

Data Warehouse Service

Quick Start

Issue	13
Date	2025-10-10



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.

Contents

1 Quickly Creating a DWS Cluster and Importing Data for Query..... 1

2 Getting Started with DWS Data Development SQL.....15

3 Getting Started with DWS..... 17

1 Quickly Creating a DWS Cluster and Importing Data for Query

Scenario

DWS is a fully-managed and enterprise-level cloud data warehouse service. It is O&M-free, compatible with the PostgreSQL ecosystem, and supports online cluster scale-out and efficient loading of multiple data sources. It helps enterprises efficiently analyze and monetize massive amounts of data online.

This section describes how to create a DWS cluster with three nodes and import CSV data from an OBS bucket to perform simple data analysis and query.

NOTICE

The DWS cluster does not support cross-region access to OBS bucket data. Before creating a DWS cluster, ensure that the cluster and OBS bucket are in the same region. Use CN-Hong Kong as an example.

- **Billing Mode:** Pay-per-use
- **Version:** Coupled storage and compute
- **Nodes:** 3 nodes
- Cluster connection mode: SQL editor
- Storage capacity: 20 GB/node
- Data source: CSV data in the OBS bucket
- Cluster version: 8.1.3.x

Procedure

Table 1-1 Procedure

Step	Description
Preparations	Register a Huawei account, subscribe to Huawei Cloud, complete real-name authentication, top up the account, and plan and create VPC and subnet resources.
Step 1: Creating a DWS Cluster	Create a three-node DWS cluster and configure information such as the cluster node quantity, storage capacity, and cluster version.
Step 2: Uploading Data to OBS	Create an OBS bucket and upload the local CSV data to the bucket.
Step 3: Connecting to the DWS Cluster and Importing Data for Analysis	Use the SQL editor to connect to the DWS database, create an OBS foreign table, import OBS bucket data to the DWS cluster through the OBS foreign table, and perform simple SQL queries.

Preparations

- Sign up a DWS account, complete the real-name authentication, and check the account status before using DWS. The account cannot be in arrears or frozen.
- Obtain the AK/SK of the account. For details, see [Access Keys](#).
- Create a VPC and subnet resources.

Step 1: Creating a DWS Cluster

Step 1 Log in to the [DWS console](#) and go to the **Create DWS Cluster** page.

Step 2 Configure the parameters according to [Table 1-2](#).

Table 1-2 Parameter configurations

Parameter	Example Value	Description
Region	Select the CN-Hong Kong region. NOTE This guide uses CN-Hong Kong as an example. If you want to select another region, ensure that other cloud services are in the same region.	For lower network latency and quicker resource access, select the nearest region. After a DWS cluster is created, the region cannot be changed. Exercise caution when selecting a region. For more information, see Regions and AZs .

Parameter	Example Value	Description
Billing Mode	Pay-per-use	Yearly/Monthly and pay-per-use billing modes are supported. For long-term usage, it is recommended to opt for the yearly or monthly billing mode to avail of a greater discount. For more information, see Billing Overview .
AZ	Choose AZ1 to create a single-AZ cluster. (If the flavor is sold out, select another AZ.)	• Single AZ: All nodes in a cluster are deployed in the same AZ. • Multiple AZs: To achieve HA across multiple AZs, deploy cluster nodes in different AZs. For services requiring even higher HA, a multi-AZ setup is recommended.

Figure 1-1 Basic configurations

The screenshot displays the 'DWS Cluster' configuration page. The top section shows basic cluster information: Current Nodes (3), Total Capacity (300), Node Specifications (Standard 3.0 | 4 vCPUs | 16 GB Memory | 100 GB Common I/O), and Billing Mode (Pay-per-use). The bottom section contains advanced configuration options:

- Plan Type:** Radio buttons for Periodicity (selected) and One-time.
- Cluster Name:** A text input field with a placeholder 'Select a logical cluster...' and a note: 'The logical cluster is mainly used for computing acceleration.'
- Bind Main Logical Cluster:** A dropdown menu with '-Select-'.
- Bind User:** A dropdown menu with '-Select-' and a note: 'After a user is bound, the tables created by the user are still stored in the original logical cluster, and the computing logic of the user is switched to the current cluster for execution.'
- Nodes:** A numeric input field set to 3, with minus and plus buttons.
- Time Range:** A date range selector showing 'Start Date — End Date'.
- Period Type:** Radio buttons for Every Week (selected) and Every Month.
- Time Configuration(UTC):** A section with two rows of time settings. Each row includes 'Creation Completion Time' (with day, time, and a clock icon) and 'Deletion Start Time' (with day, time, and a clock icon). The first row shows Sunday 01:00 and Thursday 04:00. The second row shows Friday 04:00 and Saturday 03:00. Plus and minus buttons are on the right.

Step 3 [Table 1-3](#) lists the required data warehouse parameters.

