

Graph Engine Service

Developer Guide

Issue	01
Date	2025-08-19



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

- 1 Overview..... 1
- 2 Using the Management Plane SDK..... 3
- 3 Using the Service Plane SDK..... 5
 - 3.1 Downloading and Installing the SDK..... 5
 - 3.2 Managing Connections..... 6
 - 3.3 Obtaining Initialization Parameters..... 7
 - 3.4 Java SDK..... 9
 - 3.4.1 Initializing the Client..... 9
 - 3.4.1.1 Authentication Mode..... 9
 - 3.4.1.2 Client Connection Parameters..... 11
 - 3.4.2 Usage Process..... 11
 - 3.4.3 Java SDK Sample Reference..... 16
 - 3.4.3.1 Memory Edition Sample..... 16
 - 3.4.3.1.1 Vertex Operations..... 16
 - 3.4.3.1.2 Edge Operations..... 16
 - 3.4.3.1.3 Metadata Operations..... 17
 - 3.4.3.1.4 Index Operations..... 17
 - 3.4.3.1.5 Query Languages..... 18
 - 3.4.3.1.6 Algorithms..... 18
 - 3.4.3.1.7 Path..... 19
 - 3.4.3.1.8 Graph Statistics..... 19
 - 3.4.3.1.9 Graph Operations..... 20
 - 3.4.3.1.10 Subgraph Operations..... 20
 - 3.4.3.1.11 Job Management..... 20
 - 3.4.3.1.12 Custom Operations..... 20
 - 3.4.3.1.13 Filtered-query..... 21
 - 3.4.3.1.14 Updating/Deleting Data by File..... 21
 - 3.4.3.2 Database Edition Sample..... 21
 - 3.4.3.2.1 Vertex Operations..... 21
 - 3.4.3.2.2 Edge Operations..... 21
 - 3.4.3.2.3 Metadata Operations..... 22
 - 3.4.3.2.4 Index Operations..... 22

- 3.4.3.2.5 Algorithms..... 22
- 3.4.3.2.6 Graph Statistics..... 23
- 3.4.3.2.7 Graph Operations..... 23
- 3.4.3.2.8 Job Management..... 23
- 3.5 Python SDK..... 23
 - 3.5.1 Initializing the Client..... 23
 - 3.5.1.1 Authentication Mode..... 23
 - 3.5.1.2 Client Connection Parameters..... 25
 - 3.5.2 Usage Process..... 25
 - 3.5.3 Python SDK Sample Reference..... 30
 - 3.5.3.1 Memory Edition Sample..... 30
 - 3.5.3.1.1 Vertex Operations..... 30
 - 3.5.3.1.2 Edge Operations..... 30
 - 3.5.3.1.3 Metadata Operations..... 31
 - 3.5.3.1.4 Index Operations..... 31
 - 3.5.3.1.5 Query Languages..... 32
 - 3.5.3.1.6 Path..... 32
 - 3.5.3.1.7 Graph Statistics..... 32
 - 3.5.3.1.8 Graph Operations..... 32
 - 3.5.3.1.9 Subgraph Operations..... 32
 - 3.5.3.1.10 Job Management..... 33
 - 3.5.3.1.11 Custom Operations..... 33
 - 3.5.3.1.12 Filtered-query..... 33
 - 3.5.3.2 Database Edition Sample..... 33
 - 3.5.3.2.1 Vertex Operations..... 33
 - 3.5.3.2.2 Edge Operations..... 34
 - 3.5.3.2.3 Metadata Operations..... 34
 - 3.5.3.2.4 Index Operations..... 34
 - 3.5.3.2.5 Algorithms..... 35
 - 3.5.3.2.6 Graph Statistics..... 35
 - 3.5.3.2.7 Graph Operations..... 35
 - 3.5.3.2.8 Job Management..... 35
- 4 Using Cypher JDBC Driver to Access GES..... 37**

1 Overview

Service Overview

Graph Engine Service (GES) facilitates query and analysis of graph-structured data based on various relationships. It is specifically suited for scenarios that require analyzing rich relationship data, including social relationship analysis, recommendations, precision marketing, public opinion and social listening, information communication, and fraud detection.

Developer Guide Overview

The GES Software Development Kit (SDK) encapsulates RESTful APIs offered by GES, streamlining your development process. You can directly call the APIs provided by the GES SDK to utilize GES functions.

Similar to GES APIs, GES SDKs are classified into management plane SDKs and service plane SDKs. The differences are:

SDK	Open Source (Downloadable Using Maven or Pip)	Generate Code Dynamically Using API Explorer	Supported Languages	Current Version
Management-plane SDK	Yes	Yes	Java, Python, Go, Node.js, and PHP.	v1 and v2
Service-plane SDK	No (can only be downloaded from the Connection Management page of the GES console)	No (see the sample code in the SDK package)	Java and Python, with support for other languages being gradually rolled out in the future.	v1

Content Navigation

The GES developer guide will assist you in installing and configuring the development environment, as well as conducting secondary development by utilizing the interface functions provided by the GES SDK.

Chapter	Describes
Overview	The concepts of this service and its developer guide.
Using the Management Plane SDK	Common operations using the management-plane GES SDK.
Using the Service Plane SDK	Common operations using the service-plane GES SDK.
Using Cypher JDBC Driver to Access GES	Accessing service-plane graph instances and executing Cypher queries with JDBC.

2 Using the Management Plane SDK

Obtaining SDK Installation Information

1. Access **API Explorer**. In the upper left corner, select **Graph Engine Service** from the drop-down list.
2. Locate an API in the list on the left, click the **Sample Code** tab on the right, and click **View SDK Details**. In this way, you can obtain installation instructions and usage guidelines.

Figure 2-1 Viewing sample code

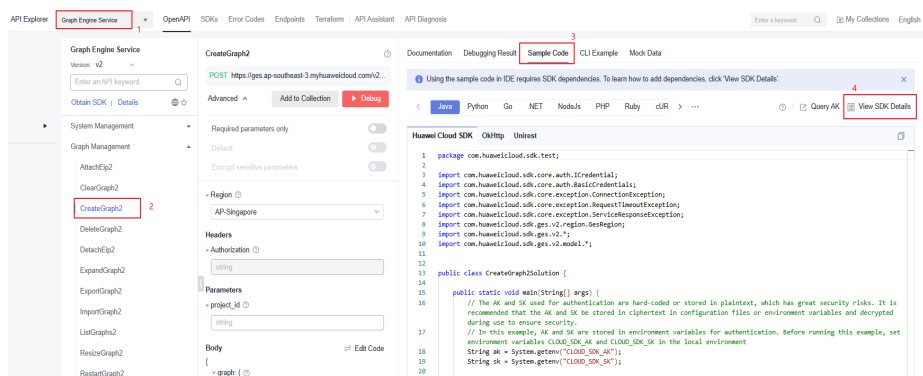
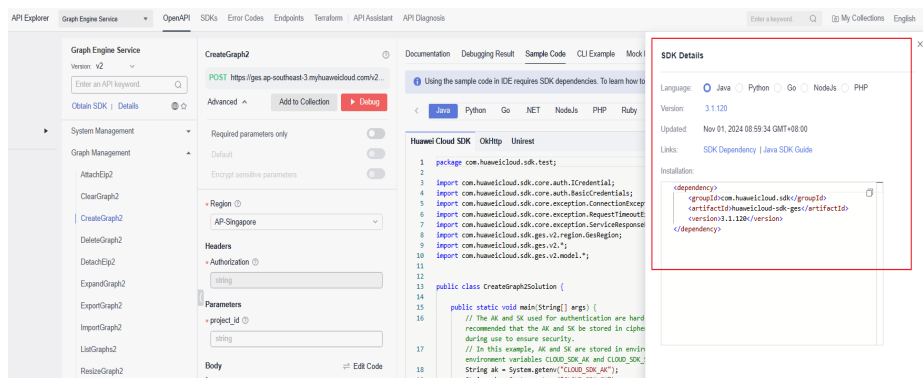


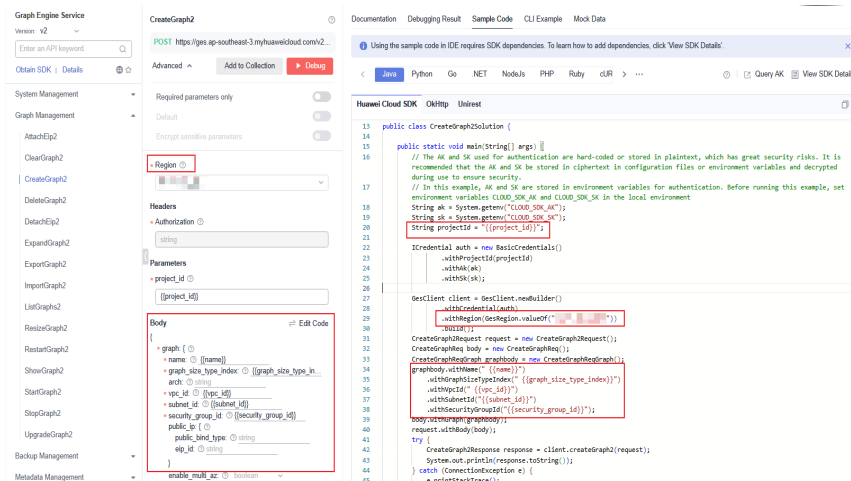
Figure 2-2 SDK information



Generating SDK Sample Code

1. Access **API Explorer**. In the upper left corner, select **Graph Engine Service** from the drop-down list.
2. Select **v1** or **v2** (recommended) for **Version**. Locate a desired API in the list on the left and set the parameters in the **Region**, **Parameters**, and **Body** areas. The **Sample Code** tab will dynamically generate SDK sample code corresponding to the API.

