry.2.MessageAttribute.1.Name=test_attribute_name_1
```

```
&SendMessageBatchRequestEntry.2.MessageAttribute.1.Value.StringValue=test_at  
tribute_value_1  
&SendMessageBatchRequestEntry.2.MessageAttribute.1.Value.DataType=String  
&Version=2012-11-05  
&Expires=2014-05-05T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<SendMessageBatchResponse>  
<SendMessageBatchResult>  
  <SendMessageBatchResultEntry>  
    <Id>test_msg_001</Id>  
    <MessageId>0a5231c7-8bff-4955-be2e-8dc7c50a25fa</MessageId>  
    <MD5OfMessageBody>0e024d309850c78cba5eabbef77cae71</MD5OfMessageBody>  
  </SendMessageBatchResultEntry>  
  <SendMessageBatchResultEntry>  
    <Id>test_msg_002</Id>  
    <MessageId>15eeled3-87e7-40c1-bdaa-2e49968ea7e9</MessageId>  
    <MD5OfMessageBody>7fb8146a82f95e0af155278f406862c2</MD5OfMessageBody>  
    <MD5OfMessageAttributes>295c5fa15a51aae6884d1d7c1d99ca50</MD5OfMessageAt  
tributes>  
  </SendMessageBatchResultEntry>  
</SendMessageBatchResult>  
<ResponseMetadata>  
  <RequestId>calad5d0-8271-408b-8d0f-1351bf547e74</RequestId>  
</ResponseMetadata>  
</SendMessageBatchResponse>
```

SetQueueAttributes

Sets the value of one or more queue attributes. When you change a queue's attributes, the change can take up to 60 seconds for most of the attributes to propagate throughout the SQS system. Changes made to the `MessageRetentionPeriod` attribute can take up to 15 minutes.

Note

Going forward, new attributes might be added. If you are writing code that calls this action, we recommend that you structure your code so that it can handle new attributes gracefully.

Request Parameters

For information about the common parameters that all actions use, see [Common Parameters \(p. 54\)](#).

Attribute.entry.N

A map of attributes to set.

The following lists the names, descriptions, and values of the special request parameters the `SetQueueAttributes` action uses:

- `DelaySeconds` - The time in seconds that the delivery of all messages in the queue will be delayed. An integer from 0 to 900 (15 minutes). The default for this attribute is 0 (zero).
- `MaximumMessageSize` - The limit of how many bytes a message can contain before Amazon SQS rejects it. An integer from 1024 bytes (1 KiB) up to 262144 bytes (256 KiB). The default for this attribute is 262144 (256 KiB).
- `MessageRetentionPeriod` - The number of seconds Amazon SQS retains a message. Integer representing seconds, from 60 (1 minute) to 1209600 (14 days). The default for this attribute is 345600 (4 days).
- `Policy` - The queue's policy. A valid AWS policy. For more information about policy structure, see [Overview of AWS IAM Policies](#) in the *Amazon IAM User Guide*.
- `ReceiveMessageWaitTimeSeconds` - The time for which a `ReceiveMessage` call will wait for a message to arrive. An integer from 0 to 20 (seconds). The default for this attribute is 0.
- `VisibilityTimeout` - The visibility timeout for the queue. An integer from 0 to 43200 (12 hours). The default for this attribute is 30. For more information about visibility timeout, see [Visibility Timeout](#) in the *Amazon SQS Developer Guide*.
- `RedrivePolicy` - The parameters for dead letter queue functionality of the source queue. For more information about `RedrivePolicy` and dead letter queues, see [Using Amazon SQS Dead Letter Queues](#) in the *Amazon SQS Developer Guide*.

Type: String to String map

Valid Map Keys: `Policy` | `VisibilityTimeout` | `MaximumMessageSize` | `MessageRetentionPeriod` | `ApproximateNumberOfMessages` | `ApproximateNumberOfMessagesNotVisible` | `CreatedTimestamp` | `LastModifiedTimestamp` | `QueueArn` | `ApproximateNumberOfMessagesDelayed` | `DelaySeconds` | `ReceiveMessageWaitTimeSeconds` | `RedrivePolicy`

Required: Yes

QueueUrl

The URL of the Amazon SQS queue to take action on.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 56\)](#).

InvalidAttributeName

The attribute referred to does not exist.

HTTP Status Code: 400

Examples

The following example Query request sets a policy that gives all users ReceiveMessage permission for the queue named testQueue. For more examples of policies, see Amazon SQS Policy Examples in the Amazon SQS Developer Guide. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to Examples of Signed Signature Version 4 Requests.

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=SetQueueAttributes  
&Version=2012-11-05  
&Attribute.Name=Policy  
&Attribute.Value=%7B%22Version%22%3A%222008-10-17%22%2C%22Id%22%3A%222F123456789012%2FtestQueue%2FSQSDefaultPolicy%22%2C%22Statement%22%3A%5B%7B%22Sid%22%3A%22Queue1ReceiveMessage%22%2C%22Effect%22%3A%22Allow%22%2C%22Principal%22%3A%7B%22AWS%22%3A%22%2A%22%7D%2C%22Action%22%3A%22SQS%3AReceiveMessage%22%2C%22Resource%22%3A%22arn%3Aaws%3Aaws%3Asqs%3Aus%2Deast%2D1%3A123456789012%3AtestQueue%22%7D%5D%7D  
&Timestamp=2009-05-06T16%3A57%3A31.000Z  
&AUTHPARAMS
```