Table 1-3 Data warehouse parameters

Parameter	Example Value	Description
Version	Coupled storage and compute	• Coupled storage and compute: Data is stored on local disks of compute nodes. • Decoupled Storage and Compute: Local DN disks are used only for data cache and metadata storage, and user data is stored on OBS.
Storage Type	Cloud SSD	• Cloud SSD: SSD EVS disks serve as data storage media, offering flexible storage capacity and supporting disk scale-out. • Local SSD: Local disks in ECS flavors serve as data storage media, offering fixed capacity and superior performance. However, disk scale-out is not supported. For more information, see What Are the Differences Between Cloud SSDs and Local SSDs?
Node Flavor	dwsx2.xlarge NOTE If this flavor is sold out, select other AZs or flavors.	Node flavors measure the compute power of DWS, including CPU architecture, vCPUs, memory, compute type, and deployment type. You need to select node flavors as needed. Typically, higher query performance requires larger node flavors. For more information, see Data Warehouse Flavors.
Hot Storage	20 GB per node	Available storage capacity of each node. The page displays the actual storage capacity for the service, such as 20 GB per node with a total of 3 nodes resulting in a total service data capacity of 60 GB. Customers with a large amount of service data should plan their storage capacity accordingly.
Nodes	3	Number of cluster nodes. The number of nodes ranges from 3 to 256.

Figure 1-2 Data warehouse configuration

Data Warehouse Configuration

Version

Coupled storage and compute

Decoupled storage and compute

Storage Type

Cloud SSD

Local SSD

Node Flavor

Flavor Name	vCPUs Memory	Hot Storage
☒ dwsx2.xlarge Trial only. Not for production	4 vCPUs 32 GB	20-2,000 GB SSD
☐ dwsx2.h.2xlarge.4.c7n	8 vCPUs 32 GB	100-4,000 GB SSD
☐ dwsx2.2xlarge	8 vCPUs 64 GB	100-4,000 GB SSD
☐ dwsx2.8xlarge	32 vCPUs 256 GB	100-16,000 GB SSD

Hot Storage

20

GB per node

Nodes

3

You can still create 1,024 nodes. You can [apply for a higher quota](#) as needed.

Total

dwsx2.xlarge | 12 vCPUs | 96 GB Memory | 60.00 GB hot storage

Step 4 Configure the parameters according to Table 1-4.

Table 1-4 Network configuration parameters

Parameter	Example Value	Description
VPC	vpc-default	VPC where DWS is located.
Subnet	subnet-default(192.168.0.0/24)	VPC subnet.
Security Group	Automatic creation	You can select Automatic creation or choose a created security group. The default option is Automatic creation, which means the system will create a security group automatically if you do not change the setting. The rule of the default security group is as follows: The outbound allows all access requests, while the inbound is open only to the database port 8000 that you set to connect to the DWS cluster.
EIP	Deselect it	If the client connected to DWS and the DWS cluster are not in the same VPC, use an EIP.

Parameter	Example Value	Description
ELB	Deselect it	If the private IP address or EIP of a CN is used to connect to a cluster, the failure of this CN will lead to cluster connection failure. ELB health checks enable quick forwarding of CN requests to healthy nodes in a cluster, minimizing access faults in case of a faulty CN. This ensures that workload is immediately shifted to a healthy node. For more information, see Associating and Disassociating ELB .

Figure 1-3 Network configuration

Network

VPC

vpc-default

View VPC

Subnet

subnet-default(192.168.0.0/24)

Security Group

Automatic creation

Automatically enables the port's access permission in the inbound rule.

Others

EIP

Do not use

Buy now

Specify

A cluster without an EIP cannot be accessed over the Internet, but can be accessed through an ECS deployed on a private network.

ELB

Do not use

Use existing

ELB is a service that automatically distributes incoming traffic across multiple backend cloud servers based on a specified forwarding policy. Clusters that do not use ELB cannot distribute traffic and are vulnerable to single points of failure (SPOFs). [Learn more](#)

Step 5 Configure the cluster by referring to [Table 1-5](#).

Table 1-5 Cluster management configuration

Parameter	Example Value	Description
Cluster Name	dws-demo	Cluster name, which is user-defined and can be modified later.
Cluster Version	Use the recommended version	Each version supports different database functions. By default, the recommended version is selected. To use the functions of a later version, select a later version. NOTE On the whitelist customer page, other cluster versions are displayed. Select a cluster version based on the site requirements. For details about the functions supported by each cluster version, see Product Notice.
Time Zone	Use default time zone	Default time zone of the cluster.
Enterprise Project	default	This parameter is displayed only when an enterprise account is used to create a DWS cluster. It enables unified management of cloud resources by project.
Tags	-	On DWS, after creating a cluster, you can add identifiers to items such as the project name, service type, and background information using tags. If you use tags in other cloud services, you are advised to create the same tag key-value pairs for cloud resources used by the same workload to keep consistency.

Figure 1-4 Advanced settings

Management

Cluster Name

dwz-demo

Enter 4 to 64 characters starting with a letter or Chinese character, and cannot exceed 64 bytes, one Chinese character is 3 bytes, and other characters are 1 byte. Only Chinese character letters digits hyphens (-) and underscores (_) are allowed.

Cluster Version

Recommended

8.3.0.108

[Release Notes](#) [Version Policies](#)

Default Database

gaussdb

Administrator Account

dbadmin

Enter 6 to 64 characters starting with a lowercase letter or an underscore (_). Only lowercase letters digits and underscores (_) are allowed.

