

Package ‘paws.analytics’

August 21, 2019

Title Amazon Web Services Analytics APIs

Version 0.1.4

Description Interface to Amazon Web Services analytics APIs, including 'Elastic MapReduce' 'Hadoop' and 'Spark' big data service, 'Elasticsearch' search engine, and more <<https://aws.amazon.com/>>.

License Apache License (>= 2.0)

Imports paws.common (>= 0.2.0)

Suggests testthat

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Collate 'athena_service.R' 'athena_interfaces.R' 'athena_operations.R'
'cloudsearch_service.R' 'cloudsearch_interfaces.R'
'cloudsearch_operations.R' 'cloudsearchdomain_service.R'
'cloudsearchdomain_interfaces.R'
'cloudsearchdomain_operations.R' 'datapipeline_service.R'
'datapipeline_interfaces.R' 'datapipeline_operations.R'
'elasticsearchservice_service.R'
'elasticsearchservice_interfaces.R'
'elasticsearchservice_operations.R' 'emr_service.R'
'emr_interfaces.R' 'emr_operations.R' 'firehose_service.R'
'firehose_interfaces.R' 'firehose_operations.R'
'glue_service.R' 'glue_interfaces.R' 'glue_operations.R'
'kafka_service.R' 'kafka_interfaces.R' 'kafka_operations.R'
'kinesis_service.R' 'kinesis_interfaces.R'
'kinesis_operations.R' 'kinesisanalytics_service.R'
'kinesisanalytics_interfaces.R' 'kinesisanalytics_operations.R'
'kinesisanalyticsv2_service.R'
'kinesisanalyticsv2_interfaces.R'
'kinesisanalyticsv2_operations.R' 'mturk_service.R'
'mturk_interfaces.R' 'mturk_operations.R'
'quicksight_service.R' 'quicksight_interfaces.R'
'quicksight_operations.R'

NeedsCompilation no

Author David Kretch [aut, cre],
Adam Banker [aut],
Amazon.com, Inc. [cph]

Maintainer David Kretch <david.kretch@gmail.com>

Repository CRAN

Date/Publication 2019-08-21 10:51:12 UTC

R topics documented:

athena	2
cloudsearch	3
cloudsearchdomain	5
datapipeline	5
elasticsearchservice	7
emr	8
firehose	9
glue	10
kafka	13
kinesis	13
kinesisanalytics	15
kinesisanalyticsv2	16
mturk	17
quicksight	18
Index	20

athena	<i>Amazon Athena</i>
--------	----------------------

Description

Amazon Athena is an interactive query service that lets you use standard SQL to analyze data directly in Amazon S3. You can point Athena at your data in Amazon S3 and run ad-hoc queries and get results in seconds. Athena is serverless, so there is no infrastructure to set up or manage. You pay only for the queries you run. Athena scales automatically—executing queries in parallel—so results are fast, even with large datasets and complex queries. For more information, see [What is Amazon Athena](#) in the *Amazon Athena User Guide*.

If you connect to Athena using the JDBC driver, use version 1.1.0 of the driver or later with the Amazon Athena API. Earlier version drivers do not support the API. For more information and to download the driver, see [Accessing Amazon Athena with JDBC](#).

For code samples using the AWS SDK for Java, see [Examples and Code Samples](#) in the *Amazon Athena User Guide*.

Usage

```
athena()
```

Operations

batch_get_named_query	Returns the details of a single named query or a list of up to 50 queries, which you provide as an array.
batch_get_query_execution	Returns the details of a single query execution or a list of up to 50 query executions, which you provide as an array.
create_named_query	Creates a named query in the specified workgroup.
create_work_group	Creates a workgroup with the specified name.
delete_named_query	Deletes the named query if you have access to the workgroup in which the query was saved.
delete_work_group	Deletes the workgroup with the specified name.
get_named_query	Returns information about a single query.
get_query_execution	Returns information about a single execution of a query if you have access to the workgroup in which the query was saved.
get_query_results	Returns the results of a single query execution specified by QueryExecutionId if you have access to the workgroup in which the query was saved.
get_work_group	Returns information about the workgroup with the specified name.
list_named_queries	Provides a list of available query IDs only for queries saved in the specified workgroup.
list_query_executions	Provides a list of available query execution IDs for the queries in the specified workgroup.
list_tags_for_resource	Lists the tags associated with this workgroup.
list_work_groups	Lists available workgroups for the account.
start_query_execution	Runs the SQL query statements contained in the Query.
stop_query_execution	Stops a query execution.
tag_resource	Adds one or more tags to the resource, such as a workgroup.
untag_resource	Removes one or more tags from the workgroup resource.
update_work_group	Updates the workgroup with the specified name.

Examples

```
svc <- athena()
svc$batch_get_named_query(
  Foo = 123
)
```

cloudsearch

*Amazon CloudSearch***Description**

Amazon CloudSearch Configuration Service

You use the Amazon CloudSearch configuration service to create, configure, and manage search domains. Configuration service requests are submitted using the AWS Query protocol. AWS Query requests are HTTP or HTTPS requests submitted via HTTP GET or POST with a query parameter named Action.

The endpoint for configuration service requests is region-specific: `cloudsearch.region.amazonaws.com`. For example, `cloudsearch.us-east-1.amazonaws.com`. For a current list of supported regions and endpoints, see [Regions and Endpoints](#).

Usage

```
cloudsearch()
```

Operations

build_suggesters	Indexes the search suggestions
create_domain	Creates a new search domain
define_analysis_scheme	Configures an analysis scheme that can be applied to a text or text-array field to define lan
define_expression	Configures an Expression for the search domain
define_index_field	Configures an IndexField for the search domain
define_suggester	Configures a suggester for a domain
delete_analysis_scheme	Deletes an analysis scheme
delete_domain	Permanently deletes a search domain and all of its data
delete_expression	Removes an Expression from the search domain
delete_index_field	Removes an IndexField from the search domain
delete_suggester	Deletes a suggester
describe_analysis_schemes	Gets the analysis schemes configured for a domain
describe_availability_options	Gets the availability options configured for a domain
describe_domains	Gets information about the search domains owned by this account
describe_expressions	Gets the expressions configured for the search domain
describe_index_fields	Gets information about the index fields configured for the search domain
describe_scaling_parameters	Gets the scaling parameters configured for a domain
describe_service_access_policies	Gets information about the access policies that control access to the domain's document a
describe_suggesters	Gets the suggesters configured for a domain
index_documents	Tells the search domain to start indexing its documents using the latest indexing options
list_domain_names	Lists all search domains owned by an account
update_availability_options	Configures the availability options for a domain
update_scaling_parameters	Configures scaling parameters for a domain
update_service_access_policies	Configures the access rules that control access to the domain's document and search endp

Examples

```
svc <- cloudsearch()
svc$build_suggesters(
  Foo = 123
)
```

cloudsearchdomain	<i>Amazon CloudSearch Domain</i>
-------------------	----------------------------------

Description

You use the AmazonCloudSearch2013 API to upload documents to a search domain and search those documents.