Figure 2-3 Generating SDK sample code



3 Using the Service Plane SDK

3.1 Downloading and Installing the SDK

The service plane SDK supports two programming languages: Java and Python. For Java SDK, there are two scenarios:

- Maven source available: A Maven source is configured and can download JAR files from open source repositories.
- Maven source unavailable: There is no Maven source configured or it cannot download JAR files from open source repositories.

Downloading the SDK

Download the SDK and driver. For details, see [Managing Connections](#). The SDK package named **huaweicloud-ges-sdk-java-xxx.zip** is obtained.

Installing the Java SDK When the Maven Source Is Available

1. Extract **huaweicloud-ges-sdk-java-xxx.zip**, go to the **maven-install** directory, and run the **ges-sdk-java-maven-install.bat** or **ges-sdk-java-maven-install.sh** file to install **graph-sdk-xxx.jar** and **cypher-jdbc-driver-xxx.jar** to the local Maven repository.
2. Create a Maven project and add the following dependency to the POM file. Use Maven to download JAR files on which the SDK depends from other open source repositories.

```
<dependency>
  <groupId>com.huawei.ges.graph</groupId>
  <artifactId>graph-sdk</artifactId>
  <version>xxx</version> // Enter the version number of the current service-plane SDK.
</dependency>
```

Installing the Java SDK When the Maven Source Is Unavailable

Create a project, extract **huaweicloud-ges-sdk-java-xxx.zip**, and import **graph-sdk-xxx-jar-with-dependencies.jar** in the **jars** directory to the project or import all packages in the **graph-sdk-xxx.jar** and **graph-sdk-lib** directories to the project.

Installing the Python SDK

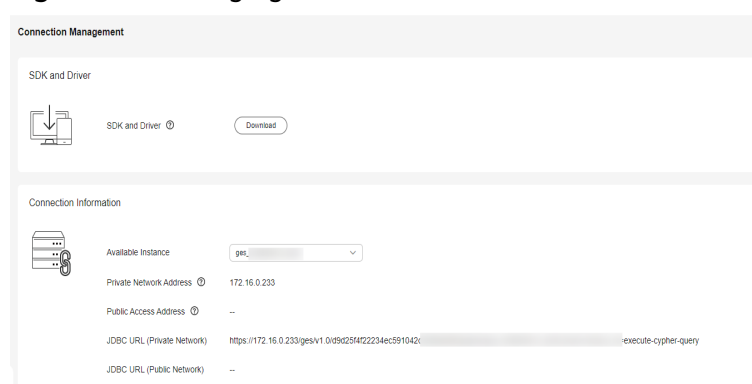
Extract **huaweicloud-ges-sdk-java-xxx.zip**, go to the **huaweicloud-ges-sdk-java-xxx** directory, and run the **pip install** command to install the **.whl** file in the **pip_install** directory. Then, you can use the Python SDK locally.

3.2 Managing Connections

You can download the required SDK and driver and view the connection information of the graph instance.

Log in to the GES management console. In the navigation pane on the left, choose **Connection Management**.

Figure 3-1 Managing connections

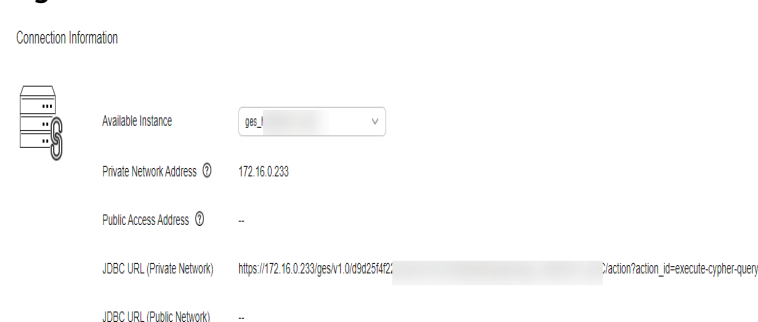


Downloading the SDK and Driver

Click **Download**.

- The SDK encapsulates service plane APIs. You are advised to use the SDK to access graph instances.
- You need to download the Cypher-JDBC driver for Cypher API access. For details, see [Using Cypher JDBC Driver to Access GES](#).

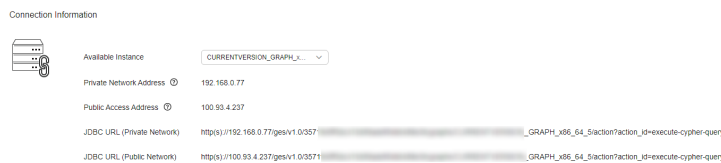
Figure 3-2 SDK and driver



Connection Information

Select a created graph instance for **Available Instance** to view the following information:

- **Private Network Address:** ECSs on the same private network can connect to the graph instance using the private network address.
- **Public Access Address:** You can use a public access address (EIP) to access the graph instance from the Internet. You can also bind an EIP to or unbind one from a graph instance.
- **JDBC URL (Private Network):** Use this configuration when the JDBC driver executor and the graph instance are on the same private network.
- **JDBC URL (Public Network):** Use this configuration when the JDBC driver executor can access the graph instance (with an EIP bound) from the Internet.

Figure 3-3 Instance information

3.3 Obtaining Initialization Parameters

Authentication Mode

When using the service-plane SDK, you need to prepare the necessary parameters for client initialization in advance. GES offers two authentication modes, and you can choose either one.

- **Token-based authentication:** Obtain a token authentication request using a username and password.
- **AK/SK-based authentication:** Requests are authenticated by encrypting the request body using an AK/SK.

AK/SK-based authentication is recommended as it provides higher security.

Obtaining Initialization Parameters

Table 3-1 Initialization parameters

Parameter	Mandatory	Description	Source
graphEndPoint	Yes	Graph access address	Log in to the GES console. In the navigation pane on the left, choose Graph Management . The value of this parameter is the Internal Access Address or External Access Address value of the graph you want to access.
graphName	Yes	Graph name	The value of this parameter is the Name/ID value of the graph you want to access on the Graph Management page of the GES console.

Parameter	Mandatory	Description	Source
ak	No. Mandatory for AK/SK-based authentication.	Access key	1. Log in to the management console. In the upper right corner of the page, hover over the username and choose My Credentials from the drop-down list. The My Credentials page is displayed. 2. In the navigation pane on the left, choose Access Keys .
sk	No. Mandatory for AK/SK-based authentication.	Secret key used together with the access key ID	
regionCode	No. Mandatory for AK/SK-based authentication.	Project the graph belongs to	1. Log in to the management console. In the upper right corner of the page, hover over the username and choose My Credentials from the drop-down list. The My Credentials page is displayed. 2. View the project ID in the project list.
domainName	No. Mandatory for password-based authentication.	Domain name	IAM account used to log in to the management console
userName	No. Mandatory for password-based authentication.	Username	IAM username used to log in to the management console
password	No. Mandatory for password-based authentication.	Password	Password of the IAM username used to log in to the management console

Parameter	Mandatory	Description	Source
projectId	No. Mandatory for password-based authentication.	Project the graph belongs to	1. Log in to the management console. In the upper right corner of the page, hover over the username and choose My Credentials from the drop-down list. The My Credentials page is displayed. 2. View the project ID in the project list.
iamEndPoint	No. Mandatory for password-based authentication.	Endpoint of the IAM service	You can obtain the region where the service is deployed by referring to Regions and Endpoints .
authToken	No. Mandatory for token-based authentication.	User token	You can obtain the token by calling the Obtaining a User Token API.

3.4 Java SDK

3.4.1 Initializing the Client

3.4.1.1 Authentication Mode

There are three ways to initialize the client based on different authentication methods. You can choose one according to your needs.

AK/SK-based Authentication

For how to set **ak**, **sk**, **regionCode**, and **graphEndpoint**, see [Obtaining Initialization Parameters](#).

```
import com.huawei.ges.graph.v1.GESGraphClient; // Memory edition client
import com.huawei.ges.graph.v1.persistence.GESGraphPersistenceClient; // Database edition client
import com.huawei.ges.graph.v1.auth.aksk.GesGraphAkSkCredentials;
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.http.HttpConfig;
import java.util.Arrays;
```

```
// Hard-coded or plaintext AK and SK are insecure. So, encrypt your AK and SK and store them in the
// configuration file or environment variables.
// In this example, the AK and SK are stored in environment variables. Before running this example, set
```

```
environment variables HUAWEICLOUD_SDK_AK and HUAWEICLOUD_SDK_SK.
String ak = System.getenv("HUAWEICLOUD_SDK_AK");
String sk = System.getenv("HUAWEICLOUD_SDK_SK");
String regionCode = "";
String graphEndpoint = "";
ICredential auth = new GesGraphAkSkCredentials().withAk(ak).withSk(sk).withRegionCode(regionCode);
HttpConfig httpConfig = HttpConfig.getDefaultHttpConfig();

// Memory edition client
GESGraphClient gesGraphClient =
GESGraphClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).withHttpCo
nfig(httpConfig).build();
// Database edition client
GESGraphPersistenceClient gesGraphPersistenceClient =
GESGraphPersistenceClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).
withHttpConfig(httpConfig).build();
```