Administrator Password

Confirm Password

Database Port

8000

Port for clients to connect to DWS. The value ranges from 8000 to 30000.

Time Zone

UTC+08:00 (Beijing, Chongqing, Hong ...

Advanced

Enterprise Project

default

[View enterprise projects.](#)

Advanced Settings

Default

Custom

Step 6 Configure the database by referring to Table 1-6.

Table 1-6 Database management, database encryption, and more configurations

Parameter	Example Value	Description
Administrator Account	dbadmin	Username of the database system administrator. Retain the default value. The database system administrator is assigned the <code>SYSADMIN</code> attribute by default. If the separation of permissions feature is not enabled, the administrator has the highest database permissions. This allows them to plan database objects, create databases and users, and authorize user roles. For more information, see Database Users .
Administrator Password	-	Password of the database system administrator.
Confirm Password	-	Enter the password again for confirmation.

Issue 13 (2025-10-10)

Copyright © Huawei Cloud Computing Technologies Co., Ltd.

8

Parameter	Example Value	Description
Database Port	8000	Port for the client to connect to the DWS database. Retain the default value 8000 .
Encrypt DataStore	-	If this function is enabled, Key Management Service (KMS) encrypts the cluster and the cluster's snapshot data.
More Configurations	-	• CNs: In a large-scale cluster, you are advised to deploy multiple CNs. • IPv6 CIDR Block: Specify whether to enable the IPv6 dual stack for the cluster. If IPv6 is selected, a client or application can connect to the database using an IPv6 address.

Step 7 In the **Configuration Summary** area on the right of the page, confirm the configurations and select **I have confirmed the current configuration**.

Step 8 Confirm the information and click **Buy Now**.

Cluster creation takes about 10 minutes. Clusters in the **Available** state are ready for use.

----End

Step 2: Uploading Data to OBS

Step 1 Create an OBS bucket. For details, see [OBS User Guide](#).

Set key parameters as follows and retain the default values for other parameters.



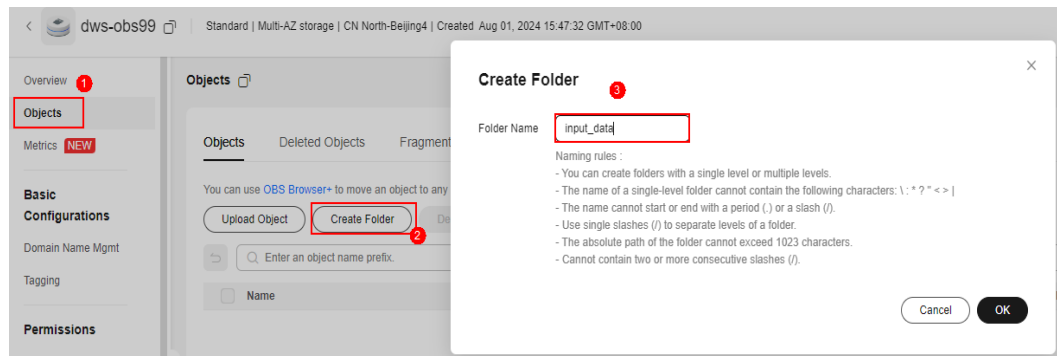
CAUTION

Cross-region access to OBS bucket data is not supported. The OBS bucket must be created in the same region as the DWS cluster. Select CN-Hong Kong.

- **Region**: Select the **CN-Hong Kong** region.
- **Bucket Name**: **dws-obs01** (If a conflict occurs, modify the number in the name to 02, 03, or a higher number)
- **Bucket Policy**: **Private**.

Step 2 After the OBS bucket is created, click the name of the newly created **dws-obs01** bucket on the bucket list. In the navigation pane, choose **Objects** and click **Create Folder**. Set the folder name to **input_data** and click **OK**.

Figure 1-5 Creating a folder



Step 3 Create three CSV data files on the local PC. The file names (including the file name extension) are **product_info0.csv**, **product_info1.csv**, and **product_info2.csv**.

Figure 1-6 Creating a test.csv file

	product_info0.csv	2024/8/1 16:05	Microsoft Excel ...	0 KB
	product_info1.csv	2024/8/1 16:05	Microsoft Excel ...	0 KB
	product_info2.csv	2024/8/1 16:05	Microsoft Excel ...	0 KB

Step 4 Use a text editor to open the three CSV files and copy the following content to each CSV file.