The endpoints for submitting UploadDocuments, Search, and Suggest requests are domain-specific. To get the endpoints for your domain, use the Amazon CloudSearch configuration service DescribeDomains action. The domain endpoints are also displayed on the domain dashboard in the Amazon CloudSearch console. You submit suggest requests to the search endpoint.

For more information, see the [Amazon CloudSearch Developer Guide](#).

Usage

```
cloudsearchdomain()
```

Operations

search	Retrieves a list of documents that match the specified search criteria
suggest	Retrieves autocomplete suggestions for a partial query string
upload_documents	Posts a batch of documents to a search domain for indexing

Examples

```
svc <- cloudsearchdomain()
svc$search(
  Foo = 123
)
```

datapipeline	<i>AWS Data Pipeline</i>
--------------	--------------------------

Description

AWS Data Pipeline configures and manages a data-driven workflow called a pipeline. AWS Data Pipeline handles the details of scheduling and ensuring that data dependencies are met so that your application can focus on processing the data.

AWS Data Pipeline provides a JAR implementation of a task runner called AWS Data Pipeline Task Runner. AWS Data Pipeline Task Runner provides logic for common data management scenarios, such as performing database queries and running data analysis using Amazon Elastic MapReduce

(Amazon EMR). You can use AWS Data Pipeline Task Runner as your task runner, or you can write your own task runner to provide custom data management.

AWS Data Pipeline implements two main sets of functionality. Use the first set to create a pipeline and define data sources, schedules, dependencies, and the transforms to be performed on the data. Use the second set in your task runner application to receive the next task ready for processing. The logic for performing the task, such as querying the data, running data analysis, or converting the data from one format to another, is contained within the task runner. The task runner performs the task assigned to it by the web service, reporting progress to the web service as it does so. When the task is done, the task runner reports the final success or failure of the task to the web service.

Usage

```
datapipeline()
```

Operations

activate_pipeline	Validates the specified pipeline and starts processing pipeline tasks
add_tags	Adds or modifies tags for the specified pipeline
create_pipeline	Creates a new, empty pipeline
deactivate_pipeline	Deactivates the specified running pipeline
delete_pipeline	Deletes a pipeline, its pipeline definition, and its run history
describe_objects	Gets the object definitions for a set of objects associated with the pipeline
describe_pipelines	Retrieves metadata about one or more pipelines
evaluate_expression	Task runners call EvaluateExpression to evaluate a string in the context of the specified object
get_pipeline_definition	Gets the definition of the specified pipeline
list_pipelines	Lists the pipeline identifiers for all active pipelines that you have permission to access
poll_for_task	Task runners call PollForTask to receive a task to perform from AWS Data Pipeline
put_pipeline_definition	Adds tasks, schedules, and preconditions to the specified pipeline
query_objects	Queries the specified pipeline for the names of objects that match the specified set of conditions
remove_tags	Removes existing tags from the specified pipeline
report_task_progress	Task runners call ReportTaskProgress when assigned a task to acknowledge that it has the task
report_task_runner_heartbeat	Task runners call ReportTaskRunnerHeartbeat every 15 minutes to indicate that they are operating
set_status	Requests that the status of the specified physical or logical pipeline objects be updated in the service
set_task_status	Task runners call SetTaskStatus to notify AWS Data Pipeline that a task is completed and provide progress
validate_pipeline_definition	Validates the specified pipeline definition to ensure that it is well formed and can be run without errors

Examples

```
svc <- datapipeline()
svc$activate_pipeline(
  Foo = 123
)
```

Description

Amazon Elasticsearch Configuration Service

Use the Amazon Elasticsearch configuration API to create, configure, and manage Elasticsearch domains.

The endpoint for configuration service requests is region-specific: `es.region.amazonaws.com`. For example, `es.us-east-1.amazonaws.com`. For a current list of supported regions and endpoints, see [Regions and Endpoints](#).

Usage

```
elasticsearchservice()
```

Operations

add_tags	Attaches tags to an existing Elasticsearch domain
cancel_elasticsearch_service_software_update	Cancels a scheduled service software update for an Amazon ES domain
create_elasticsearch_domain	Creates a new Elasticsearch domain
delete_elasticsearch_domain	Permanently deletes the specified Elasticsearch domain and all of its data
delete_elasticsearch_service_role	Deletes the service-linked role that Elasticsearch Service uses to manage domains
describe_elasticsearch_domain	Returns domain configuration information about the specified Elasticsearch domain
describe_elasticsearch_domain_config	Provides cluster configuration information about the specified Elasticsearch domain
describe_elasticsearch_domains	Returns domain configuration information about the specified Elasticsearch domains
describe_elasticsearch_instance_type_limits	Describe Elasticsearch Limits for a given InstanceType and Elasticsearch version
describe_reserved_elasticsearch_instance_offerings	Lists available reserved Elasticsearch instance offerings
describe_reserved_elasticsearch_instances	Returns information about reserved Elasticsearch instances for this account
get_compatible_elasticsearch_versions	Returns a list of upgrade compatible Elasticsearch versions
get_upgrade_history	Retrieves the complete history of the last 10 upgrades that were performed
get_upgrade_status	Retrieves the latest status of the last upgrade or upgrade eligibility check
list_domain_names	Returns the name of all Elasticsearch domains owned by the current user
list_elasticsearch_instance_types	List all Elasticsearch instance types that are supported for given Elasticsearch version
list_elasticsearch_versions	List all supported Elasticsearch versions
list_tags	Returns all tags for the given Elasticsearch domain
purchase_reserved_elasticsearch_instance_offering	Allows you to purchase reserved Elasticsearch instances
remove_tags	Removes the specified set of tags from the specified Elasticsearch domain
start_elasticsearch_service_software_update	Schedules a service software update for an Amazon ES domain
update_elasticsearch_domain_config	Modifies the cluster configuration of the specified Elasticsearch domain
upgrade_elasticsearch_domain	Allows you to either upgrade your domain or perform an Upgrade eligibility check

Examples

```
svc <- elasticsearchservice()
svc$add_tags(
  Foo = 123
)
```

 emr

Amazon Elastic MapReduce

Description

Amazon EMR is a web service that makes it easy to process large amounts of data efficiently. Amazon EMR uses Hadoop processing combined with several AWS products to do tasks such as web indexing, data mining, log file analysis, machine learning, scientific simulation, and data warehousing.

Usage

```
emr()
```

Operations

add_instance_fleet	Adds an instance fleet to a running cluster
add_instance_groups	Adds one or more instance groups to a running cluster
add_job_flow_steps	AddJobFlowSteps adds new steps to a running cluster
add_tags	Adds tags to an Amazon EMR resource
cancel_steps	Cancels a pending step or steps in a running cluster
create_security_configuration	Creates a security configuration, which is stored in the service and can be specified when a
delete_security_configuration	Deletes a security configuration
describe_cluster	Provides cluster-level details including status, hardware and software configuration, VPC s
describe_job_flows	This API is deprecated and will eventually be removed
describe_security_configuration	Provides the details of a security configuration by returning the configuration JSON
describe_step	Provides more detail about the cluster step
list_bootstrap_actions	Provides information about the bootstrap actions associated with a cluster
list_clusters	Provides the status of all clusters visible to this AWS account
list_instance_fleets	Lists all available details about the instance fleets in a cluster
list_instance_groups	Provides all available details about the instance groups in a cluster
list_instances	Provides information for all active EC2 instances and EC2 instances terminated in the last 3
list_security_configurations	Lists all the security configurations visible to this account, providing their creation dates an
list_steps	Provides a list of steps for the cluster in reverse order unless you specify stepIds with the re
modify_instance_fleet	Modifies the target On-Demand and target Spot capacities for the instance fleet with the spe
modify_instance_groups	ModifyInstanceGroups modifies the number of nodes and configuration settings of an insta
put_auto_scaling_policy	Creates or updates an automatic scaling policy for a core instance group or task instance gr
remove_auto_scaling_policy	Removes an automatic scaling policy from a specified instance group within an EMR cluste
remove_tags	Removes tags from an Amazon EMR resource

run_job_flow	RunJobFlow creates and starts running a new cluster (job flow)
set_termination_protection	SetTerminationProtection locks a cluster (job flow) so the EC2 instances in the cluster cannot be terminated
set_visible_to_all_users	Sets whether all AWS Identity and Access Management (IAM) users under your account can access the cluster
terminate_job_flows	TerminateJobFlows shuts a list of clusters (job flows) down

Examples

```
svc <- emr()
svc$add_instance_fleet(
  Foo = 123
)
```

firehose

Amazon Kinesis Firehose

Description

Amazon Kinesis Data Firehose API Reference

Amazon Kinesis Data Firehose is a fully managed service that delivers real-time streaming data to destinations such as Amazon Simple Storage Service (Amazon S3), Amazon Elasticsearch Service (Amazon ES), Amazon Redshift, and Splunk.

Usage

```
firehose()
```

Operations

create_delivery_stream	Creates a Kinesis Data Firehose delivery stream
delete_delivery_stream	Deletes a delivery stream and its data
describe_delivery_stream	Describes the specified delivery stream and gets the status
list_delivery_streams	Lists your delivery streams in alphabetical order of their names
list_tags_for_delivery_stream	Lists the tags for the specified delivery stream
put_record	Writes a single data record into an Amazon Kinesis Data Firehose delivery stream
put_record_batch	Writes multiple data records into a delivery stream in a single call, which can achieve high throughput
start_delivery_stream_encryption	Enables server-side encryption (SSE) for the delivery stream
stop_delivery_stream_encryption	Disables server-side encryption (SSE) for the delivery stream
tag_delivery_stream	Adds or updates tags for the specified delivery stream
untag_delivery_stream	Removes tags from the specified delivery stream
update_destination	Updates the specified destination of the specified delivery stream

Examples

```

svc <- firehose()
svc$create_delivery_stream(
  Foo = 123
)

```

glue

*AWS Glue***Description**

Defines the public endpoint for the AWS Glue service.

Usage

```
glue()
```

Operations

batch_create_partition	Creates one or more partitions in a batch operation
batch_delete_connection	Deletes a list of connection definitions from the Data Catalog
batch_delete_partition	Deletes one or more partitions in a batch operation
batch_delete_table	Deletes multiple tables at once
batch_delete_table_version	Deletes a specified batch of versions of a table
batch_get_crawlers	Returns a list of resource metadata for a given list of crawler names
batch_get_dev_endpoints	Returns a list of resource metadata for a given list of DevEndpoint names
batch_get_jobs	Returns a list of resource metadata for a given list of job names
batch_get_partition	Retrieves partitions in a batch request
batch_get_triggers	Returns a list of resource metadata for a given list of trigger names
batch_get_workflows	Returns a list of resource metadata for a given list of workflow names
batch_stop_job_run	Stops one or more job runs for a specified job definition
create_classifier	Creates a classifier in the user's account
create_connection	Creates a connection definition in the Data Catalog
create_crawler	Creates a new crawler with specified targets, role, configuration, and optional schedule
create_database	Creates a new database in a Data Catalog
create_dev_endpoint	Creates a new DevEndpoint
create_job	Creates a new job definition
create_partition	Creates a new partition
create_script	Transforms a directed acyclic graph (DAG) into code
create_security_configuration	Creates a new security configuration
create_table	Creates a new table definition in the Data Catalog
create_trigger	Creates a new trigger
create_user_defined_function	Creates a new function definition in the Data Catalog
create_workflow	Creates a new workflow
delete_classifier	Removes a classifier from the Data Catalog