Password-based Authentication

For how to set **domainName**, **userName**, **password**, **projectId**, **iamEndPoint**, and **graphEndpoint**, see [Obtaining Initialization Parameters](#).

```
import com.huawei.ges.graph.v1.GESGraphClient; // Memory edition client
import com.huawei.ges.graph.v1.persistence.GESGraphPersistenceClient; // Database edition client
import com.huawei.ges.graph.v1.auth.password.GesGraphPasswordCredentials;
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.http.HttpConfig;
import java.util.Arrays;

// Hard-coded or plaintext password is insecure. So, encrypt your password and store it in the configuration
// file or environment variables.
// In this example, the password is stored in environment variables. Before running this example, set the
// environment variable HUAWEICLOUD_SDK_PWD.
String password = System.getenv("HUAWEICLOUD_SDK_PWD");
String domainName = "";
String userName = "";
String projectId = "";
String iamEndPoint = "";
String graphEndpoint = "";
ICredential auth = new GesGraphPasswordCredentials(userName, domainName, password, projectId,
Arrays.asList(iamEndPoint));
HttpConfig httpConfig = HttpConfig.getDefaultHttpConfig();

// Memory edition client
GESGraphClient gesGraphClient =
GESGraphClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).withHttpCo
nfig(httpConfig).build();
// Database edition client
GESGraphPersistenceClient gesGraphPersistenceClient =
GESGraphPersistenceClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).
withHttpConfig(httpConfig).build();
```

Token-based Authentication

For how to set **authToken** and **graphEndpoint**, see [Obtaining Initialization Parameters](#).

```
import com.huawei.ges.graph.v1.GESGraphClient; // Memory edition client
import com.huawei.ges.graph.v1.persistence.GESGraphPersistenceClient; // Database edition client
import com.huawei.ges.graph.v1.auth.token.GesGraphTokenCredentials;
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.http.HttpConfig;
import java.util.Arrays;

String authToken = "";
```

```
String graphEndpoint = "";
ICredential auth = new GesGraphTokenCredentials().withXAuthToken(authToken);
HttpConfig httpConfig = HttpConfig.getDefaultHttpConfig();

// Memory edition client
GESGraphClient gesGraphClient =
GESGraphClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).withHttpCo
nfig(httpConfig).build();
// Database edition client
GESGraphPersistenceClient gesGraphPersistenceClient =
GESGraphPersistenceClient.newBuilder().withCredential(auth).withEndpoints(Arrays.asList(graphEndpoint)).
withHttpConfig(httpConfig).build();
```

3.4.1.2 Client Connection Parameters

After completing the client initialization authentication modes, you can configure HTTP according to your needs.

Configuring HTTP

You can configure HTTP based on the sample code:

```
import com.huaweicloud.sdk.core.http.HttpConfig;

// Default configuration
HttpConfig config = HttpConfig.getDefaultHttpConfig();

// Timeout configuration: The default connection timeout interval is 60 seconds, which can be changed as
// required.
config.withTimeout(60)

// SSL configuration: Configure whether to skip SSL certificate verification.
config.withIgnoreSSLVerification(true);

// Custom configuration: Customize SSLSocketFactory and TrustManager.
config.withSSLSocketFactory(sslSocketFactory).withX509TrustManager(trustManager);
```

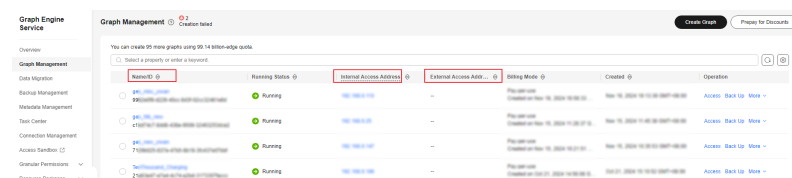
3.4.2 Usage Process

Obtain parameters and initialize the client by referring to [Obtaining Initialization Parameters](#). The following uses the memory edition's [Executing a Cypher Query](#) API as an example to demonstrate how to use the Java SDK.

Obtaining the Graph Name and Access Address

Log in to the GES console and choose **Graph Management** in the navigation pane on the left. On the displayed page, obtain the graph name (**Name/ID**) and access address (**Internal Access Address/External Access Address**) as needed.

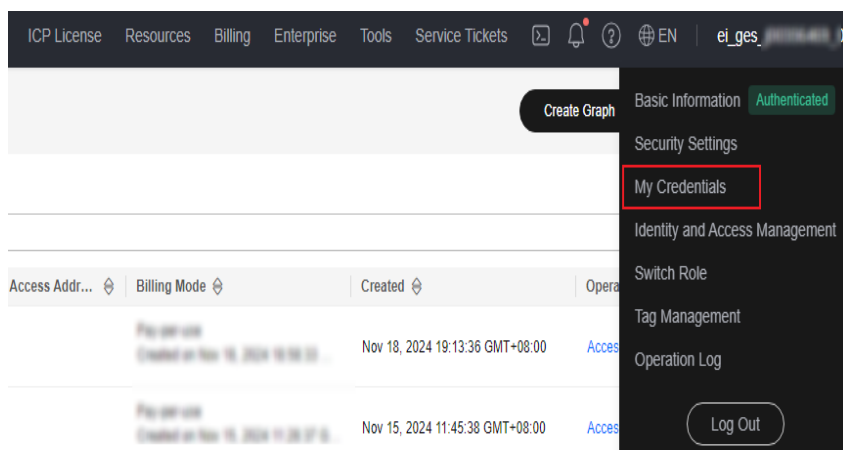
Figure 3-4 Obtaining the graph name and access address



Obtaining AK/SK-based Authentication Information

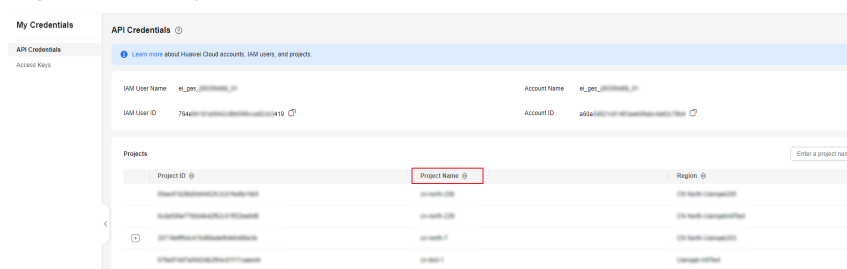
1. Log in to the Huawei Cloud management console, hover over the username in the upper right corner, and choose **My Credentials** from the drop-down list.

Figure 3-5 My Credentials



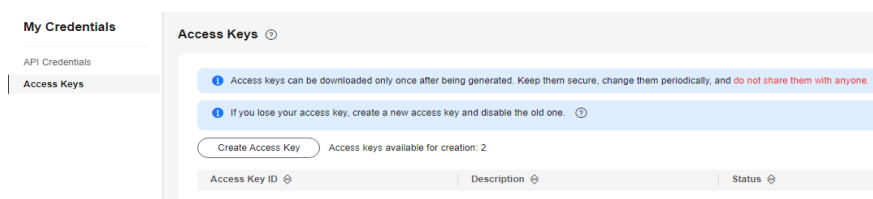
2. On the **API Credentials** page, locate the project to which the graph belongs in the project list.

Figure 3-6 Project



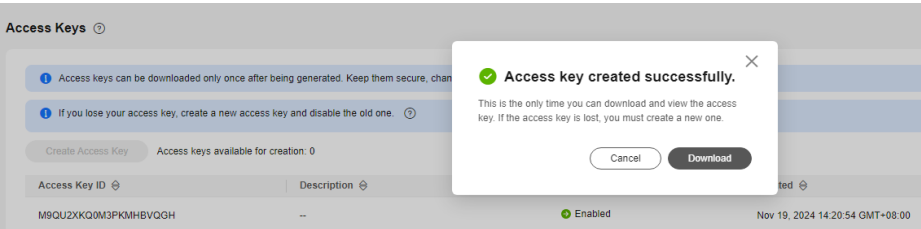
3. In the navigation pane on the left, choose **Access Keys**. On the page that appears, click **Create Access Key**.

Figure 3-7 Creating an access key



4. Once the access key is created, click **Download** to obtain the AK, SK, and region code.

Figure 3-8 Clicking Download



Obtaining Token-based Authentication Information

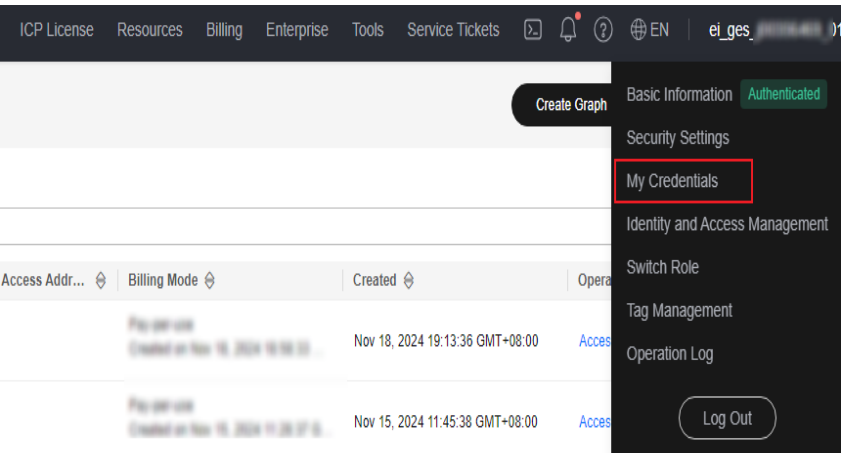
- 1. Query the IAM endpoint by referring to [Regions and Endpoints](#).