- **product_info0.csv:**
100,XHDK-A,2017-09-01,A,2017 Shirt Women,red,M,328,2017-09-04,715,good!
205,KDKE-B,2017-09-01,A,2017 T-shirt Women,pink,L,584,2017-09-05,40,very good!
300,JODL-X,2017-09-01,A,2017 T-shirt men,red,XL,15,2017-09-03,502,Bad.
310,QQPX-R,2017-09-02,B,2017 jacket women,red,L,411,2017-09-05,436,It's nice.
150,ABEF-C,2017-09-03,B,2017 Jeans Women,blue,M,123,2017-09-06,120,good.
- **product_info1.csv:**
200,BCQP-E,2017-09-04,B,2017 casual pants men,black,L,997,2017-09-10,301,good quality.
250,EABE-D,2017-09-10,A,2017 dress women,black,S,841,2017-09-15,299,This dress fits well.
108,CDXK-F,2017-09-11,A,2017 dress women,red,M,85,2017-09-14,22,It's really amazing to buy.
450,MMCE-H,2017-09-11,A,2017 jacket women,white,M,114,2017-09-14,22,very good.
260,OCDA-G,2017-09-12,B,2017 woolen coat women,red,L,2004,2017-09-15,826,Very comfortable.
- **product_info2.csv:**
980,"ZKDS-J",2017-09-13,"B","2017 Women's Cotton Clothing","red","M",112,,
98,"FKQB-I",2017-09-15,"B","2017 new shoes men","red","M",4345,2017-09-18,5473
50,"DMQY-K",2017-09-21,"A","2017 pants men","red","37",28,2017-09-25,58,"good","good","good"
80,"GKLW-L",2017-09-22,"A","2017 Jeans Men","red","39",58,2017-09-25,72,"Very comfortable."
30,"HWEC-L",2017-09-23,"A","2017 shoes women","red","M",403,2017-09-26,607,"good!"
40,"IQPD-M",2017-09-24,"B","2017 new pants Women","red","M",35,2017-09-27,52,"very good."
50,"LPEC-N",2017-09-25,"B","2017 dress Women","red","M",29,2017-09-28,47,"not good at all."
60,"NQAB-O",2017-09-26,"B","2017 jacket women","red","S",69,2017-09-29,70,"It's beautiful."
70,"HWNB-P",2017-09-27,"B","2017 jacket women","red","L",30,2017-09-30,55,"I like it so much"
80,"JKHU-Q",2017-09-29,"C","2017 T-shirt","red","M",90,2017-10-02,82,"very good."

Step 5 Use a text editor to convert the encoding format to UTF-8 and save the file.

Step 6 Return to the OBS console and click the name of the newly created **dws-obs01** bucket in the bucket list.

Step 7 In the navigation pane, choose **Objects** and click the **input_data** folder.

Step 8 Click **Upload Object**, and then click **Add File**.

Step 9 Upload the **product_info0.csv**, **product_info1.csv**, and **product_info2.csv** files to the OBS bucket.

----End

Step 3: Connecting to the DWS Cluster and Importing Data for Analysis

Step 1 Go to the DWS console.

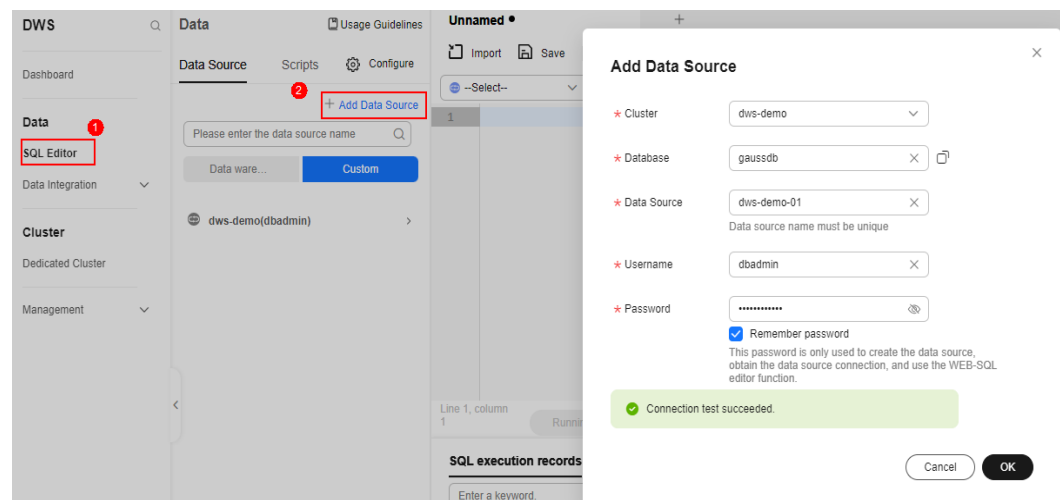
Step 2 In the navigation tree on the left, choose **Data > SQL Editor** and click **Add Data Source**.

Step 3 Enter the database connection information.

- **Cluster:** Select the created **dws-demo** cluster.
- **Database:** Retain the default value **gaussdb**.
- **Data Source:** Enter **dws-demo-01**.
- **Username:** Enter **dbadmin**.
- **Password:** Enter the password set during DWS creation in [Step 1: Creating a DWS Cluster](#).