<code>delete_connection</code>	Deletes a connection from the Data Catalog
<code>delete_crawler</code>	Removes a specified crawler from the AWS Glue Data Catalog, unless the crawler sta
<code>delete_database</code>	Removes a specified Database from a Data Catalog
<code>delete_dev_endpoint</code>	Deletes a specified DevEndpoint
<code>delete_job</code>	Deletes a specified job definition
<code>delete_partition</code>	Deletes a specified partition
<code>delete_resource_policy</code>	Deletes a specified policy
<code>delete_security_configuration</code>	Deletes a specified security configuration
<code>delete_table</code>	Removes a table definition from the Data Catalog
<code>delete_table_version</code>	Deletes a specified version of a table
<code>delete_trigger</code>	Deletes a specified trigger
<code>delete_user_defined_function</code>	Deletes an existing function definition from the Data Catalog
<code>delete_workflow</code>	Deletes a workflow
<code>get_catalog_import_status</code>	Retrieves the status of a migration operation
<code>get_classifier</code>	Retrieve a classifier by name
<code>get_classifiers</code>	Lists all classifier objects in the Data Catalog
<code>get_connection</code>	Retrieves a connection definition from the Data Catalog
<code>get_connections</code>	Retrieves a list of connection definitions from the Data Catalog
<code>get_crawler</code>	Retrieves metadata for a specified crawler
<code>get_crawler_metrics</code>	Retrieves metrics about specified crawlers
<code>get_crawlers</code>	Retrieves metadata for all crawlers defined in the customer account
<code>get_data_catalog_encryption_settings</code>	Retrieves the security configuration for a specified catalog
<code>get_database</code>	Retrieves the definition of a specified database
<code>get_databases</code>	Retrieves all Databases defined in a given Data Catalog
<code>get_dataflow_graph</code>	Transforms a Python script into a directed acyclic graph (DAG)
<code>get_dev_endpoint</code>	Retrieves information about a specified DevEndpoint
<code>get_dev_endpoints</code>	Retrieves all the DevEndpoints in this AWS account
<code>get_job</code>	Retrieves an existing job definition
<code>get_job_run</code>	Retrieves the metadata for a given job run
<code>get_job_runs</code>	Retrieves metadata for all runs of a given job definition
<code>get_jobs</code>	Retrieves all current job definitions
<code>get_mapping</code>	Creates mappings
<code>get_partition</code>	Retrieves information about a specified partition
<code>get_partitions</code>	Retrieves information about the partitions in a table
<code>get_plan</code>	Gets code to perform a specified mapping
<code>get_resource_policy</code>	Retrieves a specified resource policy
<code>get_security_configuration</code>	Retrieves a specified security configuration
<code>get_security_configurations</code>	Retrieves a list of all security configurations
<code>get_table</code>	Retrieves the Table definition in a Data Catalog for a specified table
<code>get_table_version</code>	Retrieves a specified version of a table
<code>get_table_versions</code>	Retrieves a list of strings that identify available versions of a specified table
<code>get_tables</code>	Retrieves the definitions of some or all of the tables in a given Database
<code>get_tags</code>	Retrieves a list of tags associated with a resource
<code>get_trigger</code>	Retrieves the definition of a trigger
<code>get_triggers</code>	Gets all the triggers associated with a job
<code>get_user_defined_function</code>	Retrieves a specified function definition from the Data Catalog
<code>get_user_defined_functions</code>	Retrieves a multiple function definitions from the Data Catalog
<code>get_workflow</code>	Retrieves resource metadata for a workflow

<code>get_workflow_run</code>	Retrieves the metadata for a given workflow run
<code>get_workflow_run_properties</code>	Retrieves the workflow run properties which were set during the run
<code>get_workflow_runs</code>	Retrieves metadata for all runs of a given workflow
<code>import_catalog_to_glue</code>	Imports an existing Athena Data Catalog to AWS Glue
<code>list_crawlers</code>	Retrieves the names of all crawler resources in this AWS account, or the resources with the specified prefix
<code>list_dev_endpoints</code>	Retrieves the names of all DevEndpoint resources in this AWS account, or the resources with the specified prefix
<code>list_jobs</code>	Retrieves the names of all job resources in this AWS account, or the resources with the specified prefix
<code>list_triggers</code>	Retrieves the names of all trigger resources in this AWS account, or the resources with the specified prefix
<code>list_workflows</code>	Lists names of workflows created in the account
<code>put_data_catalog_encryption_settings</code>	Sets the security configuration for a specified catalog
<code>put_resource_policy</code>	Sets the Data Catalog resource policy for access control
<code>put_workflow_run_properties</code>	Puts the specified workflow run properties for the given workflow run
<code>reset_job_bookmark</code>	Resets a bookmark entry
<code>start_crawler</code>	Starts a crawl using the specified crawler, regardless of what is scheduled
<code>start_crawler_schedule</code>	Changes the schedule state of the specified crawler to SCHEDULED, unless the crawler is already running
<code>start_job_run</code>	Starts a job run using a job definition
<code>start_trigger</code>	Starts an existing trigger
<code>start_workflow_run</code>	Starts a new run of the specified workflow
<code>stop_crawler</code>	If the specified crawler is running, stops the crawl
<code>stop_crawler_schedule</code>	Sets the schedule state of the specified crawler to NOT_SCHEDULED, but does not stop the crawler if it is already running
<code>stop_trigger</code>	Stops a specified trigger
<code>tag_resource</code>	Adds tags to a resource
<code>untag_resource</code>	Removes tags from a resource
<code>update_classifier</code>	Modifies an existing classifier (a GrokClassifier, an XMLClassifier, a JsonClassifier, or a RegexClassifier)
<code>update_connection</code>	Updates a connection definition in the Data Catalog
<code>update_crawler</code>	Updates a crawler
<code>update_crawler_schedule</code>	Updates the schedule of a crawler using a cron expression
<code>update_database</code>	Updates an existing database definition in a Data Catalog
<code>update_dev_endpoint</code>	Updates a specified DevEndpoint
<code>update_job</code>	Updates an existing job definition
<code>update_partition</code>	Updates a partition
<code>update_table</code>	Updates a metadata table in the Data Catalog
<code>update_trigger</code>	Updates a trigger definition
<code>update_user_defined_function</code>	Updates an existing function definition in the Data Catalog
<code>update_workflow</code>	Updates an existing workflow

Examples

```

svc <- glue()
svc$batch_create_partition(
  Foo = 123
)

```

kafka

Managed Streaming for Kafka

Description

Managed Streaming for Kafka

Usage

kafka()

Operations

create_cluster	Creates a new MSK cluster
create_configuration	Creates a new MSK configuration
delete_cluster	Deletes the MSK cluster specified by the Amazon Resource Name (ARN) in the request
describe_cluster	Returns a description of the MSK cluster whose Amazon Resource Name (ARN) is specified
describe_cluster_operation	Returns a description of the cluster operation specified by the ARN
describe_configuration	Returns a description of this MSK configuration
describe_configuration_revision	Returns a description of this revision of the configuration
get_bootstrap_brokers	A list of brokers that a client application can use to bootstrap
list_cluster_operations	Returns a list of all the operations that have been performed on the specified MSK cluster
list_clusters	Returns a list of all the MSK clusters in the current Region
list_configuration_revisions	Returns a list of all the MSK configurations in this Region
list_configurations	Returns a list of all the MSK configurations in this Region
list_nodes	Returns a list of the broker nodes in the cluster
list_tags_for_resource	Returns a list of the tags associated with the specified resource
tag_resource	Adds tags to the specified MSK resource
untag_resource	Removes the tags associated with the keys that are provided in the query
update_broker_storage	Updates the EBS storage associated with MSK brokers
update_cluster_configuration	Updates the cluster with the configuration that is specified in the request body

Examples

```
svc <- kafka()
svc$create_cluster(
  Foo = 123
)
```

kinesis

Amazon Kinesis

Description

Amazon Kinesis Data Streams Service API Reference

Amazon Kinesis Data Streams is a managed service that scales elastically for real-time processing of streaming big data.