Figure 3-9 Querying a VPC endpoint

Regions and Endpoints / Identity and Access Management				
Q. Enter a region name or ID				
Region Name	Region ID	Endpoint	Protocol	
ME-Abu Dhabi-OPS	ae-ad-1	iam-ae-ad-1.myhuaweicloud.com	HTTPS	
AF-Johannesburg	af-south-1	iam-af-south-1.myhuaweicloud.com	HTTPS	
CN-Hong Kong	ap-southeast-1	iam-ap-southeast-1.myhuaweicloud.com	HTTPS	
AP-Bangkok	ap-southeast-2	iam-ap-southeast-2.myhuaweicloud.com	HTTPS	
AP-Singapore	ap-southeast-3	iam-ap-southeast-3.myhuaweicloud.com	HTTPS	
AP-Jakarta	ap-southeast-4	iam-ap-southeast-4.myhuaweicloud.com	HTTPS	
CN-East-Shanghai2	cn-east-2	iam-cn-east-2.myhuaweicloud.com	HTTPS	
CN-East-Shanghai1	cn-east-3	iam-cn-east-3.myhuaweicloud.com	HTTPS	
CN-North-Beijing1	cn-north-1	iam-cn-north-1.myhuaweicloud.com	HTTPS	
CN-North-Beijing2	cn-north-2	iam-cn-north-2.myhuaweicloud.com	HTTPS	

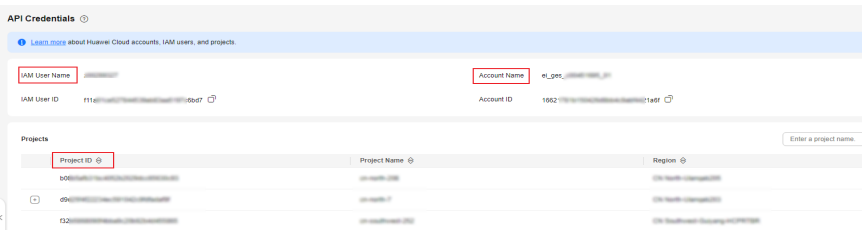
- 2. Log in to the Huawei Cloud management console, hover over the username in the upper right corner, and choose **My Credentials** from the drop-down list.

Figure 3-10 My Credentials



- 3. On the **API Credentials** page, find **IAM User Name** and **Account Name**, as well as **Project ID** in the project list.

Figure 3-11 Obtaining information



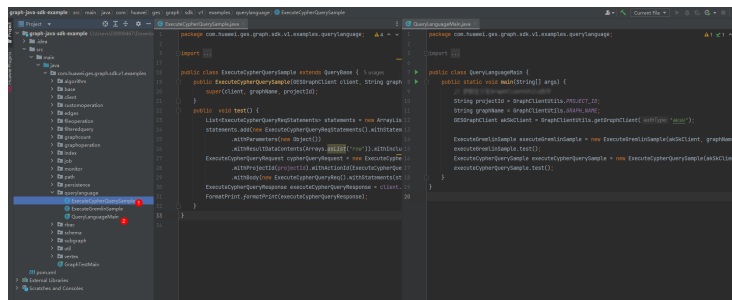
How to Use Java SDK

1. Extract **huaweicloud-ges-sdk-xxx.zip** and find a Maven project called **graph-java-sdk-example**. This project contains Java SDK sample code for GES APIs, which can be imported into an IDE for debugging by modifying API parameters.

For detailed information and parameter description of GES APIs, refer to [Service Plane APIs](#). For how to find the corresponding Java SDK sample code file for an API, refer to [Java SDK Sample Reference](#). The sample code file is located in a path that includes a Main file for debugging purposes.

2. Take the "Executing a Cypher Query" API of the memory edition as an example. Find the corresponding Java SDK sample code file in [Java SDK Sample Reference](#). The file is named **ExecuteCypherQuerySample.java**, the file is located in **com.huawei.ges.graph.sdk.v1.examples.querylanguage**. The path also includes a Main file named **QueryLanguageMain.java** for debugging.

Figure 3-12 Java SDK sample code file



3. The parameters for client initialization are defined in the **GraphClientUtils** class. Select an appropriate authentication method and set the parameters.

Figure 3-13 GraphClientUtils class



Figure 3-14 Authentication mode parameters

```

package com.huawei.ges.graph.sdk.v1.examples.util;

import ...

public class GraphClientUtils {

    // 公共静态常量
    private static final String GRAPH_ENDPOINT = ""; 3 usages
    public static final String GRAPH_NAME = "";
    public static final String PROJECT_ID = "";

    // 公共静态常量
    public static final String AK = System.getenv( name: "HUAWEICLOUD_SDK_AK"); 6 usages
    public static final String SK = System.getenv( name: "HUAWEICLOUD_SDK_SK"); 6 usages
    private static final String REGION_CODE = ""; 1 usage

    // 公共静态常量
    private static final String IAM_ENDPOINT = ""; 1 usage
    private static final String DOMAIN_NAME = ""; 1 usage
    private static final String USER_NAME = ""; 1 usage
    private static final String PASSWORD = System.getenv( name: "HUAWEICLOUD_SDK_PWD"); 1 usage

    // 公共静态常量
    private static final String TOKEN = ""; 1 usage
}

```

4. The following is an example of the result of executing a Cypher query:

Figure 3-15 Cypher query result

```

    {
      "results": {
        "data": {
          "meta": {
            "id": "test",
            "type": "node",
            "labels": ["SN"],
            "row": {
              "SN": {
                "dc_create_time": null,
                "sn_id": null,
                "sn_dc_update_time": null,
                "part_code": null
              }
            },
            "columns": ["n"],
            "errors": {}
          }
        }
      }
    }
  }
}

```

For SDK sample code for more APIs, see [Java SDK Sample Reference](#).

Specifying the Role of Nodes to Which GES Responds

By default, GES only responds to requests from the master node. To respond to requests from slave nodes, set the **requestType** parameter when initializing the request.

- When **requestType** is set to **read**, the request is load-balanced across both master and slave nodes.
- If **requestType** is set to **slave**, the request is only sent to the slave nodes.

Example as follows:

Figure 3-16 Setting the requestType parameter

```
public class ExecuteCypherQuerySample extends QueryBase { 5 usages 1
    public ExecuteCypherQuerySample(GESGraphClient client, String graphName, String projectId) { 2 usages 1
        super(client, graphName, projectId);
    }
    public void test() { 1 1
        List<ExecuteCypherQueryReqStatements> statements = new ArrayList<>();
        statements.add(new ExecuteCypherQueryReqStatements().withStatement("match (n) return n limit 1")
            .withParameters(new Object())
            .withResultDataContents(Arrays.asList("row*")).withIncludeStats(false));
        ExecuteCypherQueryRequest cypherQueryRequest = new ExecuteCypherQueryRequest().withGraphName(graphName)
            .withProjectId(projectId).withActionId(ExecuteCypherQueryRequest.ActionIdEnum.EXECUTE_CYPHER_QUERY)
            .withRequestType("read")
            .withBody(new ExecuteCypherQueryReq().withStatements(statements));
        ExecuteCypherQueryResponse executeCypherQueryResponse = client.executeCypherQuery(cypherQueryRequest);
        FormatPrint.formatPrint(executeCypherQueryResponse);
    }
}
```

 NOTE

Currently, only read requests of the memory edition support the **requestType** parameter.

3.4.3 Java SDK Sample Reference

3.4.3.1 Memory Edition Sample

3.4.3.1.1 Vertex Operations

Sample Class

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.vertex	VertexsFilterQuerySample	Querying Vertices Based on Filter Criteria
	QueryVertexsDetailsSample	Querying Vertex Details
	AddVertexSample	Adding a Vertex
	DeleteVertexSample	Deleting a Vertex
	UpdateVertexPropertiesSample	Updating Vertex Properties
	BatchVertexsQuerySample	Batch Querying Vertex Data
	BatchAddVertexsSample	Batch Adding Vertices
	BatchDeleteVertexsSample	Batch Deleting Vertices
	BatchUpdateVertexsPropertiesSample	Batch Updating Vertex Properties
	AddVertexLabelSample	Adding a Vertex Label
	DeleteVertexLabelSample	Deleting a Vertex Label
	ExportFilteredVertexsSample	Exporting Filtered Vertices
	DeleteFilteredVertexsSample	Deleting Filtered Vertices

3.4.3.1.2 Edge Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.edges	EdgesFilterQuerySample	Querying Edges Based on Filter Criteria
	QueryEdgeDetailsSample	Querying Edge Details
	AddEdgeSample	Adding an Edge
	DeleteEdgeSample	Deleting an Edge

	UpdateEdgePropertiesSample	Updating Edge Properties
	BatchEdgesQuerySample	Batch Querying Edge Data
	BatchAddEdgesSample	Batch Adding Edges
	BatchDeleteEdgesSample	Batch Deleting Edges
	BatchUpdateEdgesProperties-Sample	Batch Updating Edge Properties
	ExportFilteredEdgesSample	Exporting Filtered Edges
	DeleteFilteredEdgesSample	Deleting Filtered Edges

3.4.3.1.3 Metadata Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.schema	AddLabelSample	Adding a Label
	UpdateLabelSample	Updating a Label
	QueryGraphSchemaDetail-Sample	Querying Metadata Details
	DeleteLabelSample	Deleting a Label
	BatchAddLabelSample	Batch Adding Labels
	BuildSchemaStructureSample	Generating a Schema
	QuerySchemaStructureSample	Querying a Schema