The following example Query request sets the visibility timeout to 35 seconds for the queue named testQueue. Note: There is a 120,000 limit for the number of inflight messages per queue. Messages are inflight after they have been received by the queue, but have not yet been deleted from the queue. If you reach the 120,000 limit, you will receive an OverLimit error message from Amazon SQS. To help avoid reaching the limit, you should delete the messages from the queue after they have been processed. You can also increase the number of queues you use to process the messages. How you structure the AUTHPARAMS depends on how you are signing your API request. For information on AUTHPARAMS in Signature Version 4, go to Examples of Signed Signature Version 4 Requests.

Sample Request

```
http://sqs.us-east-1.amazonaws.com/123456789012/testQueue/  
?Action=SetQueueAttributes
```

```
&Attribute.Name=VisibilityTimeout  
&Attribute.Value=35  
&Version=2012-11-05  
&Expires=2014-04-18T22%3A52%3A43PST  
&AUTHPARAMS
```

Sample Response

```
<SetQueueAttributesResponse>  
  <ResponseMetadata>  
    <RequestId>  
      e5cca473-4fc0-4198-a451-8abb94d02c75  
    </RequestId>  
  </ResponseMetadata>  
</SetQueueAttributesResponse>
```

Data Types

The Amazon Simple Queue Service API contains several data types that various actions use. This section describes each data type in detail.

Note

The order of each element in the response is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- [BatchResultErrorEntry](#) (p. 47)
- [ChangeMessageVisibilityBatchRequestEntry](#) (p. 48)
- [ChangeMessageVisibilityBatchResultEntry](#) (p. 49)
- [DeleteMessageBatchRequestEntry](#) (p. 49)
- [DeleteMessageBatchResultEntry](#) (p. 49)
- [Message](#) (p. 50)
- [MessageAttributeValue](#) (p. 51)
- [SendMessageBatchRequestEntry](#) (p. 52)
- [SendMessageBatchResultEntry](#) (p. 53)

BatchResultErrorEntry

Description

This is used in the responses of batch API to give a detailed description of the result of an action on each entry in the request.

Contents

Code

An error code representing why the action failed on this entry.

Type: String

Required: Yes

Id

The id of an entry in a batch request.

Type: String

Required: Yes

Message

A message explaining why the action failed on this entry.

Type: String

Required: No

SenderFault

Whether the error happened due to the sender's fault.