Usage

```
kinesis()
```

Operations

add_tags_to_stream	Adds or updates tags for the specified Kinesis data stream
create_stream	Creates a Kinesis data stream
decrease_stream_retention_period	Decreases the Kinesis data stream's retention period, which is the length of time data records are available
delete_stream	Deletes a Kinesis data stream and all its shards and data
deregister_stream_consumer	To deregister a consumer, provide its ARN
describe_limits	Describes the shard limits and usage for the account
describe_stream	Describes the specified Kinesis data stream
describe_stream_consumer	To get the description of a registered consumer, provide the ARN of the consumer
describe_stream_summary	Provides a summarized description of the specified Kinesis data stream without the shard-level details
disable_enhanced_monitoring	Disables enhanced monitoring
enable_enhanced_monitoring	Enables enhanced Kinesis data stream monitoring for shard-level metrics
get_records	Gets data records from a Kinesis data stream's shard
get_shard_iterator	Gets an Amazon Kinesis shard iterator
increase_stream_retention_period	Increases the Kinesis data stream's retention period, which is the length of time data records are available
list_shards	Lists the shards in a stream and provides information about each shard
list_stream_consumers	Lists the consumers registered to receive data from a stream using enhanced fan-out, and provides information about each consumer
list_streams	Lists your Kinesis data streams
list_tags_for_stream	Lists the tags for the specified Kinesis data stream
merge_shards	Merges two adjacent shards in a Kinesis data stream and combines them into a single shard
put_record	Writes a single data record into an Amazon Kinesis data stream
put_records	Writes multiple data records into a Kinesis data stream in a single call (also referred to as batching)
register_stream_consumer	Registers a consumer with a Kinesis data stream
remove_tags_from_stream	Removes tags from the specified Kinesis data stream
split_shard	Splits a shard into two new shards in the Kinesis data stream, to increase the stream's capacity
start_stream_encryption	Enables or updates server-side encryption using an AWS KMS key for a specified stream
stop_stream_encryption	Disables server-side encryption for a specified stream
update_shard_count	Updates the shard count of the specified stream to the specified number of shards

Examples

```
svc <- kinesis()
svc$add_tags_to_stream(
  Foo = 123
)
```

Description

Overview

This documentation is for version 1 of the Amazon Kinesis Data Analytics API, which only supports SQL applications. Version 2 of the API supports SQL and Java applications. For more information about version 2, see [Amazon Kinesis Data Analytics API V2 Documentation](#).

This is the *Amazon Kinesis Analytics v1 API Reference*. The *Amazon Kinesis Analytics Developer Guide* provides additional information.

Usage

```
kinesisanalytics()
```

Operations

add_application_cloud_watch_logging_option	This documentation is for version 1 of the Amazon Kinesis Data Analyt
add_application_input	This documentation is for version 1 of the Amazon Kinesis Data Analyt
add_application_input_processing_configuration	This documentation is for version 1 of the Amazon Kinesis Data Analyt
add_application_output	This documentation is for version 1 of the Amazon Kinesis Data Analyt
add_application_reference_data_source	This documentation is for version 1 of the Amazon Kinesis Data Analyt
create_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt
delete_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt
delete_application_cloud_watch_logging_option	This documentation is for version 1 of the Amazon Kinesis Data Analyt
delete_application_input_processing_configuration	This documentation is for version 1 of the Amazon Kinesis Data Analyt
delete_application_output	This documentation is for version 1 of the Amazon Kinesis Data Analyt
delete_application_reference_data_source	This documentation is for version 1 of the Amazon Kinesis Data Analyt
describe_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt
discover_input_schema	This documentation is for version 1 of the Amazon Kinesis Data Analyt
list_applications	This documentation is for version 1 of the Amazon Kinesis Data Analyt
list_tags_for_resource	Retrieves the list of key-value tags assigned to the application
start_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt
stop_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt
tag_resource	Adds one or more key-value tags to a Kinesis Analytics application
untag_resource	Removes one or more tags from a Kinesis Analytics application
update_application	This documentation is for version 1 of the Amazon Kinesis Data Analyt

Examples

```
svc <- kinesisanalytics()
svc$add_application_cloud_watch_logging_option(
  Foo = 123
```

)

kinesisanalyticsv2 *Amazon Kinesis Analytics*

Description

Amazon Kinesis Data Analytics is a fully managed service that you can use to process and analyze streaming data using SQL or Java. The service enables you to quickly author and run SQL or Java code against streaming sources to perform time series analytics, feed real-time dashboards, and create real-time metrics.

Usage

```
kinesisanalyticsv2()
```

Operations

add_application_cloud_watch_logging_option	Adds an Amazon CloudWatch log stream to monitor application configuration
add_application_input	Adds a streaming source to your SQL-based Amazon Kinesis Data Analytics application
add_application_input_processing_configuration	Adds an InputProcessingConfiguration to an SQL-based Kinesis Data Analytics application
add_application_output	Adds an external destination to your SQL-based Amazon Kinesis Data Analytics application
add_application_reference_data_source	Adds a reference data source to an existing SQL-based Amazon Kinesis Data Analytics application
create_application	Creates an Amazon Kinesis Data Analytics application
create_application_snapshot	Creates a snapshot of the application's state data
delete_application	Deletes the specified application
delete_application_cloud_watch_logging_option	Deletes an Amazon CloudWatch log stream from an Amazon Kinesis Data Analytics application
delete_application_input_processing_configuration	Deletes an InputProcessingConfiguration from an input
delete_application_output	Deletes the output destination configuration from your SQL-based Amazon Kinesis Data Analytics application
delete_application_reference_data_source	Deletes a reference data source configuration from the specified SQL-based Amazon Kinesis Data Analytics application
delete_application_snapshot	Deletes a snapshot of application state
describe_application	Returns information about a specific Amazon Kinesis Data Analytics application
describe_application_snapshot	Returns information about a snapshot of application state data
discover_input_schema	Infers a schema for an SQL-based Amazon Kinesis Data Analytics application
list_application_snapshots	Lists information about the current application snapshots
list_applications	Returns a list of Amazon Kinesis Data Analytics applications in your account
list_tags_for_resource	Retrieves the list of key-value tags assigned to the application
start_application	Starts the specified Amazon Kinesis Data Analytics application
stop_application	Stops the application from processing data
tag_resource	Adds one or more key-value tags to a Kinesis Analytics application
untag_resource	Removes one or more tags from a Kinesis Analytics application
update_application	Updates an existing Amazon Kinesis Data Analytics application