3.4.3.1.4 Index Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.index	ListIndicesSample	Querying Indexes
	CreateIndexSample	Creating an Index
	DeleteIndexSample	Deleting an Index

3.4.3.1.5 Query Languages

Sample Code File Path	Sample Code File Name	API
com.huawei.ges .graph.sdk.v1.ex amples.queryla nguage	ExecuteGremlinSample	Executing a Gremlin Query
	ExecuteCypherQuerySample	Executing a Cypher Query

3.4.3.1.6 Algorithms

Sample Code File Path	Sample Code File Name	API
com.huawei.ges .graph.sdk.v1.ex amples.algorith m	PagerankSample	PageRank
	PersonalrankSample	PersonalRank
	KcoreSample	K-core
	KhopSample	K-Hop
	ShortestPathSample	Shortest Path
	AllShortestPathsSample	All Shortest Paths
	FilteredShortestPathSample	Filtered Shortest Path
	SsspSample	Single Source Shortest Path
	ShortestPathOfVertexSets-Sample	Vertex Set Shortest Path
	NpathsSample	n-Paths
	ClosenessSample	Closeness Centrality
	LabelPropagationSample	LabelPropagation
	LouvainSample	Louvain
	LinkPredictionSample	Link Prediction
	Node2vecSample	Node2vec
	RealtimeRecommendation-Sample	Real-time Recommendation
	CommonNeighborsSample	Common Neighbors
	ConnectedComponentSam- ple	Connected Component
	DegreeCorrelationSample	Degree Correlation

	TriangleCountSample	Triangle Count
	ClusterCoefficientSample	Cluster Coefficient
	BetweennessSample	Betweenness Centrality
	EdgeBetweennessSample	Edge Betweenness Centrality
	OdBetweennessSample	Origin-Destination Betweenness Centrality
	SingleVertexCirclesDetectionSample	Circle Detection with a Single Vertex
	CommonNeighborsOfVertexSetsSample	Common Neighbors of Vertex Sets
	AllShortestPathsOfVertexSetsSample	All Shortest Paths of Vertex Sets
	FilteredCircleDetectionSample	Filtered Circle Detection
	SubgraphMatchingSample	Subgraph Matching
	FilteredAllPairsShortestPathsSample	Filtered All Pairs Shortest Paths
	TopicrankSample	TopicRank
	FilteredNPathsSample	Filtered n-Paths

3.4.3.1.7 Path

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.path	ShowPathDetailSample	Querying Path Details

3.4.3.1.8 Graph Statistics

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.graphcount	ShowGraphVersionSample	Querying the Version of a Graph
	ShowGraphSummarySample	Querying General Information About a Graph

3.4.3.1.9 Graph Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.graphoperation	ImportGraphSample	Importing a Graph
	ExportGraphSample	Exporting a Graph
	ClearGraphSample	Clearing a Graph

3.4.3.1.10 Subgraph Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.subgraph	QuerySubgraphSample	Querying a Subgraph
	SubgraphExecuteAlgorithmSample	Executing an Algorithm on a Subgraph

3.4.3.1.11 Job Management

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.job	ShowJobSample	Querying the Status of a Job
	DeleteJobSample	Canceling a Job
	ExportJobResultSample	Exporting Job Results to a File
	ListJobsSample	Listing Jobs

3.4.3.1.12 Custom Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.customoperation	ExecuteCustomActionSample	Performing a Custom Operation

3.4.3.1.13 Filtered-query

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.customoperation	FilteredQuerySample	Filtered-query
	FilteredQueryV2Sample	Filtered-query V2

3.4.3.1.14 Updating/Deleting Data by File

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.fileoperation	ImportPropertiesSample	Updating Specified Properties of Vertices and Edges by Importing a File
	DeleteByFileSample	Deleting Vertices and Edges by Reading Files

3.4.3.2 Database Edition Sample

3.4.3.2.1 Vertex Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testQueryVertexsDetails	Querying Vertex Details
	testBatchVertexsQuery	Batch Querying Vertex Data
	testBatchAddVertexs	Batch Adding Vertices
	testBatchDeleteVertexs	Batch Deleting Vertices
	testBatchUpdateVertexsProperties	Batch Updating Vertex Properties

3.4.3.2.2 Edge Operations

Sample Code File Path	Sample Method	API
-----------------------	---------------	-----

com.huawei.ges.graph.sdk.v1.examples.persistence	testQueryEdgeDetails	Querying Edge Details
	testBatchEdgesQuery	Batch Querying Edge Data
	testBatchAddEdges	Batch Adding Edges
	testBatchDeleteEdges	Batch Deleting Edges
	testBatchUpdateEdgesProperties	Batch Updating Edge Properties

3.4.3.2.3 Metadata Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testAddLabel	Adding a Label
	testUpdateLabel	Updating a Label
	testQueryGraphSchemaDetail	Querying Metadata Details
	testQueryLabel	Querying Labels

3.4.3.2.4 Index Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testListIndices	Querying Indexes
	testCreateIndex	Creating an Index
	testDeleteIndex	Deleting an Index

3.4.3.2.5 Algorithms

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testShortestPath	Shortest Path
	testShortestPathOfVertexSets	Vertex Set Shortest Path
	testCommonNeighborsOfVertexSets	Common Neighbors of Vertex Sets

3.4.3.2.6 Graph Statistics

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testShowGraphVersion	Querying the Version of a Graph
	testShowGraphSummary	Querying General Information About a Graph

3.4.3.2.7 Graph Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testImportGraphPersistence	Importing a Graph
	testClearGraphPersistence	Clearing a Graph

3.4.3.2.8 Job Management

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	waitJob	Querying the Status of a Job
	testListJobs	Listing Jobs

3.5 Python SDK

3.5.1 Initializing the Client

There are three ways to initialize the client based on different authentication methods. You can choose one according to your needs.

During client initialization, you can configure HTTP settings as needed.

3.5.1.1 Authentication Mode

AK/SK-based Authentication

For how to set **ak**, **sk**, **regionCode**, and **graphEndpoint**, see [Obtaining Initialization Parameters](#).

```
import os
from gesgraphsdk.v1.aksk_credentials import GesGraphAkSkCredentials
```

```
from gesgraphsdk.v1.gesgraph_client import GESGraphClient # Memory edition client
from gesgraphsdk.v1.persistence.gesgraphpersistence_client import GESGraphPersistenceClient # Database
edition client
from huaweicloudsdkcore.http.http_config import HttpConfig

# Hard-coded or plaintext AK and SK are insecure. So, encrypt your AK and SK and store them in the
configuration file or environment variables.
# In this example, the AK and SK are stored in environment variables. Before running this example, set
environment variables HUAWEICLOUD_SDK_AK and HUAWEICLOUD_SDK_SK.
ak = os.getenv("HUAWEICLOUD_SDK_AK")
sk = os.getenv("HUAWEICLOUD_SDK_SK")
region_code = "{regionCode}"
graph_endpoint = "{graphEndpoint}"
auth = GesGraphAkSkCredentials(ak=ak, sk=sk, region_code=region_code)
http_conf = HttpConfig.get_default_config()

# Memory edition client
ges_graph_client =
GESGraphClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_config(http_
conf).build();
# Database edition client
ges_graph_persistence_client =
GESGraphPersistenceClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_c
onfig(http_conf).build();
```

Password-based Authentication

For how to set **domainName**, **userName**, **password**, **projectId**, **iamEndPoint**, and **graphEndPoint**, see [Obtaining Initialization Parameters](#).

```
import os
from gesgraphsdk.v1.gesgraph_client import GESGraphClient # Memory edition client
from gesgraphsdk.v1.persistence.gesgraphpersistence_client import GESGraphPersistenceClient # Database
edition client
from gesgraphsdk.v1.token_credentials import GesGraphTokenCredentials
from huaweicloudsdkcore.http.http_config import HttpConfig

// Hard-coded or plaintext password is insecure. So, encrypt your password and store it in the configuration
file or environment variables.
// In this example, the password is stored in environment variables. Before running this example, set the
environment variable HUAWEICLOUD_SDK_PWD.
password = os.getenv("HUAWEICLOUD_SDK_PWD")
domain_name = "{domainName}"
user_name = "{userName}"
project_id = "{projectId}"
graph_endpoint = "{graphEndpoint}"
iam_endpoint = "{iamEndPoint}"
auth = GesGraphTokenCredentials(iam_endpoint=iam_endpoint, domain_name=domain_name,
user_name=user_name, password=password, project_id=project_id)
http_conf = HttpConfig.get_default_config()

# Memory edition client
ges_graph_client =
GESGraphClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_config(http_
conf).build()
# Database edition client
ges_graph_persistence_client =
GESGraphPersistenceClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_c
onfig(http_conf).build();
```

Token-based Authentication

For how to set **authToken** and **graphEndPoint**, see [Obtaining Initialization Parameters](#).

```
from gesgraphsdk.v1.gesgraph_client import GESGraphClient # Memory edition client
from gesgraphsdk.v1.persistence.gesgraphpersistence_client import GESGraphPersistenceClient # Database
edition client
from gesgraphsdk.v1.token_credentials import GesGraphTokenCredentials
from huaweicloudsdkcore.http.http_config import HttpConfig

graph_endpoint = "{graphEndpoint}"
token = "{authToken}"
auth = GesGraphTokenCredentials(token=token)
http_conf = HttpConfig.get_default_config()

# Memory edition client
ges_graph_client =
GESGraphClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_config(http_
conf).build()
# Database edition client
ges_graph_persistence_client
GESGraphPersistenceClient.new_builder().with_credentials(auth).with_endpoint(graph_endpoint).with_http_c
onfig(http_conf).build();
```

3.5.1.2 Client Connection Parameters

During client initialization, you can configure HTTP settings as needed.

```
import com.huaweicloud.sdk.core.http.HttpConfig;

# Default configuration
http_config = HttpConfig.get_default_config()

# Timeout configuration
# Set the connection timeout and read timeout to 120 seconds.
http_config.timeout = 120
# Set the connection timeout to 60 seconds and the read timeout to 120 seconds.
http_config.timeout = (60, 120)

# SSL configuration: Configure whether to skip SSL certificate verification.
http_config.ignore_ssl_verification = True

# Configure the server CA certificate so that the SDK can verify the server certificate.
http_config.ssl_ca_cert = '/path/to/certfile'
```

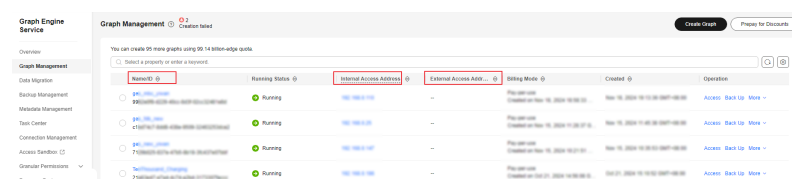
3.5.2 Usage Process

Obtain parameters and initialize the client by referring to [Obtaining Initialization Parameters](#). The following uses the memory edition's [Executing a Cypher Query](#) API as an example to demonstrate how to use the Python SDK.

Obtaining the Graph Name and Access Address

Log in to the GES console and choose **Graph Management** in the navigation pane on the left. On the displayed page, obtain the graph name (**Name/ID**) and access address (**Internal Access Address/External Access Address**) as needed.

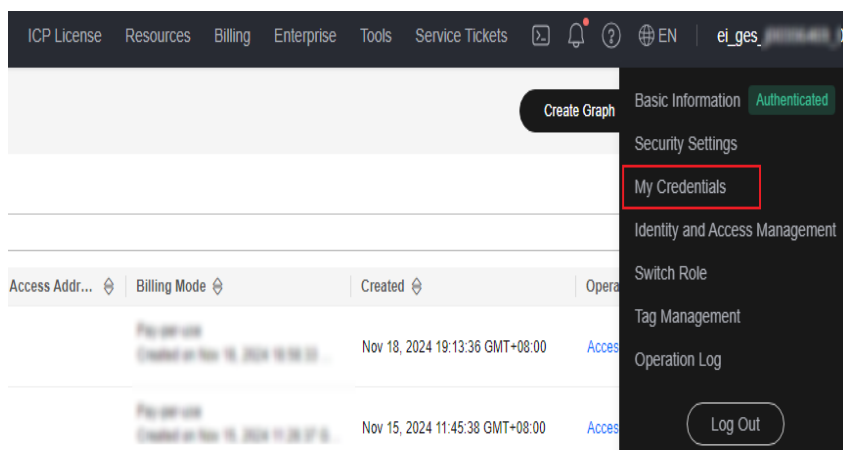
Figure 3-17 Obtaining the graph name and access address



Obtaining AK/SK-based Authentication Information

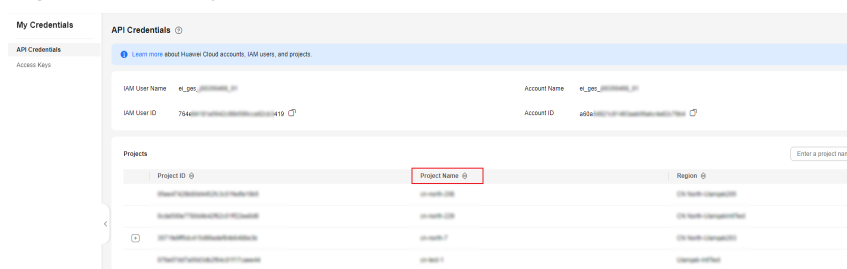
1. Log in to the Huawei Cloud management console, hover over the username in the upper right corner, and choose **My Credentials** from the drop-down list.

Figure 3-18 My Credentials



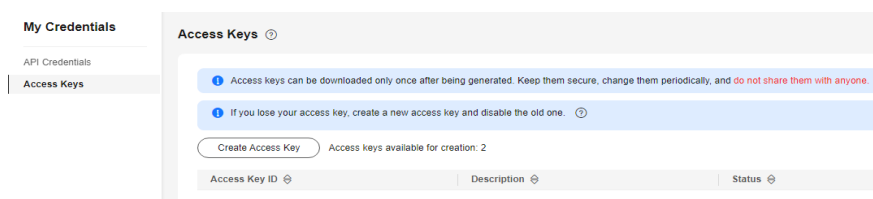
2. On the **API Credentials** page, locate the project to which the graph belongs in the project list.

Figure 3-19 Project



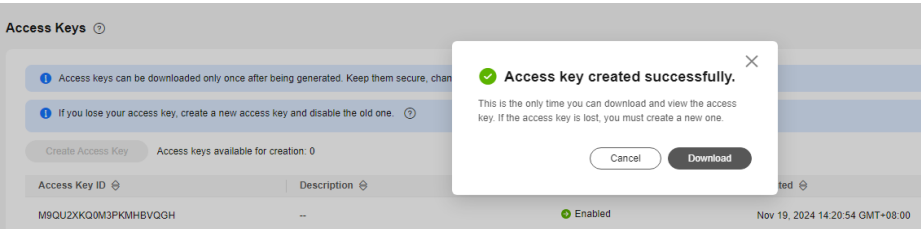
3. In the navigation pane on the left, choose **Access Keys**. On the page that appears, click **Create Access Key**.

Figure 3-20 Creating an access key



4. Once the access key is created, click **Download** to obtain the AK, SK, and region code.

Figure 3-21 Clicking Download



Obtaining Token-based Authentication Information

- 1. Query the IAM endpoint by referring to [Regions and Endpoints](#).

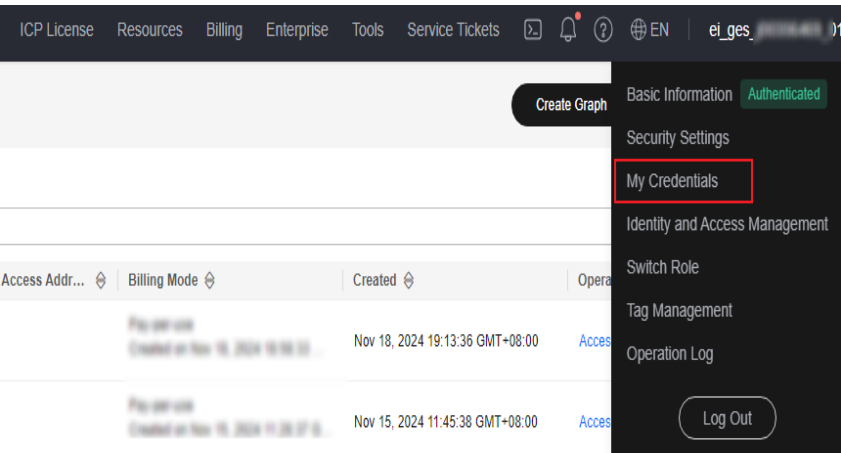
Figure 3-22 Querying a VPC endpoint

Regions and Endpoints / Identity and Access Management

Region Name	Region ID	Endpoint	Protocol
ME-Abu Dhabi-OPS	ae-sd-1	iam-ae-sd-1.myhuaweicloud.com	HTTPS
AF-Johannesburg	af-south-1	iam-af-south-1.myhuaweicloud.com	HTTPS
CN-Hong Kong	ap-southeast-1	iam-ap-southeast-1.myhuaweicloud.com	HTTPS
AP-Bangkok	ap-southeast-2	iam-ap-southeast-2.myhuaweicloud.com	HTTPS
AP-Singapore	ap-southeast-3	iam-ap-southeast-3.myhuaweicloud.com	HTTPS
AP-Jakarta	ap-southeast-4	iam-ap-southeast-4.myhuaweicloud.com	HTTPS
CN-East-Shanghai2	cn-east-2	iam-cn-east-2.myhuaweicloud.com	HTTPS
CN-East-Shanghai1	cn-east-3	iam-cn-east-3.myhuaweicloud.com	HTTPS
CN-North-Beijing1	cn-north-1	iam-cn-north-1.myhuaweicloud.com	HTTPS
CN-North-Beijing2	cn-north-2	iam-cn-north-2.myhuaweicloud.com	HTTPS

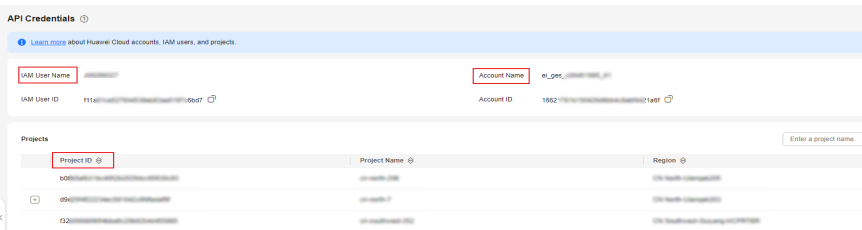
- 2. Log in to the Huawei Cloud management console, hover over the username in the upper right corner, and choose **My Credentials** from the drop-down list.

Figure 3-23 My Credentials



- 3. On the **API Credentials** page, find **IAM User Name** and **Account Name**, as well as **Project ID** in the project list.

Figure 3-24 Obtaining information



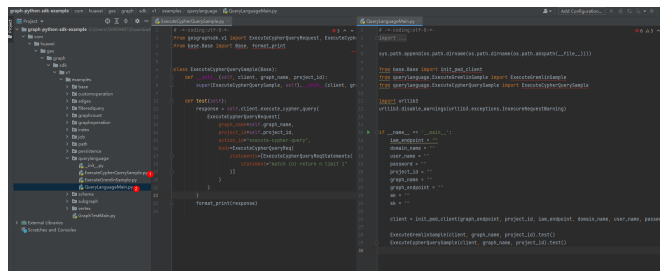
How to Use the SDK

1. Extract **huaweicloud-ges-sdk-xxx.zip** and find a Python project called **graph-python-sdk-example**. This project contains Python SDK sample code for GES APIs, which can be imported into an IDE for debugging by modifying API parameters.

For detailed information and parameter description of GES APIs, refer to [Service Plane APIs](#). For how to find the corresponding Python SDK sample code file for an API, refer to [Python SDK Sample Reference](#). The sample code file is located in a path that includes a Main file for debugging purposes.

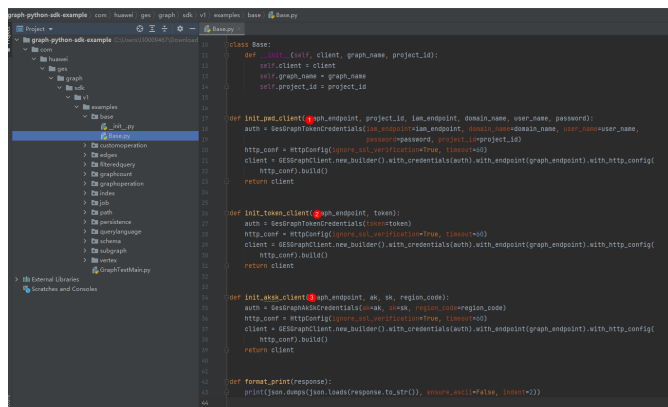
2. Take the "Executing a Cypher Query" API of the memory edition as an example. Find the corresponding Python SDK sample code file in [Python SDK Sample Reference](#). The file is named **ExecuteCypherQuerySample.py**, the file is located in **com.huawei.ges.graph.sdk.v1.examples.querylanguage**. The path also includes a Main file named **QueryLanguageMain.py** for debugging.

Figure 3-25 Python SDK sample code file

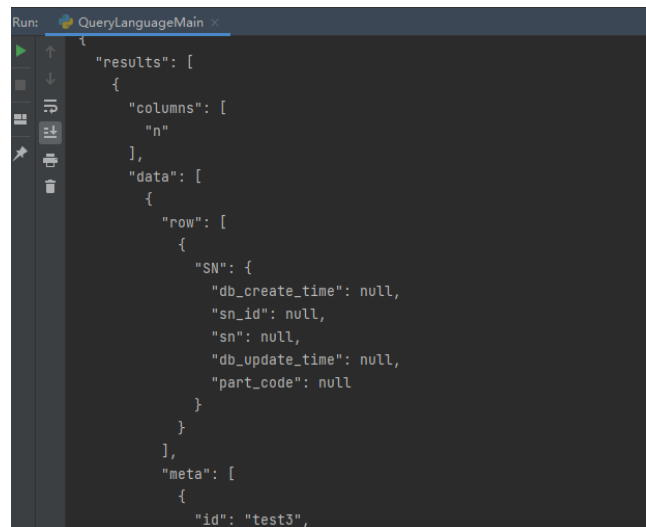


3. In the package path **com.huawei.ges.graph.sdk.v1.examples.base**, you can find the **Base.py** file that provides client initialization examples for different authentication modes. Choose the one that suits your needs.

Figure 3-26 Varying authentication modes



4. The following is an example of the result of executing a Cypher query:

Figure 3-27 Cypher query result

For SDK sample code for more APIs, see [Python SDK Sample Reference](#).

Specifying the Role of Nodes to Which GES Responds

By default, GES only responds to requests from the master node. To respond to requests from slave nodes, set the **request_type** parameter when initializing the request.

- When **request_type** is set to **read**, the request is load-balanced across both master and slave nodes.
- If **request_type** is set to **slave**, the request is only sent to the slave nodes.

Example as follows:

Figure 3-28 Setting the request_type parameter

```
class ExecuteCypherQuerySample(Base):
    def __init__(self, client, graph_name, project_id):
        super(ExecuteCypherQuerySample, self).__init__(client, graph_name, project_id)

    def test(self):
        response = self.client.execute_cypher_query(
            ExecuteCypherQueryRequest(
                request_type="read",
                graph_name=self.graph_name,
                project_id=self.project_id,
                action_id="execute-cypher-query",
                body=ExecuteCypherQueryReq(
                    statements=[ExecuteCypherQueryReqStatements(
                        statement="match (n) return n limit 1"
                    )]
                )
            )
        )
        format_print(response)
```

 NOTE

Currently, only read requests of the memory edition support the **request_type** parameter.

3.5.3 Python SDK Sample Reference

3.5.3.1 Memory Edition Sample

3.5.3.1.1 Vertex Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.vertices	VertexsFilterQuerySample	Querying Vertices Based on Filter Criteria
	QueryVertexsDetailsSample	Querying Vertex Details
	AddVertexSample	Adding a Vertex
	DeleteVertexSample	Deleting a Vertex
	UpdateVertexPropertiesSample	Updating Vertex Properties
	BatchVertexsQuerySample	Batch Querying Vertex Data
	BatchAddVertexsSample	Batch Adding Vertices
	BatchDeleteVertexsSample	Batch Deleting Vertices
	BatchUpdateVertexsPropertiesSample	Batch Updating Vertex Properties
	AddVertexLabelSample	Adding a Vertex Label
	DeleteVertexLabelSample	Deleting a Vertex Label
	ExportFilteredVertexsSample	Exporting Filtered Vertices
	DeleteFilteredVertexsSample	Deleting Filtered Vertices

3.5.3.1.2 Edge Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.edges	EdgesFilterQuerySample	Querying Edges Based on Filter Criteria
	QueryEdgeDetailsSample	Querying Edge Details
	AddEdgeSample	Adding an Edge

	DeleteEdgeSample	Deleting an Edge
	UpdateEdgePropertiesSample	Updating Edge Properties
	BatchEdgesQuerySample	Batch Querying Edge Data
	BatchAddEdgesSample	Batch Adding Edges
	BatchDeleteEdgesSample	Batch Deleting Edges
	BatchUpdateEdgesProperties-Sample	Batch Updating Edge Properties
	ExportFilteredEdgesSample	Exporting Filtered Edges
	DeleteFilteredEdgesSample	Deleting Filtered Edges

3.5.3.1.3 Metadata Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.schema	AddLabelSample	Adding a Label
	UpdateLabelSample	Updating a Label
	QueryGraphSchemaDetail-Sample	Querying Metadata Details
	DeleteLabelSample	Deleting a Label
	BatchAddLabelSample	Batch Adding Labels
	BuildSchemaStructureSample	Generating a Schema
	QuerySchemaStructureSample	Querying a Schema

3.5.3.1.4 Index Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.index	ListIndicesSample	Querying Indexes
	CreateIndexSample	Creating an Index
	DeleteIndexSample	Deleting an Index

3.5.3.1.5 Query Languages

Sample Code File Path	Sample Code File Name	API
com.huawei.ges. graph.sdk.v1.ex amples.queryla nguage	ExecuteGremlinSample	Executing a Gremlin Query
	ExecuteCypherQuerySample	Executing a Cypher Query

3.5.3.1.6 Path

Sample Code File Path	Sample Code File Name	API
com.huawei.ges. graph.sdk.v1.exa mples.path	ShowPathDetailSample	Querying Path Details

3.5.3.1.7 Graph Statistics

Sample Code File Path	Sample Code File Name	API
com.huawei.ges. graph.sdk.v1.exa mples.graphcou nt	ShowGraphVersionSample	Querying the Version of a Graph
	ShowGraphSummarySample	Querying General Information About a Graph

3.5.3.1.8 Graph Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges. graph.sdk.v1.exa mples.graphoper ation	ImportGraphSample	Importing a Graph
	ExportGraphSample	Exporting a Graph
	ClearGraphSample	Clearing a Graph

3.5.3.1.9 Subgraph Operations

Sample Code File Path	Sample Code File Name	API
-----------------------	-----------------------	-----

com.huawei.ges.graph.sdk.v1.examples.subgraph	QuerySubgraphSample	Querying a Subgraph
	SubgraphExecuteAlgorithmSample	Executing an Algorithm on a Subgraph

3.5.3.1.10 Job Management

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.job	ShowJobSample	Querying the Status of a Job
	DeleteJobSample	Canceling a Job
	ExportJobResultSample	Exporting Job Results to a File
	ListJobsSample	Listing Jobs

3.5.3.1.11 Custom Operations

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.customoperation	ExecuteCustomActionSample	Performing a Custom Operation

3.5.3.1.12 Filtered-query

Sample Code File Path	Sample Code File Name	API
com.huawei.ges.graph.sdk.v1.examples.filteredquery	FilteredQuerySample	Filtered-query
	FilteredQueryV2Sample	Filtered-query V2

