I'm not a robot

Aws athena api example github

You can’t perform that action at this time. You can’t perform that action at this time. Package athena provides the API client, operations, and parameter types for Amazon Athena. 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. Constants func NewDefaultEndpointResolver() *internalendpoints.Resolver func WithAPIOptions(optFns ...func(*middleware.Stack) error) func(*Options) func WithEndpointResolver(v EndpointResolver) func(*Options)deprecated func WithEndpointResolverV2(v EndpointResolverV2) func(*Options) func WithSigV4SigningName(name string) func(*Options) func WithSigV4SigningRegion(region string) func(*Options) type AuthResolverParameters type AuthSchemeResolver type BatchGetNamedQueryInput type BatchGetNamedQueryOutput type BatchGetPreparedStatementInput type BatchGetPreparedStatementOutput type BatchGetQueryExecutionInput type BatchGetQueryExecutionOutput type CancelCapacityReservationInput type CancelCapacityReservationOutput type Client func (c *Client) BatchGetNamedQuery(ctx context.Context, params *BatchGetNamedQueryInput, optFns ...func(*Options)) (*BatchGetNamedQueryOutput, error) func (c *Client) BatchGetPreparedStatement(ctx context.Context, params *BatchGetPreparedStatementInput, ...) (*BatchGetPreparedStatementOutput, error) func (c *Client) BatchGetQueryExecution(ctx context.Context, params *BatchGetQueryExecutionInput, ...) (*BatchGetQueryExecutionOutput, error) func (c *Client) CancelCapacityReservation(ctx context.Context, params *CancelCapacityReservationInput, ...) (*CancelCapacityReservationOutput, error) func (c *Client) CreateCapacityReservation(ctx context.Context, params *CreateCapacityReservationInput, ...) (*CreateCapacityReservationOutput, error) func (c *Client) CreateDataCatalog(ctx context.Context, params *CreateDataCatalogInput, optFns ...func(*Options)) (*CreateDataCatalogOutput, error) func (c *Client) CreateNamedQuery(ctx context.Context, params *CreateNamedQueryInput, optFns ...func(*Options)) (*CreateNamedQueryOutput, error) func (c *Client) CreateNotebook(ctx context.Context, params *CreateNotebookInput, optFns ...func(*Options)) (*CreateNotebookOutput, error) func (c *Client) CreatePreparedStatement(ctx context.Context, params *CreatePreparedStatementInput, ...) (*CreatePreparedStatementOutput, error) func (c *Client) CreatePresignedNotebookUrl(ctx context.Context, params *CreatePresignedNotebookUrlInput, ...) (*CreatePresignedNotebookUrlOutput, error) func (c *Client) CreateWorkGroup(ctx context.Context, params *CreateWorkGroupInput, optFns ...func(*Options)) (*CreateWorkGroupOutput, error) func (c *Client) DeleteCapacityReservation(ctx context.Context, params *DeleteCapacityReservationInput, ...) (*DeleteCapacityReservationOutput, error) func (c *Client) DeleteDataCatalog(ctx context.Context, params *DeleteDataCatalogInput, optFns ...func(*Options)) (*DeleteDataCatalogOutput, error) func (c *Client) DeleteNamedQuery(ctx context.Context, params *DeleteNamedQueryInput, optFns ...func(*Options)) (*DeleteNamedQueryOutput, error) func (c *Client) DeleteNotebook(ctx context.Context, params *DeleteNotebookInput, optFns ...func(*Options)) (*DeleteNotebookOutput, error) func (c *Client) DeletePreparedStatement(ctx context.Context, params *DeletePreparedStatementInput, ...) (*DeletePreparedStatementOutput, error) func (c *Client) DeleteWorkGroup(ctx context.Context, params *DeleteWorkGroupInput, optFns ...func(*Options)) (*DeleteWorkGroupOutput, error) func (c *Client) ExportNotebook(ctx context.Context, params *ExportNotebookInput, optFns ...func(*Options)) (*ExportNotebookOutput, error) func (c *Client) GetCalculationExecution(ctx context.Context, params *GetCalculationExecutionInput, ...) (*GetCalculationExecutionOutput, error) func (c *Client) GetCalculationExecutionCode(ctx context.Context, params *GetCalculationExecutionCodeInput, ...) (*GetCalculationExecutionCodeOutput, error) func (c *Client) GetCalculationExecutionStatus(ctx context.Context, params *GetCalculationExecutionStatusInput, ...) (*GetCalculationExecutionStatusOutput, error) func (c *Client) GetCapacityAssignmentConfiguration(ctx context.Context, params *GetCapacityAssignmentConfigurationInput, ...) (*GetCapacityAssignmentConfigurationOutput, error) func (c *Client) GetCapacityReservation(ctx context.Context, params *GetCapacityReservationInput, ...) (*GetCapacityReservationOutput, error) func (c *Client) GetDataCatalog(ctx context.Context, params *GetDataCatalogInput, optFns ...func(*Options)) (*GetDataCatalogOutput, error) func (c *Client) GetDatabase(ctx context.Context, params *GetDatabaseInput, optFns ...func(*Options)) (*GetDatabaseOutput, error) func (c *Client) GetNamedQuery(ctx context.Context, params *GetNamedQueryInput, optFns ...func(*Options)) (*GetNamedQueryOutput, error) func (c *Client) GetNotebookMetadata(ctx context.Context, params *GetNotebookMetadataInput, ...) (*GetNotebookMetadataOutput, error) func (c *Client) GetPreparedStatement(ctx context.Context, params *GetPreparedStatementInput, ...) (*GetPreparedStatementOutput, error) func (c *Client) GetQueryExecution(ctx context.Context, params *GetQueryExecutionInput, optFns ...func(*Options)) (*GetQueryExecutionOutput, error) func (c *Client) GetQueryResults(ctx context.Context, params *GetQueryResultsInput, optFns ...func(*Options)) (*GetQueryResultsOutput, error) func (c *Client) GetQueryRuntimeStatistics(ctx context.Context, params *GetQueryRuntimeStatisticsInput, ...) (*GetQueryRuntimeStatisticsOutput, error) func (c *Client) GetSession(ctx context.Context, params *GetSessionInput, optFns ...func(*Options)) (*GetSessionOutput, error) func (c *Client) GetSessionStatus(ctx context.Context, params *GetSessionStatusInput, optFns ...func(*Options)) (*GetSessionStatusOutput, error) func (c *Client) GetTableMetadata(ctx context.Context, params *GetTableMetadataInput, optFns ...func(*Options)) (*GetTableMetadataOutput, error) func (c *Client) GetWorkGroup(ctx context.Context, params *GetWorkGroupInput, optFns ...func(*Options)) (*GetWorkGroupOutput, error) func (c *Client) ImportNotebook(ctx context.Context, params *ImportNotebookInput, optFns ...func(*Options)) (*ImportNotebookOutput, error) func (c *Client) ListApplicationDPUSizes(ctx context.Context, params *ListApplicationDPUSizesInput, ...) (*ListApplicationDPUSizesOutput, error) func (c *Client) ListCalculationExecutions(ctx context.Context, params *ListCalculationExecutionsInput, ...) (*ListCalculationExecutionsOutput, error) func (c *Client) ListCapacityReservations(ctx context.Context, params *ListCapacityReservationsInput, ...) (*ListCapacityReservationsOutput, error) func (c *Client) ListDataCatalogs(ctx context.Context, params *ListDataCatalogsInput, optFns ...func(*Options)) (*ListDataCatalogsOutput, error) func (c *Client) ListDatabases(ctx context.Context, params *ListDatabasesInput, optFns ...func(*Options)) (*ListDatabasesOutput, error) func (c *Client) ListEngineVersions(ctx context.Context, params *ListEngineVersionsInput, optFns ...func(*Options)) (*ListEngineVersionsOutput, error) func (c *Client) ListExecutors(ctx context.Context, params *ListExecutorsInput, optFns ...func(*Options)) (*ListExecutorsOutput, error) func (c *Client) ListNamedQueries(ctx context.Context, params *ListNamedQueriesInput, optFns ...func(*Options)) (*ListNamedQueriesOutput, error) func (c *Client) ListNotebookMetadata(ctx context.Context, params *ListNotebookMetadataInput, ...) (*ListNotebookMetadataOutput, error) func (c *Client) ListNotebookSessions(ctx context.Context, params *ListNotebookSessionsInput, ...) (*ListNotebookSessionsOutput, error) func (c *Client) ListPreparedStatements(ctx context.Context, params *ListPreparedStatementsInput, ...) (*ListPreparedStatementsOutput, error) func (c *Client) ListQueryExecutions(ctx context.Context, params *ListQueryExecutionsInput, ...) (*ListQueryExecutionsOutput, error) func (c *Client) ListSessions(ctx context.Context, params *ListSessionsInput, optFns ...func(*Options)) (*ListSessionsOutput, error) func (c *Client) ListTableMetadata(ctx context.Context, params *ListTableMetadataInput, optFns ...func(*Options)) (*ListTableMetadataOutput, error) func (c *Client) ListTagsForResource(ctx context.Context, params *ListTagsForResourceInput, ...) (*ListTagsForResourceOutput, error) func (c *Client) ListWorkGroups(ctx context.Context, params *ListWorkGroupsInput, optFns ...func(*Options)) (*ListWorkGroupsOutput, error) func (c *Client) Options() Options func (c *Client) PutCapacityAssignmentConfiguration(ctx context.Context, params *PutCapacityAssignmentConfigurationInput, ...) (*PutCapacityAssignmentConfigurationOutput, error) func (c *Client) StartCalculationExecution(ctx context.Context, params *StartCalculationExecutionInput, ...) (*StartCalculationExecutionOutput, error) func (c *Client) StartQueryExecution(ctx context.Context, params *StartQueryExecutionInput, ...) (*StartQueryExecutionOutput, error) func (c *Client) StartSession(ctx context.Context, params *StartSessionInput, optFns ...func(*Options)) (*StartSessionOutput, error) func (c *Client) StopCalculationExecution(ctx context.Context, params *StopCalculationExecutionInput, ...) (*StopCalculationExecutionOutput, error) func (c *Client) StopQueryExecution(ctx context.Context, params *StopQueryExecutionInput, optFns ...func(*Options)) (*StopQueryExecutionOutput, error) func (c *Client) TagResource(ctx context.Context, params *TagResourceInput, optFns ...func(*Options)) (*TagResourceOutput, error) func (c *Client) TerminateSession(ctx context.Context, params *TerminateSessionInput, optFns ...func(*Options)) (*TerminateSessionOutput, error) func (c *Client) UntagResource(ctx context.Context, params *UntagResourceInput, optFns ...func(*Options)) (*UntagResourceOutput, error) func (c *Client) UpdateCapacityReservation(ctx context.Context, params *UpdateCapacityReservationInput, ...) (*UpdateCapacityReservationOutput, error) func (c *Client) UpdateDataCatalog(ctx context.Context, params *UpdateDataCatalogInput, optFns ...func(*Options)) (*UpdateDataCatalogOutput, error) func (c *Client) UpdateNamedQuery(ctx context.Context, params *UpdateNamedQueryInput, optFns ...func(*Options)) (*UpdateNamedQueryOutput, error) func (c *Client) UpdateNotebook(ctx context.Context, params *UpdateNotebookInput, optFns ...func(*Options)) (*UpdateNotebookOutput, error) func (c *Client) UpdateNotebookMetadata(ctx context.Context, params *UpdateNotebookMetadataInput, ...) (*UpdateNotebookMetadataOutput, error) func (c *Client) UpdatePreparedStatement(ctx context.Context, params *UpdatePreparedStatementInput, ...) (*UpdatePreparedStatementOutput, error) func (c *Client) UpdateWorkGroup(ctx context.Context, params *UpdateWorkGroupInput, optFns ...func(*Options)) (*UpdateWorkGroupOutput, error) type CreateCapacityReservationInput type CreateCapacityReservationOutput type CreateDataCatalogInput type CreateDataCatalogOutput type CreateNamedQueryInput type CreateNamedQueryOutput type CreateNotebookInput type CreateNotebookOutput type CreatePreparedStatementInput type CreatePreparedStatementOutput type CreatePresignedNotebookUrlInput type CreatePresignedNotebookUrlOutput type CreateWorkGroupInput type CreateWorkGroupOutput type DeleteCapacityReservationInput type DeleteCapacityReservationOutput type DeleteDataCatalogInput type DeleteDataCatalogOutput type DeleteNamedQueryInput type DeleteNamedQueryOutput type DeleteNotebookInput type DeleteNotebookOutput type DeletePreparedStatementInput type DeletePreparedStatementOutput type DeleteWorkGroupInput type DeleteWorkGroupOutput type EndpointParameters type EndpointResolver type EndpointResolverFunc type EndpointResolverOptions type EndpointResolverV2 type NewDefaultEndpointResolverV2() EndpointResolverV2 type ExportNotebookInput type ExportNotebookOutput type GetCalculationExecutionCodeInput type GetCalculationExecutionCodeOutput type GetCalculationExecutionInput type GetCalculationExecutionOutput type GetCalculationExecutionStatusInput type GetCalculationExecutionStatusOutput type GetCapacityAssignmentConfigurationInput type GetCapacityAssignmentConfigurationOutput type GetCapacityReservationInput type GetCapacityReservationOutput type GetDataCatalogInput type GetDataCatalogOutput type GetDatabaseInput type GetDatabaseOutput type GetNamedQueryInput type GetNamedQueryOutput type GetNotebookMetadataInput type GetNotebookMetadataOutput type GetPreparedStatementInput type GetPreparedStatementOutput type GetQueryExecutionInput type GetQueryExecutionOutput type GetQueryResultsAPIClient type GetQueryResultsInput type GetQueryResultsOutput type GetQueryResultsPaginator type GetQueryResultsPaginatorOptions type GetQueryRuntimeStatisticsInput type GetQueryRuntimeStatisticsOutput type GetSessionInput type GetSessionOutput type GetSessionStatusInput type GetSessionStatusOutput type GetTableMetadataInput type GetTableMetadataOutput type GetWorkGroupInput type GetWorkGroupOutput type HTTPClient type HTTPSignerV4 type IdempotencyTokenProvider type ImportNotebookInput type ImportNotebookOutput type ListApplicationDPUSizesAPIClient type ListApplicationDPUSizesInput type ListApplicationDPUSizesOutput type ListApplicationDPUSizesPaginator type ListApplicationDPUSizesPaginatorOptions type ListCalculationExecutionsAPIClient type ListCalculationExecutionsInput type ListCalculationExecutionsOutput type ListCalculationExecutionsPaginator type ListCalculationExecutionsPaginatorOptions type ListCapacityReservationsAPIClient type ListCapacityReservationsInput type ListCapacityReservationsOutput type ListCapacityReservationsPaginator type ListCapacityReservationsPaginatorOptions type ListDataCatalogsAPIClient type ListDataCatalogsInput type ListDataCatalogsOutput type ListDataCatalogsPaginator type ListDataCatalogsPaginatorOptions type ListDatabasesAPIClient type ListDatabasesInput type ListDatabasesOutput type ListDatabasesPaginator type ListDatabasesPaginatorOptions type ListEngineVersionsAPIClient type ListEngineVersionsInput type ListEngineVersionsOutput type ListEngineVersionsPaginator type ListEngineVersionsPaginatorOptions type ListExecutorsAPIClient type ListExecutorsInput type ListExecutorsOutput type ListExecutorsPaginator type ListExecutorsPaginatorOptions type ListNamedQueriesAPIClient type ListNamedQueriesInput type ListNamedQueriesOutput type ListNamedQueriesPaginator type ListNamedQueriesPaginatorOptions type ListNotebookMetadataInput type ListNotebookMetadataOutput type ListNotebookSessionsInput