Examples

```

svc <- kinesisanalyticsv2()
svc$add_application_cloud_watch_logging_option(
  Foo = 123
)

```

mturk

*Amazon Mechanical Turk***Description**

Amazon Mechanical Turk API Reference

Usage

```
mturk()
```

Operations

[accept_qualification_request](#)
[approve_assignment](#)
[associate_qualification_with_worker](#)
[create_additional_assignments_for_hit](#)
[create_hit](#)
[create_hit_type](#)
[create_hit_with_hit_type](#)
[create_qualification_type](#)
[create_worker_block](#)
[delete_hit](#)
[delete_qualification_type](#)
[delete_worker_block](#)
[disassociate_qualification_from_worker](#)
[get_account_balance](#)
[get_assignment](#)
[get_file_upload_url](#)
[get_hit](#)
[get_qualification_score](#)
[get_qualification_type](#)
[list_assignments_for_hit](#)
[list_bonus_payments](#)
[list_hi_ts](#)
[list_hi_ts_for_qualification_type](#)
[list_qualification_requests](#)
[list_qualification_types](#)
[list_review_policy_results_for_hit](#)
[list_reviewable_hi_ts](#)

The AcceptQualificationRequest operation approves a Worker's request for a Quali
 The ApproveAssignment operation approves the results of a completed assignment
 The AssociateQualificationWithWorker operation gives a Worker a Qualification
 The CreateAdditionalAssignmentsForHIT operation increases the maximum number
 The CreateHIT operation creates a new Human Intelligence Task (HIT)
 The CreateHITType operation creates a new HIT type
 The CreateHITWithHITType operation creates a new Human Intelligence Task (HI
 The CreateQualificationType operation creates a new Qualification type, which is re
 The CreateWorkerBlock operation allows you to prevent a Worker from working on
 The DeleteHIT operation is used to delete HIT that is no longer needed
 The DeleteQualificationType deletes a Qualification type and deletes any HIT types
 The DeleteWorkerBlock operation allows you to reinstate a blocked Worker to work
 The DisassociateQualificationFromWorker revokes a previously granted Qualificati
 The GetAccountBalance operation retrieves the amount of money in your Amazon
 The GetAssignment operation retrieves the details of the specified Assignment
 The GetFileUploadURL operation generates and returns a temporary URL
 The GetHIT operation retrieves the details of the specified HIT
 The GetQualificationScore operation returns the value of a Worker's Qualification f
 The GetQualificationTypeoperation retrieves information about a Qualification type
 The ListAssignmentsForHIT operation retrieves completed assignments for a HIT
 The ListBonusPayments operation retrieves the amounts of bonuses you have paid
 The ListHITs operation returns all of a Requester's HITs
 The ListHITsForQualificationType operation returns the HITs that use the given Qu
 The ListQualificationRequests operation retrieves requests for Qualifications of a p
 The ListQualificationTypes operation returns a list of Qualification types, filtered by
 The ListReviewPolicyResultsForHIT operation retrieves the computed results and t
 The ListReviewableHITs operation retrieves the HITs with Status equal to Reviewa

[list_worker_blocks](#)
[list_workers_with_qualification_type](#)
[notify_workers](#)
[reject_assignment](#)
[reject_qualification_request](#)
[send_bonus](#)
[send_test_event_notification](#)
[update_expiration_for_hit](#)
[update_hit_review_status](#)
[update_hit_type_of_hit](#)
[update_notification_settings](#)
[update_qualification_type](#)

The ListWorkersBlocks operation retrieves a list of Workers who are blocked from
 The ListWorkersWithQualificationType operation returns all of the Workers that ha
 The NotifyWorkers operation sends an email to one or more Workers that you speci
 The RejectAssignment operation rejects the results of a completed assignment
 The RejectQualificationRequest operation rejects a user's request for a Qualificatio
 The SendBonus operation issues a payment of money from your account to a Work
 The SendTestEventNotification operation causes Amazon Mechanical Turk to send
 The UpdateExpirationForHIT operation allows you update the expiration time of a
 The UpdateHITReviewStatus operation updates the status of a HIT
 The UpdateHITTypeOfHIT operation allows you to change the HITType properties
 The UpdateNotificationSettings operation creates, updates, disables or re-enables n
 The UpdateQualificationType operation modifies the attributes of an existing Quali

Examples

```

svc <- mturk()
svc$accept_qualification_request(
  Foo = 123
)
  
```

quicksight

Amazon QuickSight

Description

Amazon QuickSight API Reference

Amazon QuickSight is a fully managed, serverless, cloud business intelligence service that makes it easy to extend data and insights to every user in your organization. This API interface reference contains documentation for a programming interface that you can use to manage Amazon QuickSight.

Usage

```
quicksight()
```

Operations

create_group	Creates an Amazon QuickSight group
create_group_membership	Adds an Amazon QuickSight user to an Amazon QuickSight group
delete_group	Removes a user group from Amazon QuickSight
delete_group_membership	Removes a user from a group so that the user is no longer a member of the group
delete_user	Deletes the Amazon QuickSight user that is associated with the identity of the AWS Identity ar
delete_user_by_principal_id	Deletes a user identified by its principal ID
describe_group	Returns an Amazon QuickSight group's description and Amazon Resource Name (ARN)

describe_user	Returns information about a user, given the user name
get_dashboard_embed_url	Generates a server-side embeddable URL and authorization code
list_group_memberships	Lists member users in a group
list_groups	Lists all user groups in Amazon QuickSight
list_user_groups	Lists the Amazon QuickSight groups that an Amazon QuickSight user is a member of
list_users	Returns a list of all of the Amazon QuickSight users belonging to this account
register_user	Creates an Amazon QuickSight user, whose identity is associated with the AWS Identity and A
update_group	Changes a group description
update_user	Updates an Amazon QuickSight user

Examples

```
svc <- quicksight()  
svc$create_group(  
  Foo = 123  
)
```

Index

accept_qualification_request, [17](#)
activate_pipeline, [6](#)
add_application_cloud_watch_logging_option,
 [15, 16](#)
add_application_input, [15, 16](#)
add_application_input_processing_configuration,
 [15, 16](#)
add_application_output, [15, 16](#)
add_application_reference_data_source,
 [15, 16](#)
add_instance_fleet, [8](#)
add_instance_groups, [8](#)
add_job_flow_steps, [8](#)
add_tags, [6–8](#)
add_tags_to_stream, [14](#)
approve_assignment, [17](#)
associate_qualification_with_worker,
 [17](#)
athena, [2](#)

batch_create_partition, [10](#)
batch_delete_connection, [10](#)
batch_delete_partition, [10](#)
batch_delete_table, [10](#)
batch_delete_table_version, [10](#)
batch_get_crawlers, [10](#)
batch_get_dev_endpoints, [10](#)
batch_get_jobs, [10](#)
batch_get_named_query, [3](#)
batch_get_partition, [10](#)
batch_get_query_execution, [3](#)
batch_get_triggers, [10](#)
batch_get_workflows, [10](#)
batch_stop_job_run, [10](#)
build_suggesters, [4](#)

cancel_elasticsearch_service_software_update,
 [7](#)
cancel_steps, [8](#)
cloudsearch, [3](#)

cloudsearchdomain, [5](#)
create_additional_assignments_for_hit,
 [17](#)
create_application, [15, 16](#)
create_application_snapshot, [16](#)
create_classifier, [10](#)
create_cluster, [13](#)
create_configuration, [13](#)
create_connection, [10](#)
create_crawler, [10](#)
create_database, [10](#)
create_delivery_stream, [9](#)
create_dev_endpoint, [10](#)
create_domain, [4](#)
create_elasticsearch_domain, [7](#)
create_group, [18](#)
create_group_membership, [18](#)
create_hit, [17](#)
create_hit_type, [17](#)
create_hit_with_hit_type, [17](#)
create_job, [10](#)
create_named_query, [3](#)
create_partition, [10](#)
create_pipeline, [6](#)
create_qualification_type, [17](#)
create_script, [10](#)
create_security_configuration, [8, 10](#)
create_stream, [14](#)
create_table, [10](#)
create_trigger, [10](#)
create_user_defined_function, [10](#)
create_work_group, [3](#)
create_worker_block, [17](#)
create_workflow, [10](#)

datapipeline, [5](#)
deactivate_pipeline, [6](#)
decrease_stream_retention_period, [14](#)
define_analysis_scheme, [4](#)
define_expression, [4](#)

- define_index_field, [4](#)
- define_suggester, [4](#)
- delete_analysis_scheme, [4](#)
- delete_application, [15](#), [16](#)
- delete_application_cloud_watch_logging_options, [15](#), [16](#)
- delete_application_input_processing_configuration, [15](#), [16](#)
- delete_application_output, [15](#), [16](#)
- delete_application_reference_data_source, [15](#), [16](#)
- delete_application_snapshot, [16](#)
- delete_classifier, [10](#)
- delete_cluster, [13](#)
- delete_connection, [11](#)
- delete_crawler, [11](#)
- delete_database, [11](#)
- delete_delivery_stream, [9](#)
- delete_dev_endpoint, [11](#)
- delete_domain, [4](#)
- delete_elasticsearch_domain, [7](#)
- delete_elasticsearch_service_role, [7](#)
- delete_expression, [4](#)
- delete_group, [18](#)
- delete_group_membership, [18](#)
- delete_hit, [17](#)
- delete_index_field, [4](#)
- delete_job, [11](#)
- delete_named_query, [3](#)
- delete_partition, [11](#)
- delete_pipeline, [6](#)
- delete_qualification_type, [17](#)
- delete_resource_policy, [11](#)
- delete_security_configuration, [8](#), [11](#)
- delete_stream, [14](#)
- delete_suggester, [4](#)
- delete_table, [11](#)
- delete_table_version, [11](#)
- delete_trigger, [11](#)
- delete_user, [18](#)
- delete_user_by_principal_id, [18](#)
- delete_user_defined_function, [11](#)
- delete_work_group, [3](#)
- delete_worker_block, [17](#)
- delete_workflow, [11](#)
- deregister_stream_consumer, [14](#)
- describe_analysis_schemes, [4](#)
- describe_application, [15](#), [16](#)
- describe_application_snapshot, [16](#)
- describe_availability_options, [4](#)
- describe_cluster, [8](#), [13](#)
- describe_cluster_operation, [13](#)
- describe_configuration, [13](#)
- describe_configuration_revision, [13](#)
- describe_delivery_stream, [9](#)
- describe_domains, [4](#)
- describe_elasticsearch_domain, [7](#)
- describe_elasticsearch_domain_config, [7](#)
- describe_elasticsearch_domains, [7](#)
- describe_elasticsearch_instance_type_limits, [7](#)
- describe_expressions, [4](#)
- describe_group, [18](#)
- describe_index_fields, [4](#)
- describe_job_flows, [8](#)
- describe_limits, [14](#)
- describe_objects, [6](#)
- describe_pipelines, [6](#)
- describe_reserved_elasticsearch_instance_offerings, [7](#)
- describe_reserved_elasticsearch_instances, [7](#)
- describe_scaling_parameters, [4](#)
- describe_security_configuration, [8](#)
- describe_service_access_policies, [4](#)
- describe_step, [8](#)
- describe_stream, [14](#)
- describe_stream_consumer, [14](#)
- describe_stream_summary, [14](#)
- describe_suggesters, [4](#)
- describe_user, [19](#)
- disable_enhanced_monitoring, [14](#)
- disassociate_qualification_from_worker, [17](#)
- discover_input_schema, [15](#), [16](#)
- elasticsearchservice, [7](#)
- emr, [8](#)
- enable_enhanced_monitoring, [14](#)
- evaluate_expression, [6](#)
- firehose, [9](#)
- get_account_balance, [17](#)
- get_assignment, [17](#)
- get_bootstrap_brokers, [13](#)

get_catalog_import_status, [11](#)
get_classifier, [11](#)
get_classifiers, [11](#)
get_compatible_elasticsearch_versions,
[7](#)
get_connection, [11](#)
get_connections, [11](#)
get_crawler, [11](#)
get_crawler_metrics, [11](#)
get_crawlers, [11](#)
get_dashboard_embed_url, [19](#)
get_data_catalog_encryption_settings,
[11](#)
get_database, [11](#)
get_databases, [11](#)
get_dataflow_graph, [11](#)
get_dev_endpoint, [11](#)
get_dev_endpoints, [11](#)
get_file_upload_url, [17](#)
get_hit, [17](#)
get_job, [11](#)
get_job_run, [11](#)
get_job_runs, [11](#)
get_jobs, [11](#)
get_mapping, [11](#)
get_named_query, [3](#)
get_partition, [11](#)
get_partitions, [11](#)
get_pipeline_definition, [6](#)
get_plan, [11](#)
get_qualification_score, [17](#)
get_qualification_type, [17](#)
get_query_execution, [3](#)
get_query_results, [3](#)
get_records, [14](#)
get_resource_policy, [11](#)
get_security_configuration, [11](#)
get_security_configurations, [11](#)
get_shard_iterator, [14](#)
get_table, [11](#)
get_table_version, [11](#)
get_table_versions, [11](#)
get_tables, [11](#)
get_tags, [11](#)
get_trigger, [11](#)
get_triggers, [11](#)
get_upgrade_history, [7](#)
get_upgrade_status, [7](#)
get_user_defined_function, [11](#)
get_user_defined_functions, [11](#)
get_work_group, [3](#)
get_workflow, [11](#)
get_workflow_run, [12](#)
get_workflow_run_properties, [12](#)
get_workflow_runs, [12](#)
glue, [10](#)

import_catalog_to_glue, [12](#)
increase_stream_retention_period, [14](#)
index_documents, [4](#)

kafka, [13](#)
kinesis, [13](#)
kinesisanalytics, [15](#)
kinesisanalyticsv2, [16](#)

list_application_snapshots, [16](#)
list_applications, [15, 16](#)
list_assignments_for_hit, [17](#)
list_bonus_payments, [17](#)
list_bootstrap_actions, [8](#)
list_cluster_operations, [13](#)
list_clusters, [8, 13](#)
list_configuration_revisions, [13](#)
list_configurations, [13](#)
list_crawlers, [12](#)
list_delivery_streams, [9](#)
list_dev_endpoints, [12](#)
list_domain_names, [4, 7](#)
list_elasticsearch_instance_types, [7](#)
list_elasticsearch_versions, [7](#)
list_group_memberships, [19](#)
list_groups, [19](#)
list_hi_ts, [17](#)
list_hi_ts_for_qualification_type, [17](#)
list_instance_fleets, [8](#)
list_instance_groups, [8](#)
list_instances, [8](#)
list_jobs, [12](#)
list_named_queries, [3](#)
list_nodes, [13](#)
list_pipelines, [6](#)
list_qualification_requests, [17](#)
list_qualification_types, [17](#)
list_query_executions, [3](#)
list_review_policy_results_for_hit, [17](#)
list_reviewable_hi_ts, [17](#)

[list_security_configurations](#), 8
[list_shards](#), 14
[list_steps](#), 8
[list_stream_consumers](#), 14
[list_streams](#), 14
[list_tags](#), 7
[list_tags_for_delivery_stream](#), 9
[list_tags_for_resource](#), 3, 13, 15, 16
[list_tags_for_stream](#), 14
[list_triggers](#), 12
[list_user_groups](#), 19
[list_users](#), 19
[list_work_groups](#), 3
[list_worker_blocks](#), 18
[list_workers_with_qualification_type](#), 18
[list_workflows](#), 12

[merge_shards](#), 14
[modify_instance_fleet](#), 8
[modify_instance_groups](#), 8
[mturk](#), 17

[notify_workers](#), 18

[poll_for_task](#), 6
[purchase_reserved_elasticsearch_instance_offering](#), 7
[put_auto_scaling_policy](#), 8
[put_data_catalog_encryption_settings](#), 12
[put_pipeline_definition](#), 6
[put_record](#), 9, 14
[put_record_batch](#), 9
[put_records](#), 14
[put_resource_policy](#), 12
[put_workflow_run_properties](#), 12

[query_objects](#), 6
[quicksight](#), 18

[register_stream_consumer](#), 14
[register_user](#), 19
[reject_assignment](#), 18
[reject_qualification_request](#), 18
[remove_auto_scaling_policy](#), 8
[remove_tags](#), 6–8
[remove_tags_from_stream](#), 14
[report_task_progress](#), 6

[report_task_runner_heartbeat](#), 6
[reset_job_bookmark](#), 12
[run_job_flow](#), 9

[search](#), 5
[send_bonus](#), 18
[send_test_event_notification](#), 18
[set_status](#), 6
[set_task_status](#), 6
[set_termination_protection](#), 9
[set_visible_to_all_users](#), 9
[split_shard](#), 14
[start_application](#), 15, 16
[start_crawler](#), 12
[start_crawler_schedule](#), 12
[start_delivery_stream_encryption](#), 9
[start_elasticsearch_service_software_update](#), 7
[start_job_run](#), 12
[start_query_execution](#), 3
[start_stream_encryption](#), 14
[start_trigger](#), 12
[start_workflow_run](#), 12
[stop_application](#), 15, 16
[stop_crawler](#), 12
[stop_crawler_schedule](#), 12
[stop_delivery_stream_encryption](#), 9
[stop_query_execution](#), 3
[stop_stream_encryption](#), 14
[stop_trigger](#), 12
[suggest](#), 5

[tag_delivery_stream](#), 9
[tag_resource](#), 3, 12, 13, 15, 16
[terminate_job_flows](#), 9

[untag_delivery_stream](#), 9
[untag_resource](#), 3, 12, 13, 15, 16
[update_application](#), 15, 16
[update_availability_options](#), 4
[update_broker_storage](#), 13
[update_classifier](#), 12
[update_cluster_configuration](#), 13
[update_connection](#), 12
[update_crawler](#), 12
[update_crawler_schedule](#), 12
[update_database](#), 12
[update_destination](#), 9
[update_dev_endpoint](#), 12

update_elasticsearch_domain_config, [7](#)
update_expiration_for_hit, [18](#)
update_group, [19](#)
update_hit_review_status, [18](#)
update_hit_type_of_hit, [18](#)
update_job, [12](#)
update_notification_settings, [18](#)
update_partition, [12](#)
update_qualification_type, [18](#)
update_scaling_parameters, [4](#)
update_service_access_policies, [4](#)
update_shard_count, [14](#)
update_table, [12](#)
update_trigger, [12](#)
update_user, [19](#)
update_user_defined_function, [12](#)
update_work_group, [3](#)
update_workflow, [12](#)
upgrade_elasticsearch_domain, [7](#)
upload_documents, [5](#)

validate_pipeline_definition, [6](#)