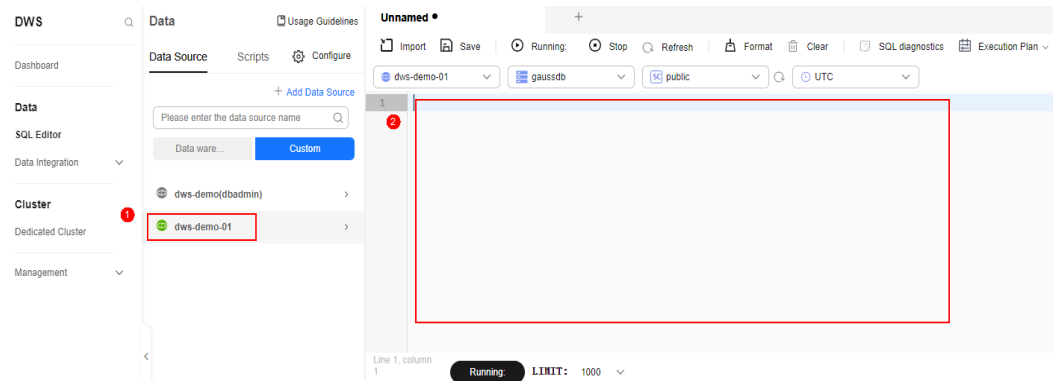
Step 4 Select **Remember password** and click **Test Connection**.

Figure 1-7 Database connection information

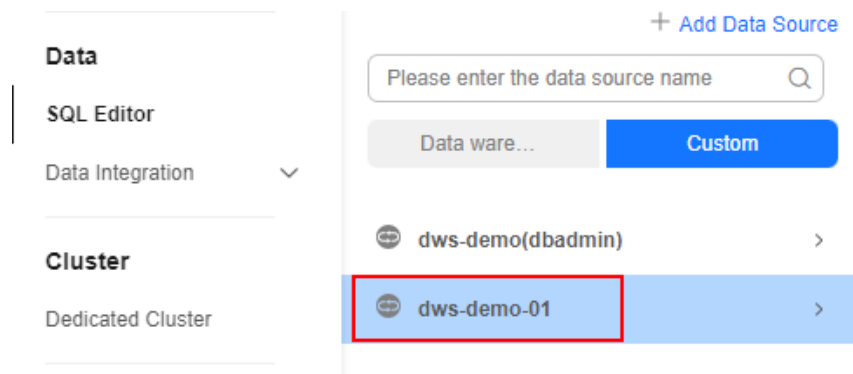


Step 5 Click **OK**.

The green icon  before the data source name in the following figure indicates that the database is connected. The right pane is the SQL editor window, where you can write and execute SQL statements.

Figure 1-8 Successful database connection**NOTE**

If the connection session times out or the browser is refreshed,  will appear grayed out, indicating that the connection has been disconnected. To reconnect to the database, double-click the data source name.