type ListNotebookSessionsOutput type ListPreparedStatementsAPIClient type ListPreparedStatementsInput type ListPreparedStatementsOutput type ListPreparedStatementsPaginator type ListPreparedStatementsPaginatorOptions type ListQueryExecutionsAPIClient type ListQueryExecutionsInput type ListQueryExecutionsOutput type ListQueryExecutionsPaginator type ListQueryExecutionsPaginatorOptions type ListSessionsAPIClient type ListSessionsInput type ListSessionsOutput type ListSessionsPaginator type ListSessionsPaginatorOptions type ListTableMetadataAPIClient type ListTableMetadataInput type ListTableMetadataOutput type ListTableMetadataPaginator type ListTableMetadataPaginatorOptions type ListTagsForResourceAPIClient type ListTagsForResourceInput type ListTagsForResourceOutput type ListTagsForResourcePaginator type ListTagsForResourcePaginatorOptions type ListWorkGroupsAPIClient type ListWorkGroupsInput type ListWorkGroupsOutput type ListWorkGroupsPaginator type ListWorkGroupsPaginatorOptions type Options type PutCapacityAssignmentConfigurationInput type PutCapacityAssignmentConfigurationOutput type ResolveEndpoint type StartCalculationExecutionInput type StartCalculationExecutionOutput type StartQueryExecutionInput type StartQueryExecutionOutput type StartSessionInput type StartSessionOutput type StopCalculationExecutionInput type StopCalculationExecutionOutput type StopQueryExecutionInput type StopQueryExecutionOutput type TagResourceInput type TagResourceOutput type TerminateSessionInput type TerminateSessionOutput type UntagResourceInput type UntagResourceOutput type UpdateCapacityReservationInput type UpdateCapacityReservationOutput type UpdateDataCatalogInput type UpdateDataCatalogOutput type UpdateNamedQueryInput type UpdateNamedQueryOutput type UpdateNotebookInput type UpdateNotebookMetadataInput type UpdateNotebookMetadataOutput type UpdateNotebookOutput type UpdatePreparedStatementInput type UpdatePreparedStatementOutput type UpdateWorkGroupInput type UpdateWorkGroupOutput View Source const ServiceAPIVersion = "2017-05-18" View Source const ServiceID = "Athena" This section is empty. func NewDefaultEndpointResolver() *internalendpoints.Resolver NewDefaultEndpointResolver constructs a new service endpoint resolver func WithAPIOptions(optFns ...func(*middleware.Stack) error) func(*Options) WithAPIOptions returns a functional option for setting the Client's APIOptions option. func WithEndpointResolver(v EndpointResolver) func(*Options) Deprecated: EndpointResolver and WithEndpointResolver. Providing a value for this field will likely prevent you from using any endpoint-related service features released after the introduction of EndpointResolverV2 and BaseEndpoint. To migrate an EndpointResolver implementation that uses a custom endpoint, set the client option BaseEndpoint instead. func WithEndpointResolverV2(v EndpointResolverV2) func(*Options) WithEndpointResolverV2 returns a functional option for setting the Client's EndpointResolverV2 option. func WithSigV4SigningName(name string) func(*Options) WithSigV4SigningName applies an override to the authentication workflow to use the given signing name for SigV4-authenticated operations. This is an advanced setting. The value here is FINAL, taking precedence over the resolved signing name from both auth scheme resolution and endpoint resolution. func WithSigV4SigningRegion(region string) func(*Options) WithSigV4SigningRegion applies an override to the authentication workflow to use the given signing region for SigV4-authenticated operations. This is an advanced setting. The value here is FINAL, taking precedence over the resolved signing region from both auth scheme resolution and endpoint resolution. type AuthResolverParameters struct { Operation string Region string } AuthResolverParameters contains the set of inputs necessary for auth scheme resolution. type BatchGetNamedQueryInput struct { NamedQueryIds []string } Contains an array of named query IDs. type BatchGetPreparedStatementInput struct { PreparedStatementNames []string WorkGroup *string } type BatchGetQueryExecutionInput struct { QueryExecutionIds []string } Contains an array of query execution IDs. type CancelCapacityReservationInput struct { Name *string } Client provides the API client to make operations call for Amazon Athena. func New(options Options, optFns ...func(*Options)) *Client New returns an initialized Client based on the functional options. Provide additional functional options to further configure the behavior of the client, such as changing the client's endpoint or adding custom middleware behavior. func NewFromConfig(cfg aws.Config, optFns ...func(*Options)) *Client NewFromConfig returns a new client from the provided config. Returns the details of a single named query or a list of up to 50 queries, which you provide as an array of query