3.5.3.2 Database Edition Sample

3.5.3.2.1 Vertex Operations

Sample Code File Path	Sample Method	API
-----------------------	---------------	-----

com.huawei.ges.graph.sdk.v1.examples.persistence	testQueryVertexsDetails	Querying Vertex Details
	testBatchVertexsQuery	Batch Querying Vertex Data
	testBatchAddVertexs	Batch Adding Vertices
	testBatchDeleteVertexs	Batch Deleting Vertices
	testBatchUpdateVertexsProperties	Batch Updating Vertex Properties

3.5.3.2.2 Edge Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testQueryEdgeDetails	Querying Edge Details
	testBatchEdgesQuery	Batch Querying Edge Data
	testBatchAddEdges	Batch Adding Edges
	testBatchDeleteEdges	Batch Deleting Edges
	testBatchUpdateEdgesProperties	Batch Updating Edge Properties

3.5.3.2.3 Metadata Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testAddLabel	Adding a Label
	testUpdateLabel	Updating a Label
	testQueryGraphSchemaDetail	Querying Metadata Details
	testQueryLabel	Querying Labels

3.5.3.2.4 Index Operations

Sample Code File Path	Sample Method	API
-----------------------	---------------	-----

com.huawei.ges.graph.sdk.v1.examples.persistence	testListIndices	Querying Indexes
	testCreateIndex	Creating an Index
	testDeleteIndex	Deleting an Index

3.5.3.2.5 Algorithms

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testShortestPath	Shortest Path
	testShortestPathOfVertexSets	Vertex Set Shortest Path
	testCommonNeighborsOfVertexSets	Common Neighbors of Vertex Sets

3.5.3.2.6 Graph Statistics

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testShowGraphVersion	Querying the Version of a Graph
	testShowGraphSummary	Querying General Information About a Graph

3.5.3.2.7 Graph Operations

Sample Code File Path	Sample Method	API
com.huawei.ges.graph.sdk.v1.examples.persistence	testImportGraphPersistence	Importing a Graph
	testClearGraphPersistence	Clearing a Graph

3.5.3.2.8 Job Management

Sample Code File Path	Sample Method	API
-----------------------	---------------	-----

com.huawei.ges.g raph.sdk.v1.exam ples.persistence	waitJob	Querying the Status of a Job
	testListJobs	Listing Jobs

4 Using Cypher JDBC Driver to Access GES

Introduction

The Cypher JDBC Driver is designed for GES. It is developed based on Neo4j JDBC Driver and provides a method of using JDBC to access GES and perform cypher queries.

The driver greatly reduces the CPU and memory usage for returning a large amount of data requested by high-concurrent cypher queries to avoid JVM caching of a complete request body. The driver parses a response body into streaming data instead obtaining an entire body and then parsing it.

Configuring Dependencies

1. Download the SDK and driver. For details, see [Managing Connections](#).
2. If the Maven source is available (configured with a Maven source that can download JAR files from an open source repository), decompress **huaweicloud-ges-sdk-java-xxx.zip** and go to the **maven-install** directory. From there, execute either the **ges-sdk-java-maven-install.bat** or **ges-sdk-java-maven-install.sh** file to install **graph-sdk-xxx.jar** and **cypher-jdbc-driver-xxx.jar** to your local Maven repository. This will allow you to configure your POM dependencies and use the Cypher JDBC Driver in your Maven project.

```
<dependency>
  <groupId>com.huawei.ges</groupId>
  <artifactId>cypher-jdbc-driver</artifactId>
  <version>xxx</version> // Enter the version number of the Cypher JDBC driver.
</dependency>
<dependency>
  <groupId>org.neo4j</groupId>
  <artifactId>neo4j-jdbc</artifactId>
  <version>xxx</version>
</dependency>
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpclient</artifactId>
  <version>xxx</version>
</dependency>
```

3. If the Maven source is unavailable (not configured with a Maven source or the Maven source cannot download JAR files from an open source repository), decompress **huaweicloud-ges-sdk-java-xxx.zip** and import the **cypher-jdbc-driver-xxx-with-dependencies.jar** file in the **jars** directory to the project or

import all packages in the **cypher-jdbc-driver-xxx.jar** and **cypher-jdbc-lib** directories to the project.

Parameter Description

Table 4-1 JDBC getConnection parameter description

Parameter	Description
url	The jdbc:ges: prefix is used to concatenate the URL for the GES Cypher API. On the Connection Management page of the GES console, select the name of the graph you want to access from the Available Instance drop-down list. The URL of the Cypher API is the value of <i>JDBC connection string</i>.
prop	Properties object, including configurations required for connecting to GES APIs. For details, see Table 4-2 .

Table 4-2 Properties parameter description

Parameter	Description
X-Auth-Token	Token obtained through IAM authentication.
parse-json	Whether data is converted to vertices and edges. The default value is false. - When set to false, vertices and edges in the Cypher return body are returned in map format. - When set to true, vertices and edges are returned in GesElement format.
deserializer-type	Policy for parsing Cypher statements. The options are lazy and eager, with lazy as the default. - When set to lazy, Cypher statements are parsed in streaming mode, and the Cypher response body does not reside in memory. - When set to eager, the entire JSON file is obtained for parsing.
limit	Flow rate, with a default value of 100000. The kernel returns data to the server-side web application in batches, which are then organized into a stream and sent to the frontend. limit determines the batch size when the kernel returns data to the web application. For the same query, a smaller limit value results in more interactions between the GES kernel and the web application, but the JDBC client receives the first record faster. However, this may increase the overall query time.

Authentication

GES Cypher JDBC Driver supports two authentication modes: Token-based and AK/SK-based.

1. The following is sample code for token-based authentication:

```
import com.huawei.ges.jdbc.io.model.GesElement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Properties;

public class CypherJDBCClient {

    public static void main(String[] args) throws ClassNotFoundException, IllegalAccessException,
InstantiationException {
        String token = ""; // Call the IAM API to obtain the value.
        String url = ""; // The value is jdbc:ges:JDBC connection string. You can obtain JDBC connection string from the Connection Management page of the GES console.
        Class.forName("com.huawei.ges.jdbc.Driver").newInstance();
        Properties prop = new Properties();
        prop.setProperty("X-Auth-Token", token);
        prop.setProperty("deserializer-type", "lazy");
        prop.setProperty("parse-json", "true");
        prop.setProperty("limit", "10000");
        try (Connection conn = DriverManager.getConnection(url, prop)) {
            String query = "match (m) return m limit 1";
            try (PreparedStatement stmt = conn.prepareStatement(query)) {
                try (ResultSet rs = stmt.executeQuery()) {
                    while (rs.next()) {
                        GesElement.GesVertex vertex = (GesElement.GesVertex) rs.getObject("m");
                        System.out.println(vertex.getId());
                        System.out.println(vertex.getLabels());
                        System.out.println(vertex.getProperties());
                    }
                }
            }
        } catch (SQLException e) {
            System.out.println("Execute SQL query error.");
        }
    }
}
```

2. To authenticate using an AK/SK, you can generate a signature using the methods provided by the GES service-plane SDK. For how to import the dependencies for the service-plane SDK, refer to [Downloading and Installing the SDK](#).

The following is sample code for AK/SK-based authentication. For how to obtain the AK, SK, and region code, see [Obtaining Initialization Parameters](#).

```
import com.huawei.ges.graph.v1.auth.aksk.HttpRestClient;
import com.huawei.ges.jdbc.io.model.GesElement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Map;
import java.util.Properties;

public class CypherJDBCClientByAKSK {

    public static void main(String[] args) throws ClassNotFoundException, IllegalAccessException,
InstantiationException {
        // Hard-coded or plaintext AK and SK are insecure. So, encrypt your AK and SK and store them in
        the configuration file or environment variables.
```

```
// In this example, the AK and SK are stored in environment variables. Before running this example,
set environment variables HUAWEICLOUD_SDK_AK and HUAWEICLOUD_SDK_SK.
String ak = System.getenv("HUAWEICLOUD_SDK_AK");
String sk = System.getenv("HUAWEICLOUD_SDK_SK");
String regionCode = "";
String url = ""; // The value is jdbc:ges:JDBC connection string. You can obtain JDBC connection
string from the Connection Management page of the GES console.
Map<String, String> iamHeader = HttpClient.getIamSignHeaders(ak, sk, regionCode); //
Methods provided by the GES service-plane SDK, which provides guidance on generating signatures.
Class.forName("com.huawei.ges.jdbc.Driver").newInstance();
doCypherQuery(url, iamHeader);
}

public static void doCypherQuery(String url, Map<String, String> iamHeaders) {
    Properties prop = new Properties();
    for (Map.Entry<String, String> pair : iamHeaders.entrySet()) {
        prop.setProperty(pair.getKey(), pair.getValue());
    }
    prop.setProperty("deserializer-type", "lazy");
    prop.setProperty("parse-json", "true");
    prop.setProperty("limit", "10000");
    try (Connection conn = DriverManager.getConnection(url, prop)) {
        String query = "match (m) return m limit 1";
        try (PreparedStatement stmt = conn.prepareStatement(query)) {
            try (ResultSet rs = stmt.executeQuery()) {
                while (rs.next()) {
                    GesElement.GesVertex vertex = (GesElement.GesVertex) rs.getObject("m");
                    System.out.println(vertex.getId());
                    System.out.println(vertex.getLabels());
                    System.out.println(vertex.getProperties());
                }
            }
        }
    } catch (SQLException e) {
        System.out.println("Execute SQL query error.");
    }
}
```