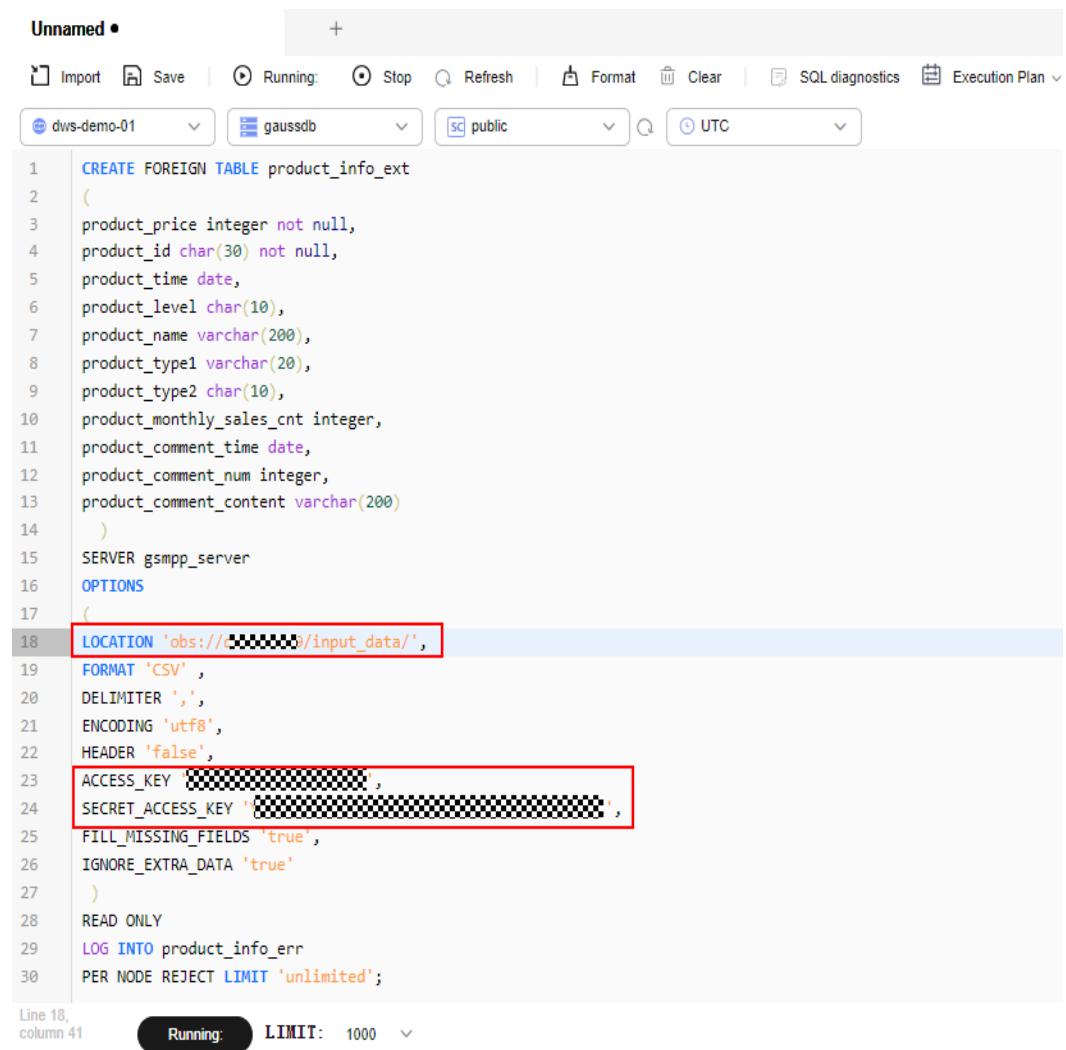


Step 6 Copy the following SQL statement to the SQL editor window and replace the OBS bucket name, AK value, and SK value with the actual values.

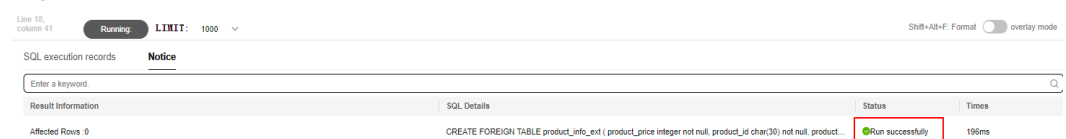
In this document, the OBS bucket name is **dws-obs01**. Obtain the AK and SK from [Access Keys](#).

```
CREATE FOREIGN TABLE product_info_ext
(
  product_price integer not null,
  product_id char(30) not null,
  product_time date,
  product_level char(10),
  product_name varchar(200),
  product_type1 varchar(20),
  product_type2 char(10),
  product_monthly_sales_cnt integer,
  product_comment_time date,
  product_comment_num integer,
  product_comment_content varchar(200)
)
SERVER gsmpp_server
OPTIONS
(
  LOCATION 'obs://OBS bucket name/input_data/',
  FORMAT 'CSV',
  DELIMITER ',',
  ENCODING 'utf8',
```

```
HEADER 'false',  
ACCESS_KEY 'AK value',  
SECRET_ACCESS_KEY 'SK value',  
FILL_MISSING_FIELDS 'true',  
IGNORE_EXTRA_DATA 'true'  
)  
READ ONLY  
LOG INTO product_info_err  
PER NODE REJECT LIMIT 'unlimited';
```

Figure 1-9 Creating an OBS foreign table

Step 7 Verify that the SQL statement is correct and click **Running**. The OBS foreign table is created.

Figure 1-10 Successful creation

Step 8 Copy the following SQL statement and run it in the SQL window to create an ordinary table.

```
CREATE TABLE product_info
(
product_price integer not null,
product_id char(30) not null,
product_time date ,
product_level char(10) ,
product_name varchar(200) ,
product_type1 varchar(20) ,
product_type2 char(10) ,
product_monthly_sales_cnt integer ,
product_comment_time date ,
product_comment_num integer ,
product_comment_content varchar(200)
)
WITH
(
orientation = column,
compression=middle
)
DISTRIBUTE BY hash (product_id);
```

Step 9 Copy the following SQL statement and execute it in the SQL window. Run the **INSERT** statement to import the OBS foreign table data to the DWS database.

```
INSERT INTO product_info SELECT * FROM product_info_ext;
```

Step 10 Copy the following SQL statements and run them in the SQL window to perform the **VACUUM** and **ANALYZE** operations.

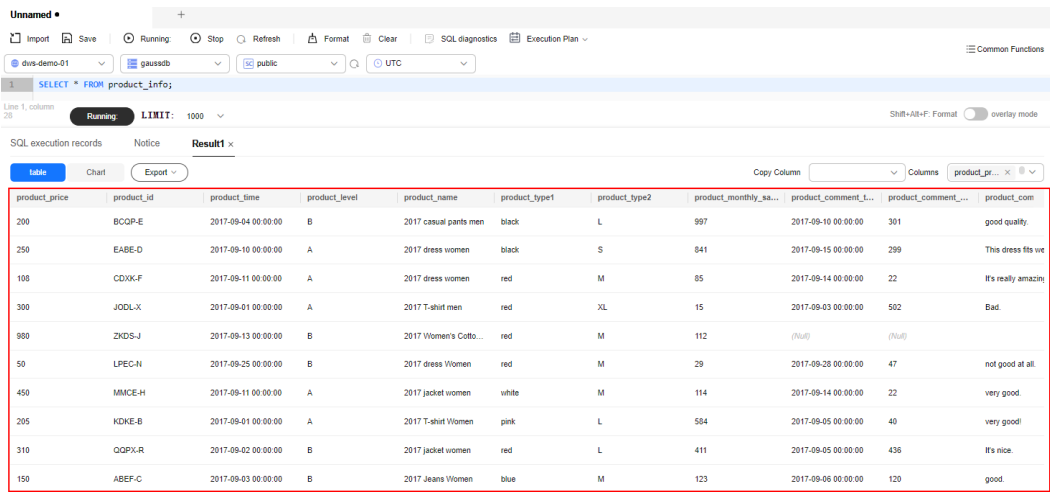
For details about the syntax, see the related SQL syntax section.

```
VACUUM FULL product_info;
ANALYZE product_info;
```

Step 11 Copy the following SQL statement and run it in the SQL window. The data details are successfully queried.

```
SELECT * FROM product_info;
```

Figure 1-11 Successful query



product_price	product_id	product_time	product_level	product_name	product_type1	product_type2	product_monthly_sales_cnt	product_comment_time	product_comment_num	product_comment_content
200	BCOP-E	2017-09-04 00:00:00	B	2017 casual pants men	black	L	997	2017-09-10 00:00:00	301	good quality.
250	EABE-D	2017-09-10 00:00:00	A	2017 dress women	black	S	841	2017-09-15 00:00:00	299	This dress fits me
108	CDXK-F	2017-09-11 00:00:00	A	2017 dress women	red	M	85	2017-09-14 00:00:00	22	It's really amazing
300	JODL-X	2017-09-01 00:00:00	A	2017 T-shirt men	red	XL	15	2017-09-03 00:00:00	502	Bad.
980	ZKDS-J	2017-09-13 00:00:00	B	2017 Women's Cotton	red	M	112	(null)	(null)	
50	LPEC-N	2017-09-25 00:00:00	B	2017 dress Women	red	M	29	2017-09-28 00:00:00	47	not good at all.
450	MMCE-H	2017-09-11 00:00:00	A	2017 Jacket women	white	M	114	2017-09-14 00:00:00	22	very good.
205	KDKE-B	2017-09-01 00:00:00	A	2017 T-shirt Women	pink	L	584	2017-09-05 00:00:00	40	very good!
310	QGPX-R	2017-09-02 00:00:00	B	2017 Jacket women	red	L	411	2017-09-05 00:00:00	436	It's nice.
150	ABEF-C	2017-09-03 00:00:00	B	2017 Jeans Women	blue	M	123	2017-09-06 00:00:00	120	good.

----End

2 Getting Started with DWS Data Development SQL

This section describes some basic SQL operations in DWS.

Creating, Viewing, and Deleting a Database

- Run the **CREATE DATABASE** statement to create a database.
`CREATE DATABASE test_db ENCODING 'UTF8' template = template0;`
- Query the database list using the **\l** meta-command.
`\l`
- Querying the database list using the **PG_DATABASE** system catalog
`SELECT datname FROM pg_database;`
- Run the **DROP DATABASE** statement to delete a database.
`DROP DATABASE test_db;`

Creating, Viewing, Modifying, and Deleting a Table

- Run the **CREATE TABLE** statement to create a table.
`CREATE TABLE customer_t1(id INT, name CHAR(40),age TINYINT);`
- Use the **PG_GET_TABLEDEF()** function to view the table creation statement. Replace **customer_t1** with the name of the table to be queried.
`SELECT * FROM PG_GET_TABLEDEF('customer_t1');`
- Run the **ALTER TABLE** statement to modify a table.
Add a column:
`ALTER TABLE customer_t1 ADD (address VARCHAR(100));`
Delete a column:
`ALTER TABLE customer_t1 DROP COLUMN address;`
Modify a column type:
`ALTER TABLE customer_t1 MODIFY age INTEGER NOT NULL;`
- Run the **DROP TABLE** statement to delete a table.
`DROP TABLE customer_t1;`

Creating, Viewing, and Deleting Indexes

- Run the **CREATE INDEX** or **ALTER TABLE** statement to create a common index.
`CREATE INDEX c_id_index on customer_t1(id);`
`ALTER TABLE customer_t1 ADD INDEX c_id_index (id);`

- Use the **PG_INDEXES** system catalog to view all indexes in a table.
`SELECT * FROM pg_indexes WHERE tablename = 'customer_t1';`
- Run the **ALTER TABLE** or **DROP INDEX** statement to delete an index.
`DROP INDEX c_id_index;`
`ALTER TABLE customer_t1 DROP INDEX c_id_index;`

Adding, Deleting, and Modifying Table Data

- Run the **INSERT INTO** statement to insert table data.
`INSERT INTO customer_t1 VALUES(1001,'user1',22);`
- Run the **SELECT** statement to query table data.
`SELECT * FROM customer_t1;`
- Run the **UPDATE** statement to update table data.
`UPDATE customer_t1 SET id = 1009 WHERE id = '1001';`
- Use the **DELETE** statement to delete table data.
`DELETE FROM customer_t1 WHERE id = '1009';`

3

Getting Started with DWS

After creating a DWS cluster, you can try some best practices provided by DWS to meet your workload requirements.

Table 3-1 Best practices

Practice		Description
Data Import and Export	Importing Data from OBS to a Cluster	This practice demonstrates how to upload sample data to OBS and import OBS data to the target table on DWS, helping you quickly learn how to import data from OBS to a DWS cluster. You can import data in TXT, CSV, ORC, PARQUET, CARBONDATA, or JSON format from OBS to a DWS cluster for query.
	Using GDS to Import Data from a Remote Server	This practice demonstrates how to use General Data Service (GDS) to import data from a remote server to DWS. DWS allows you to import data in TXT, CSV, or FIXED format.

Practice		Description
	Exporting ORC Data from a DWS Cluster to an MRS Cluster	DWS allows you to export ORC data to MRS using an HDFS foreign table. You can specify the export mode and export data format in the foreign table. Data is exported from DWS in parallel using multiple DNs and stored in HDFS. In this way, the overall export performance is improved.
Data Migration	Using CDM to Migrate Oracle Data to a DWS Cluster	This practice demonstrates how to migrate Oracle data to DWS.
	Using a Flink Job of DLI to Synchronize Kafka Data to a DWS Cluster in Real Time	This practice demonstrates how to use DLI Flink jobs to synchronize consumption data from Kafka to DWS in real time. This practice takes about 90 minutes. The cloud services used in this practice include Virtual Private Cloud (VPC) and subnets, Elastic Load Balance (ELB), Elastic Cloud Server (ECS), Object Storage Service (OBS), Distributed Message Service (DMS) for Kafka, Data Lake Insight (DLI), and DWS.

Practice		Description
Optimization Table	Table Optimization Practices	In this practice, you will learn how to optimize the design of your tables. You will start by creating tables without specifying their storage mode, distribution key, distribution mode, or compression mode. Load test data into these tables and test system performance. Then, follow excellent practices to create the tables again using new storage modes, distribution keys, distribution modes, and compression modes. Load the test data and test performance again. Compare the two test results to find out how table design affects the storage space, and the loading and query performance of the tables. Estimated time: 60 minutes
Advanced Features	Best Practices of Hot and Cold Data Management	In massive big data scenarios, with the growing of data, data storage and consumption increase rapidly. The need for data may vary in different time periods, therefore, data is managed in a hierarchical manner, improving data analysis performance and reducing service costs. In some data usage scenarios, data can be classified into hot data and cold data by accessing frequency.

Practice		Description
	Best Practices for Automatic Partition Management	For partition tables whose partition columns are time, the automatic partition management function can be added to automatically create partitions and delete expired partitions, reducing partition table maintenance costs and improving query performance. To facilitate data query and maintenance, the time column is often used as the partition column of a partitioned table that stores time-related data, such as e-commerce order information and real-time IoT data. When the time-related data is imported to a partitioned table, the table should have partitions of the corresponding time ranges. Common partition tables do not automatically create new partitions or delete expired partitions. Therefore, maintenance personnel need to periodically create new partitions and delete expired partitions, leading to increased O&M costs. Addressing this, DWS introduces the automatic partition management feature. You can set the table-level parameters period and ttd to enable the automatic partition management function, which automatically creates partitions and deletes expired partitions, reducing partitioned table maintenance costs and improving query performance.

Practice		Description
Database Management	Best Practices of Resource Management	This practice demonstrates how to use DWS for resource management, helping enterprises eliminate bottlenecks in concurrent queries. SQL jobs can run smoothly without affecting each other and consume less resources than before.
	Excellent Practices for SQL Queries	Based on a large number of SQL execution mechanisms and practices, we can optimize SQL statements following certain rules to more quickly execute SQL statements and obtain correct results.
	Excellent Practices for Data Skew Queries	This practice includes the following storage skew cases: • Real-Time Detection of Storage Skew During Data Import • Quickly Locating the Tables That Cause Data Skew
	Best Practices of User Management	A DWS cluster mainly consists of system administrators and common users. This practice describes the permissions of system administrators and common users and describes how to create users and query user information.
	Viewing Table and Database Information	This practice demonstrates some basic database query cases: • Querying Table Information • Querying the Table Size • Querying Database Information • Querying the Database Size

Practice		Description
Sample Data Analysis	Checkpoint Vehicle Analysis	This practice demonstrates how to analyze vehicles that have passed through traffic checkpoints. The process involves loading 890 million pieces of data from these checkpoints into a single database table on DWS for accurate and fuzzy query. It is a great example of how DWS can handle high-performance queries of historical data.
	Supply Chain Requirement Analysis (TPC-H Data Set)	This practice demonstrates how to load the sample data set from OBS to a DWS cluster and perform data queries. It comprises multi-table analysis and theme analysis in data analysis scenarios.
	Operations Status Analysis of a Retail Department Store	In this practice, the daily business data of each retail store is loaded from OBS to the corresponding table in the data warehouse cluster for summarizing and querying KPIs. This data includes store turnover, customer flow, monthly sales ranking, monthly customer flow conversion rate, monthly price-rent ratio, and sales per unit area. This practice demonstrates the multidimensional query and analysis of DWS in retail scenarios.

Practice		Description
Data Security	Encrypting and Decrypting Data Columns	Data encryption is widely used in information systems to prevent unauthorized access and data leakage. As the core of an information system, the DWS data warehouse also provides transparent encryption and encryption using SQL functions. This section describes SQL function encryption.