ID strings. Requires you to have access to the workgroup in which the queries were saved. Use ListNamedQueriesInputto get the list of named query IDs in the specified workgroup. If information could not be retrieved for a submitted query ID, information about the query ID submitted is listed under UnprocessedNamedQueryId . Named queries differ from executed queries. Use BatchGetQueryExecutionInputto get details about each unique query execution, and ListQueryExecutionsInputto get a list of query execution IDs. Returns the details of a single prepared statement or a list of up to 256 prepared statements for the array of prepared statement names that you provide. Requires you to have access to the workgroup to which the prepared statements belong. If a prepared statement cannot be retrieved for the name specified, the statement is listed in UnprocessedPreparedStatementNames . Returns the details of a single query execution or a list of up to 50 query executions, which you provide as an array of query execution ID strings. Requires you to have access to the workgroup in which the queries ran. To get a list of query execution IDs, use ListQueryExecutionsInput$WorkGroup. Query executions differ from named (saved) queries. Use BatchGetNamedQueryInputto get details about named queries. Cancels the capacity reservation with the specified name. Cancelled reservations remain in your account and will be deleted 45 days after cancellation. During the 45 days, you cannot re-purpose or reuse a reservation that has been cancelled, but you can refer to its tags and view it for historical reference. Creates (registers) a data catalog with the specified name and properties. Catalogs created are visible to all users of the same Amazon Web Services account. For a FEDERATED catalog, this API operation creates the following resources. CFN Stack Name with a maximum length of 128 characters and prefix athenafederatedcatalog-CATALOG_NAME_SANITIZED with length 23 characters. Lambda Function Name with a maximum length of 64 characters and prefix athenafederatedcatalog_CATALOG_NAME_SANITIZED with length 23 characters. Glue Connection Name with a maximum length of 255 characters and a prefix athenafederatedcatalog_CATALOG_NAME_SANITIZED with length 23 characters. Deletes a cancelled capacity reservation. A reservation must be cancelled before it can be deleted. A deleted reservation is immediately removed from your account and can no longer be referenced, including by its ARN. A deleted reservation cannot be called by GetCapacityReservation , and deleted reservations do not appear in the output of ListCapacityReservations . Streams the results of a single query execution specified by QueryExecutionId from the Athena query results location in Amazon S3. For more information, see Working with query results, recent queries, and output files in the Amazon Athena User Guide. This request does not execute the query but returns results. Use StartQueryExecutionto run a query. To stream query results successfully, the IAM principal with permission to call GetQueryResults also must have permissions to the Amazon S3 GetObject action for the Athena query results location. IAM principals with permission to the Amazon S3 GetObject action for the query results location are able to retrieve query results from Amazon S3 even if permission to the GetQueryResults action is denied. To restrict user or role access, ensure that Amazon S3 permissions to the Athena query location are denied. Returns query execution runtime statistics related to a single execution of a query if you have access to the workgroup in which the query ran. Statistics from the Timeline section of the response object are available as soon as QueryExecutionStatus$State is in a SUCCEEDED or FAILED state. The remaining non-timeline statistics in the response (like stage-level input and output row count and data size) are updated asynchronously and may not be available immediately after a query completes. The non-timeline statistics are also not included when a query has row-level filters defined in Lake Formation. Imports a single ipynb file to a Spark enabled workgroup. To import the notebook, the request must specify a value for either Payload or NoteBookS3LocationUri . If neither is specified or both are specified, an InvalidRequestException occurs. The maximum file size that can be imported is 10 megabytes. If an ipynb file with the same name already exists in the workgroup, throws an error. Provides a list of available query IDs only for queries saved in the specified workgroup. Requires that you have access to the specified workgroup. If a workgroup is not specified, lists the saved queries for the primary workgroup. Provides a list of available query execution IDs for the queries in the specified workgroup. Athena keeps a query