Type: Boolean

Required: Yes

ChangeMessageVisibilityBatchRequestEntry

Description

Encloses a receipt handle and an entry id for each message in [ChangeMessageVisibilityBatch](#) (p. 7).

Important

All of the following parameters are list parameters that must be prefixed with `ChangeMessageVisibilityBatchRequestEntry.n`, where `n` is an integer value starting with 1. For example, a parameter list for this action might look like this:

```
&ChangeMessageVisibilityBatchRequestEntry.1.Id=change_visibility_msg_2
```

```
&ChangeMessageVisibilityBatchRequestEntry.1.ReceiptHandle=Your_Receipt_Handle
```

```
&ChangeMessageVisibilityBatchRequestEntry.1.VisibilityTimeout=45
```

Contents

Id

An identifier for this particular receipt handle. This is used to communicate the result. Note that the Ids of a batch request need to be unique within the request.

Type: String

Required: Yes

ReceiptHandle

A receipt handle.

Type: String

Required: Yes

VisibilityTimeout

The new value (in seconds) for the message's visibility timeout.

Type: Integer

Required: No

ChangeMessageVisibilityBatchResultEntry

Description

Encloses the id of an entry in [ChangeMessageVisibilityBatch](#) (p. 7).

Contents

Id

Represents a message whose visibility timeout has been changed successfully.

Type: String

Required: Yes

DeleteMessageBatchRequestEntry

Description

Encloses a receipt handle and an identifier for it.

Contents

Id

An identifier for this particular receipt handle. This is used to communicate the result. Note that the Ids of a batch request need to be unique within the request.

Type: String

Required: Yes

ReceiptHandle

A receipt handle.

Type: String

Required: Yes

DeleteMessageBatchResultEntry

Description

Encloses the id an entry in [DeleteMessageBatch](#) (p. 15).

Contents

Id

Represents a successfully deleted message.

Type: String

Required: Yes

Message

Description

An Amazon SQS message.

Contents

Attributes

SenderId, SentTimestamp, ApproximateReceiveCount, and/or ApproximateFirstReceiveTimestamp. SentTimestamp and ApproximateFirstReceiveTimestamp are each returned as an integer representing the [epoch time](#) in milliseconds.

Type: String to String map

Valid Map Keys: Policy | VisibilityTimeout | MaximumMessageSize | MessageRetentionPeriod | ApproximateNumberOfMessages | ApproximateNumberOfMessagesNotVisible | CreatedTimestamp | LastModifiedTimestamp | QueueArn | ApproximateNumberOfMessagesDelayed | DelaySeconds | ReceiveMessageWaitTimeSeconds | RedrivePolicy

Required: No

Body

The message's contents (not URL-encoded).

Type: String

Required: No

MD5OfBody

An MD5 digest of the non-URL-encoded message body string.

Type: String

Required: No

MD5OfMessageAttributes

An MD5 digest of the non-URL-encoded message attribute string. This can be used to verify that Amazon SQS received the message correctly. Amazon SQS first URL decodes the message before creating the MD5 digest. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.

Type: String

Required: No

MessageAttributes

Each message attribute consists of a Name, Type, and Value. For more information, see [Message Attribute Items](#).

Type: String to [MessageAttributeValue](#) (p. 51) map

Required: No

MessageId

A unique identifier for the message. Message IDs are considered unique across all AWS accounts for an extended period of time.

Type: String

Required: No

ReceiptHandle

An identifier associated with the act of receiving the message. A new receipt handle is returned every time you receive a message. When deleting a message, you provide the last received receipt handle to delete the message.

Type: String

Required: No

MessageAttributeValue

Description

The user-specified message attribute value. For string data types, the value attribute has the same restrictions on the content as the message body. For more information, see [SendMessage](#).

Name, type, and value must not be empty or null. In addition, the message body should not be empty or null. All parts of the message attribute, including name, type, and value, are included in the message size restriction, which is currently 256 KB (262,144 bytes).

Contents

BinaryListValues

Not implemented. Reserved for future use.

Type: Blob list

Required: No

BinaryValue

Binary type attributes can store any binary data, for example, compressed data, encrypted data, or images.

Type: Blob

Required: No

DataType

Amazon SQS supports the following logical data types: String, Number, and Binary. In addition, you can append your own custom labels. For more information, see [Message Attribute Data Types](#).

Type: String

Required: Yes

StringListValues

Not implemented. Reserved for future use.

Type: String list

Required: No

StringValue

Strings are Unicode with UTF8 binary encoding. For a list of code values, see http://en.wikipedia.org/wiki/ASCII#ASCII_printable_characters.

Type: String

Required: No

SendMessageBatchRequestEntry

Description

Contains the details of a single Amazon SQS message along with a `Id`.

Contents

DelaySeconds

The number of seconds for which the message has to be delayed.

Type: Integer

Required: No

Id

An identifier for the message in this batch. This is used to communicate the result. Note that the `Ids` of a batch request need to be unique within the request.

Type: String

Required: Yes

MessageAttributes

Each message attribute consists of a Name, Type, and Value. For more information, see [Message Attribute Items](#).

Type: String to [MessageAttributeValue](#) (p. 51) map

Required: No

MessageBody

Body of the message.

Type: String

Required: Yes

SendMessageBatchResultEntry

Description

Encloses a message ID for successfully enqueued message of a [SendMessageBatch](#) (p. 41).

Contents

Id

An identifier for the message in this batch.

Type: String

Required: Yes

MD5OfMessageAttributes

An MD5 digest of the non-URL-encoded message attribute string. This can be used to verify that Amazon SQS received the message batch correctly. Amazon SQS first URL decodes the message before creating the MD5 digest. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.

Type: String

Required: No

MD5OfMessageBody

An MD5 digest of the non-URL-encoded message body string. This can be used to verify that Amazon SQS received the message correctly. Amazon SQS first URL decodes the message before creating the MD5 digest. For information about MD5, go to <http://www.faqs.org/rfcs/rfc1321.html>.

Type: String

Required: Yes

MessageId

An identifier for the message.

Type: String

Required: Yes

Common Parameters

This section lists the request parameters that all actions use. Any action-specific parameters are listed in the topic for the action.

Action

The action to be performed.

Default: None

Type: string

Required: Yes

AuthParams

The parameters that are required to authenticate a Conditional request. Contains:

- AWSAccessKeyID
- SignatureVersion
- Timestamp
- Signature

Default: None

Required: Conditional

AWSAccessKeyID

The access key ID that corresponds to the secret access key that you used to sign the request.

Default: None

Type: string

Required: Yes

Expires

The date and time when the request signature expires, expressed in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard.

Condition: Requests must include either *Timestamp* or *Expires*, but not both.

Default: None

Type: string

Required: Conditional

SecurityToken

The temporary security token that was obtained through a call to AWS Security Token Service. For a list of services that support AWS Security Token Service, go to [Using Temporary Security Credentials to Access AWS](#) in **Using Temporary Security Credentials**.

Default: None

Type: string

Required: No

Signature

The digital signature that you created for the request. For information about generating a signature, go to the service's developer documentation.

Default: None

Type: string

Required: Yes

SignatureMethod

The hash algorithm that you used to create the request signature.

Default: None

Type: string

Valid Values: HmacSHA256 | HmacSHA1

Required: Yes

SignatureVersion

The signature version you use to sign the request. Set this to the value that is recommended for your service.

Default: None

Type: string

Required: Yes

Timestamp

The date and time when the request was signed, expressed in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard.

Condition: Requests must include either *Timestamp* or *Expires*, but not both.

Default: None

Type: string

Required: Conditional

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Default: None

Type: string

Required: Yes

Common Errors

This section lists the common errors that all actions return. Any action-specific errors are listed in the topic for the action.

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

Throttling

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400