history for 45 days. If a workgroup is not specified, returns a list of query execution IDs for the primary workgroup. Requires you to have access to the workgroup in which the queries ran. func (c *Client) Options() Options Options returns a copy of the client configuration. Callers SHOULD NOT perform mutations on any inner structures within client config. Config overrides should instead be made on a per-operation basis through functional options. Submits calculations for execution within a session. You can supply the code to run as an inline code block within the request. The request syntax requires the StartCalculationExecutionRequest$CodeBlock parameter or the CalculationConfiguration$CodeBlock parameter, but not both. Because CalculationConfiguration$CodeBlockis deprecated, use the StartCalculationExecutionRequest$CodeBlock parameter instead. Requests the cancellation of a calculation. A StopCalculationExecution call on a calculation that is already in a terminal state (for example, STOPPED , FAILED , or COMPLETED) succeeds but has no effect. Cancelling a calculation is done on a best effort basis. If a calculation cannot be cancelled, you can be charged for its completion. If you are concerned about being charged for a calculation that cannot be cancelled, consider terminating the session in which the calculation is running. Adds one or more tags to an Athena resource. A tag is a label that you assign to a resource. Each tag consists of a key and an optional value, both of which you define. For example, you can use tags to categorize Athena workgroups, data catalogs, or capacity reservations by purpose, owner, or environment. Use a consistent set of tag keys to make it easier to search and filter the resources in your account. For best practices, see Tagging Best Practices. Tag keys can be from 1 to 128 UTF-8 Unicode characters, and tag values can be from 0 to 256 UTF-8 Unicode characters. Tags can use letters and numbers representable in UTF-8, and the following characters: + - = . _ : / @. Tag keys and values are case-sensitive. Tag keys must be unique per resource. If you specify more than one tag, separate them by commas. Terminates an active session. A TerminateSession call on a session that is already inactive (for example, in a FAILED , TERMINATED or TERMINATING state) succeeds but has no effect. Calculations running in the session when TerminateSession is called are forcefully stopped, but may display as FAILED instead of STOPPED . type CreateNotebookInput struct { Name *string WorkGroup *string ClientRequestToken *string } type CreatePreparedStatementInput struct { QueryStatement *string StatementName *string WorkGroup *string Description *string } type CreatePresignedNotebookUrlInput struct { SessionId *string } type CreateWorkGroupOutput struct { ResultMetadata middleware.Metadata } type DeleteCapacityReservationInput struct { Name *string } type DeleteDataCatalogInput struct { Name *string DeleteCatalogOnly bool } type DeleteNamedQueryInput struct { NamedQueryId *string } type DeleteNamedQueryOutput struct { ResultMetadata middleware.Metadata } type DeleteNotebookInput struct { NotebookId *string } type DeleteNotebookOutput struct { ResultMetadata middleware.Metadata } type DeletePreparedStatementInput struct { StatementName *string WorkGroup *string } type DeleteWorkGroupInput struct { WorkGroup *string RecursiveDeleteOption *bool } type DeleteWorkGroupOutput struct { ResultMetadata middleware.Metadata } EndpointParameters provides the parameters that influence how endpoints are resolved. EndpointResolver interface for resolving service endpoints. EndpointResolverFromURL returns an EndpointResolver configured using the provided endpoint url. By default, the resolved endpoint resolver uses the client region as signing region, and the endpoint source is set to EndpointSourceCustom.You can provide functional options to configure endpoint values for the resolved endpoint. type EndpointResolverOptions = internalendpoints.Options EndpointResolverOptions is the service endpoint resolver options type ExportNotebookInput struct { NotebookId *string } type GetCalculationExecutionCodeInput struct { CalculationExecutionId *string } type GetCalculationExecutionInput struct { CalculationExecutionId *string } type GetCalculationExecutionStatusInput struct { CalculationExecutionId *string } type GetCapacityAssignmentConfigurationInput struct { CapacityReservationName *string } type GetCapacityReservationInput struct { Name *string } type GetDataCatalogInput struct { Name *string WorkGroup *string } type GetDatabaseInput struct { CatalogName *string DatabaseName *string WorkGroup *string } type GetNamedQueryInput struct { NamedQueryId *string } type GetNotebookMetadataInput struct { NotebookId *string } type GetPreparedStatementInput struct { StatementName *string WorkGroup *string } type GetQueryExecutionInput struct { QueryExecutionId *string } type GetQueryResultsInput struct { QueryExecutionId *string MaxResults *int32 NextToken *string } type GetQueryResultsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } GetQueryResultsPaginatorOptions is the paginator options for GetQueryResults type GetQueryRuntimeStatisticsInput struct { QueryExecutionId *string } type GetSessionInput struct { SessionId *string } type GetSessionStatusInput struct { SessionId *string } type GetWorkGroupInput struct { WorkGroup *string } type HTTPClient interface { Do(*http.Request) (*http.Response, error) } type IdempotencyTokenProvider interface { GetIdempotencyToken() (string, error) } IdempotencyTokenProvider interface for providing idempotency token type ListApplicationDPUSizesInput struct { MaxResults *int32 NextToken *string } type ListApplicationDPUSizesPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListApplicationDPUSizesPaginatorOptions is the paginator options for ListApplicationDPUSizes type ListCalculationExecutionsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListCalculationExecutionsPaginatorOptions is the paginator options for ListCalculationExecutions type ListCapacityReservationsInput struct { MaxResults *int32 NextToken *string } type ListCapacityReservationsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListCapacityReservationsPaginatorOptions is the paginator options for ListCapacityReservations type ListDataCatalogsInput struct { MaxResults *int32 NextToken *string WorkGroup *string } type ListDataCatalogsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListDataCatalogsPaginatorOptions is the paginator options for ListDataCatalogs type ListDatabasesPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListDatabasesPaginatorOptions is the paginator options for ListDatabases type ListEngineVersionsInput struct { MaxResults *int32 NextToken *string } type ListEngineVersionsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListEngineVersionsPaginatorOptions is the paginator options for ListEngineVersions type ListExecutorsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListExecutorsPaginatorOptions is the paginator options for ListExecutors type ListNamedQueriesInput struct { MaxResults *int32 NextToken *string WorkGroup *string } type ListNamedQueriesPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListNamedQueriesPaginatorOptions is the paginator options for ListNamedQueries type ListNotebookSessionsInput struct { NotebookId *string MaxResults *int32 NextToken *string } type ListPreparedStatementsInput struct { WorkGroup *string MaxResults *int32 NextToken *string } type ListPreparedStatementsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListPreparedStatementsPaginatorOptions is the paginator options for ListPreparedStatements type ListQueryExecutionsInput struct { MaxResults *int32 NextToken *string WorkGroup *string } type ListQueryExecutionsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListQueryExecutionsPaginatorOptions is the paginator options for ListQueryExecutions type ListSessionsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListSessionsPaginatorOptions is the paginator options for ListSessions type ListTableMetadataAPIClient interface { } ListTableMetadataAPIClient is a client that implements the ListTableMetadata operation. type ListTableMetadataPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListTableMetadataPaginatorOptions is the paginator options for ListTableMetadata type ListTagsForResourceInput struct { ResourceARN *string MaxResults *int32 NextToken *string } type ListTagsForResourcePaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListTagsForResourcePaginatorOptions is the paginator options for ListTagsForResource type ListWorkGroupsInput struct { MaxResults *int32 NextToken *string } type ListWorkGroupsPaginatorOptions struct { Limit int32 StopOnDuplicateToken bool } ListWorkGroupsPaginatorOptions is the paginator options for ListWorkGroups type PutCapacityAssignmentConfigurationOutput struct { ResultMetadata middleware.Metadata } type StopCalculationExecutionInput struct { CalculationExecutionId *string } type StopQueryExecutionInput struct { QueryExecutionId *string } type StopQueryExecutionOutput struct { ResultMetadata middleware.Metadata } type TagResourceInput struct { ResourceARN *string Tags []types.Tag } type TagResourceOutput struct { ResultMetadata middleware.Metadata } type TerminateSessionInput struct { SessionId *string } type UntagResourceInput struct { ResourceARN *string TagKeys []string } type UntagResourceOutput struct { ResultMetadata middleware.Metadata } type UpdateCapacityReservationInput struct { Name *string TargetDpus *int32 } type UpdateDataCatalogOutput struct { ResultMetadata middleware.Metadata } type UpdateNamedQueryOutput struct { ResultMetadata middleware.Metadata } type UpdateNotebookMetadataInput struct { Name *string NotebookId *string ClientRequestToken *string } type UpdateNotebookOutput struct { ResultMetadata middleware.Metadata } type UpdatePreparedStatementInput struct { QueryStatement *string StatementName *string WorkGroup *string Description *string } type UpdateWorkGroupOutput struct { ResultMetadata middleware.Metadata } Our objective is to create a secure Amazon API Gateway, AWS Lambda function (Python 3), Amazon Athena. When the end-user invokes the API end-point, API Gateway will do the basic request validations, and then the respective lambda function will be invoked. The lambda function will use the python boto3 library to interact with Amazon Athena tables, and the result will send as API response.What is Amazon API Gateway ?Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. APIs act as the “front door” for applications to access data, business logic, or functionality from your backend services. Using API Gateway, you can create RESTful APIs and WebSocket APIs that enable real-time two-way communication applications. API Gateway supports containerized and serverless workloads, as well as web applications.To know more about Amazon API GatewayWhat is AWS Lambda?AWS Lambda is an event-driven, serverless computing platform provided by Amazon as a part of Amazon Web Services. It is a computing service that runs code in response to events and automatically manages the computing resources required by that code.To know more about AWS Lambda and FAQsWhat is Amazon Athena?Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run.Athena is easy to use. Simply point to your data in Amazon S3, define the schema, and start querying using standard SQL. Most results are delivered within seconds. With Athena, there’s no need for complex ETL jobs to prepare your data for analysis. This makes it easy for anyone with SQL skills to quickly analyze large-scale datasets.To know more about Amazon AthenaNow, it’s time to create an APIPrerequisitesYou should have an AWS account and basic knowledge of CloudBasic knowledge of PythonInstall Postman in your machine to test the APILet’s prepare the backend environment first.Sample data source (Human Resources.csv)in S3For this demonstration, I have downloaded a sample Human Resources CSV file online and upload the same into the S3 bucket.Now, create a table in Athena.Create table in Athena from S3Navigate to Athena console, select option creates a table from left side pane and then choose “From S3 bucket data.”Athena table creation configurationThe page will open with a few options. First, create a new database if you want, else select an existing one. For this demonstration, I will choose the default database. And give some meaningful text to identify the table name. Then, Copy the S3 Location URI and paste it into the “Location of Input data set” text box. Then click on the Next button.URI: s3://aws-tech-talks/athena_demo/Note: Don’t give the S3 file URI; we need to provide the location, not the filenameOn the Data Format page, choose the CSV option, then click Next.In the Columns page, give the column you need to add in the data source data. Then click on Next.On the Partition page, Since the dataset is small, I’m not going to create any partition for this. Click Create Table.AWS Athena will create a SQL create statement query and execute it on your behalf.Now we have a table in Athena.Create Lambda FunctionBefore creating the lambda function, you need to create an I’m role, which will give access to execute the query in Athena from the Lambda function.{ "Version": "2012-10-17", "Statement": [{ "Sid": "VisualEditor0", "Effect": "Allow", "Action": ["athena:GetWorkGroup", "s3:PutObject", "s3:GetObject", "athena:StartQueryExecution", "s3:AbortMultipartUpload", "lambda:InvokeFunction", "athena:CancelQueryExecution", "athena:StopQueryExecution", "athena:GetQueryExecution", "athena:GetQueryResults", "s3:ListMultipartUploadParts"], "Resource": ["arn:aws:athena:*:123456789012:workgroup/primary", "arn:aws:s3:::aws-tech-talks/*","arn:aws:lambda:*:123456789012:function:aws_tech_talks_apigw"] }, { "Sid": "VisualEditor1", "Effect": "Allow", "Action": "athena:ListWorkGroups", "Resource": "*" }, { "Sid": "VisualEditor2", "Effect": "Allow", "Action": ["s3:ListBucket", "s3:GetBucketLocation"], "Resource": "arn:aws:s3:::aws-tech-talks" }]}Update the unique 12 digit account number in this policy and attach it to your lambda function.Copy-paste the below python code to execute the Athena SQL query from the Lambda function.SummaryFrom this article, we have created the Amazon Athena table from the S3 CSV file and, using the AWS Lambda function, executed the SQL query to fetch the results from the Athena table.GitHub link: