GaussDB

Getting Started

Issue	01
Date	2024-10-08

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 Buying a GaussDB Instance and Connecting to It Through the DAS Console (Recommended)

Data Admin Service (DAS) enables you to manage instances on a web-based console, simplifying database management and improving working efficiency. You can connect to and manage instances through DAS.

This section describes how to use DAS to connect to a GaussDB instance you have bought on the GaussDB management console. For more methods of connecting to an instance, see **Connecting to a GaussDB Instance**.

- **Step 1: Buy an Instance**
- **Step 2: Log In to a GaussDB Instance Through DAS**

Buying an Instance

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner and select a region and project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **GaussDB**.

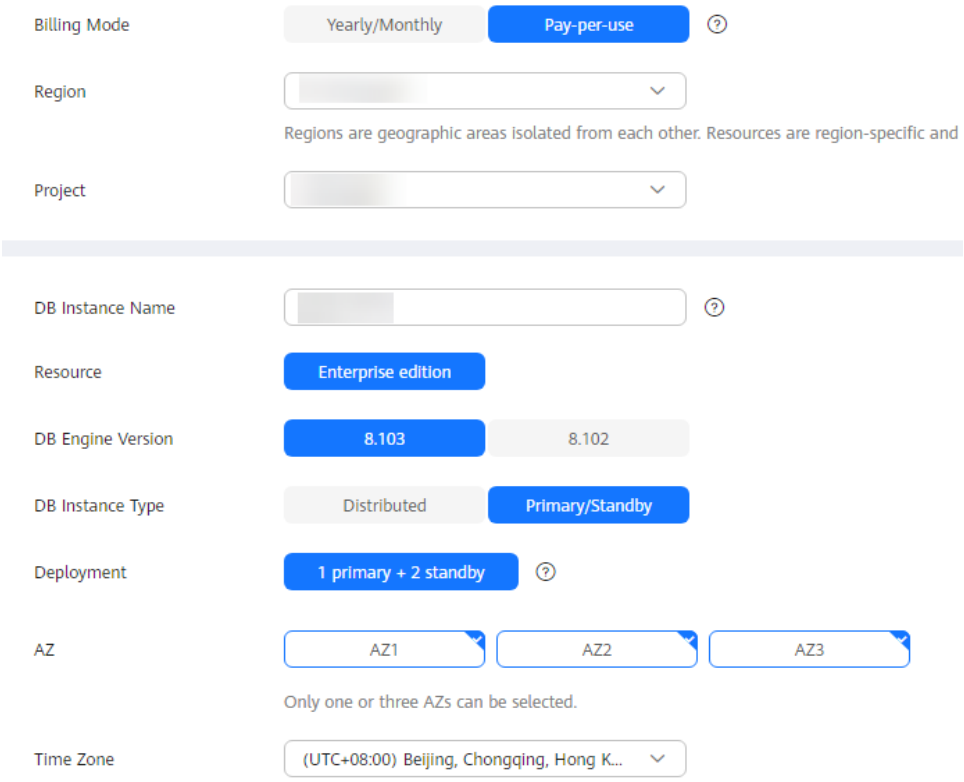
Step 4 On the **Instances** page, click **Buy DB Instance**.

Step 5 On the displayed page, set **Billing Mode**, enter **DB Instance Name**, select **Edition Type**, **DB Engine Version**, **DB Instance Type**, and **Deployment Model**, and complete other configurations.

- **Billing Mode**: **Pay-per-use**
- Region and Project: Retain the default settings.
- **DB Instance Name**: Enter a name that is easy to identify.
- **Edition Type**: **Enterprise edition**
- **DB Engine Version**: **8.103**
- **DB Instance Type**: **Primary/Standby**
- **Deployment Model**: **1 primary + 2 standby**

- **AZ**: **AZ1**
- **Time Zone**: Retain the default value.

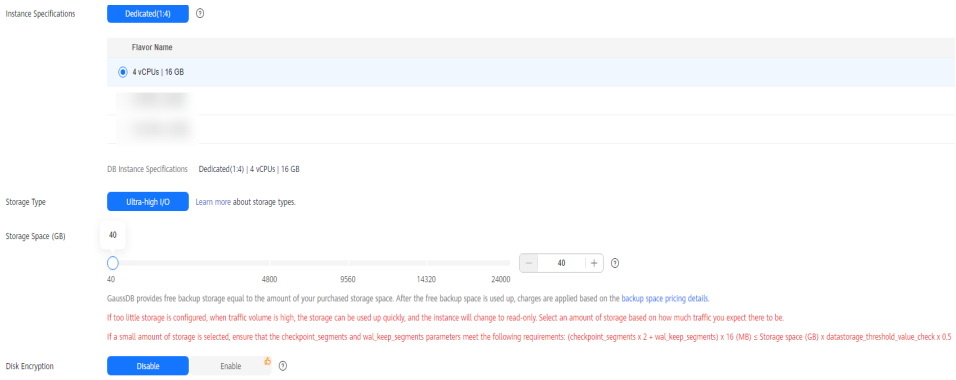
Figure 1-1 Billing mode and basic information



Step 6 Configure instance specifications.

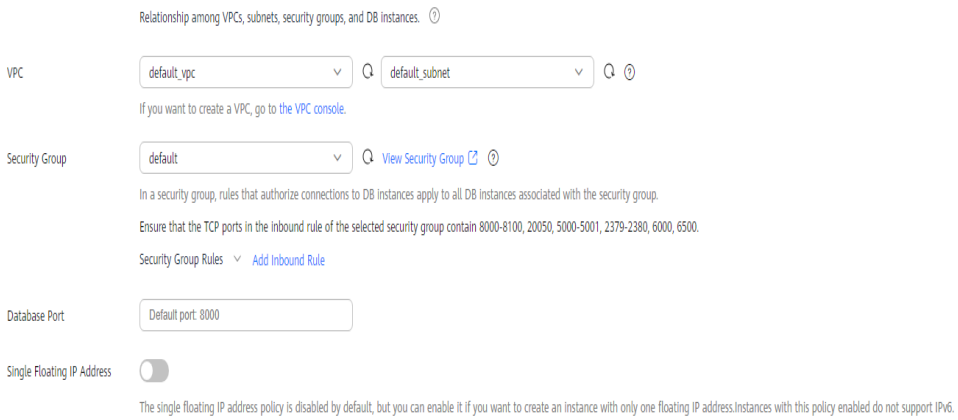
- **Instance Specifications**: **Dedicated(1:4)** and **4 vCPUs | 16 GB**
- **Storage Type**: **Ultra-high I/O**
- **Storage Space**: **40 GB**
- **Disk Encryption**: **Disable**

Figure 1-2 Specifications and storage



Step 7 Retain the default settings for the network information.

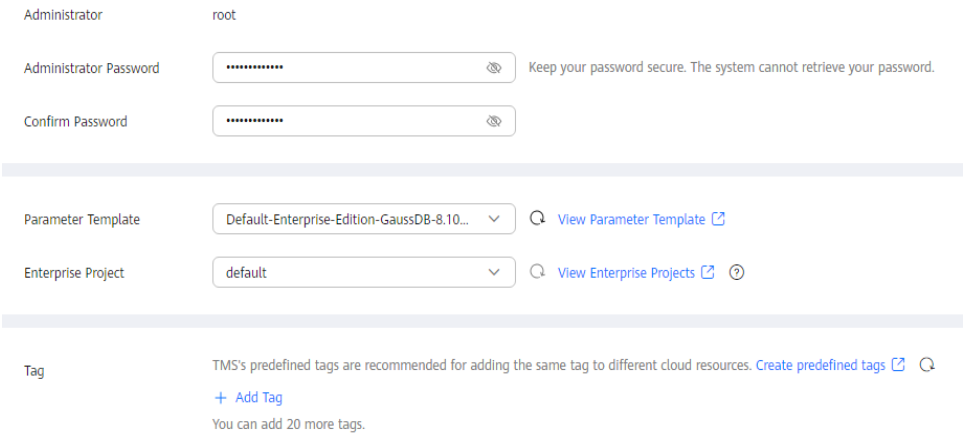
Figure 1-3 Network configuration



Step 8 Configure the instance password and enterprise project.

- **Administrator Password**: Set a strong password to improve security.
- **Confirm Password**: Enter the administrator password again.
- **Enterprise Project**: **default**
- Retain the default values for other parameters.

Figure 1-4 Database configuration



Step 9 Click **Next**, confirm the instance information, and click **Submit**.

Step 10 Go to the instance list.

If status of the instance becomes **Available**, the instance has been created.

----End

Logging In to a GaussDB Instance Through DAS

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **GaussDB**.

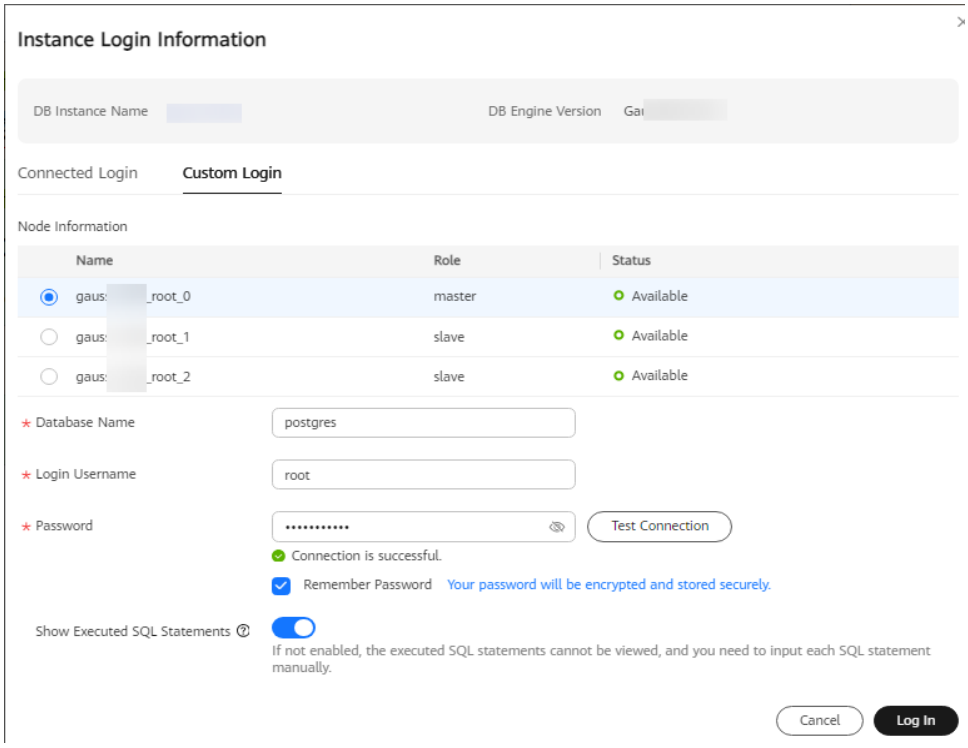
Step 4 On the **Instances** page, locate the DB instance you want to log in to and click **Log In** in the **Operation** column.

Alternatively, click the DB instance name on the **Instances** page. On the displayed **Basic Information** page, click **Log In** in the upper right corner of the page.

Step 5 On the **Custom Login** page, select the node to be logged in to. Enter the correct database username and password, and click **Test Connection**. After the connection test is successful, click **Log In**.

- **Login Username**: **root**
- **Database Name**: **postgres**
- **Password**: Enter the password of the **root** user. It is the administrator password you set when you buy an instance.
- **Show Executed SQL Statements**: Enable it.

Figure 1-5 Login page



----End

Getting Started with SQL

After logging in to an instance, you can create database users, databases, and tables through DAS.

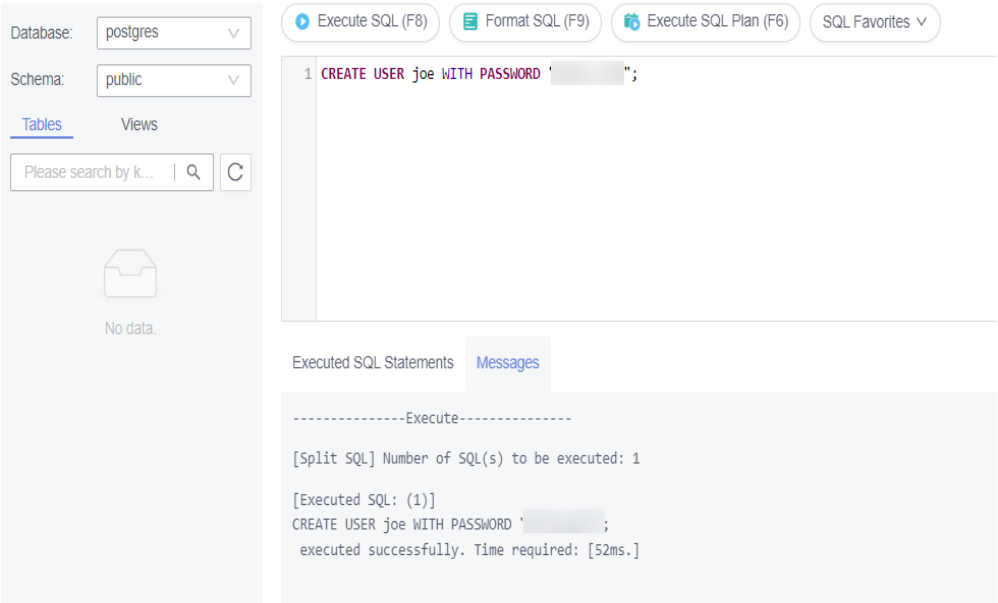
Step 1 Create a database user.

Only administrators that are created during the instance installation can access the initial database by default. You can also create other database users.

CREATE USER *joe* **WITH PASSWORD** "xxxxxxxxx";

If information similar to the following is displayed, the creation is successful.

Figure 1-6 Creating a database user



In this case, you have created a user named **joe**, and the user password is **xxxxxxx**.

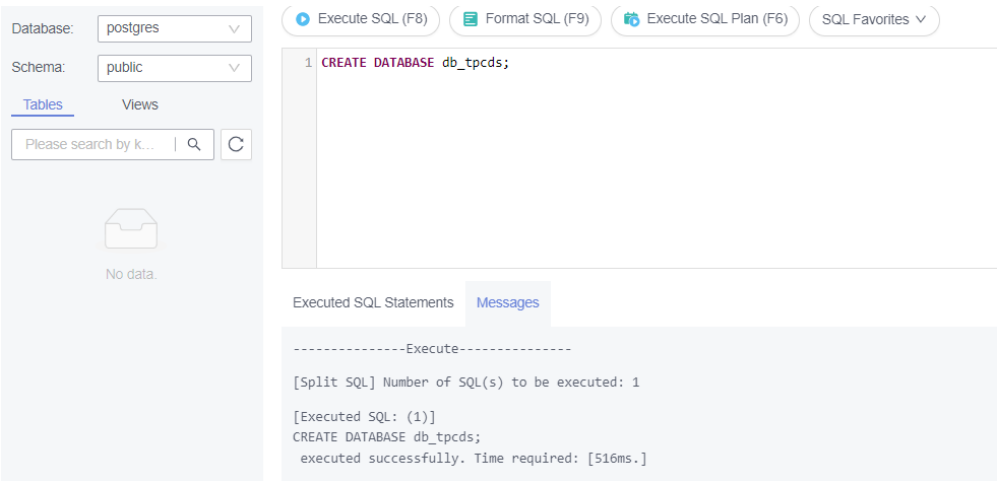
For more information about database users, see **Users and Permissions**.

Step 2 Create a database.

CREATE DATABASE *db_tpcds*;

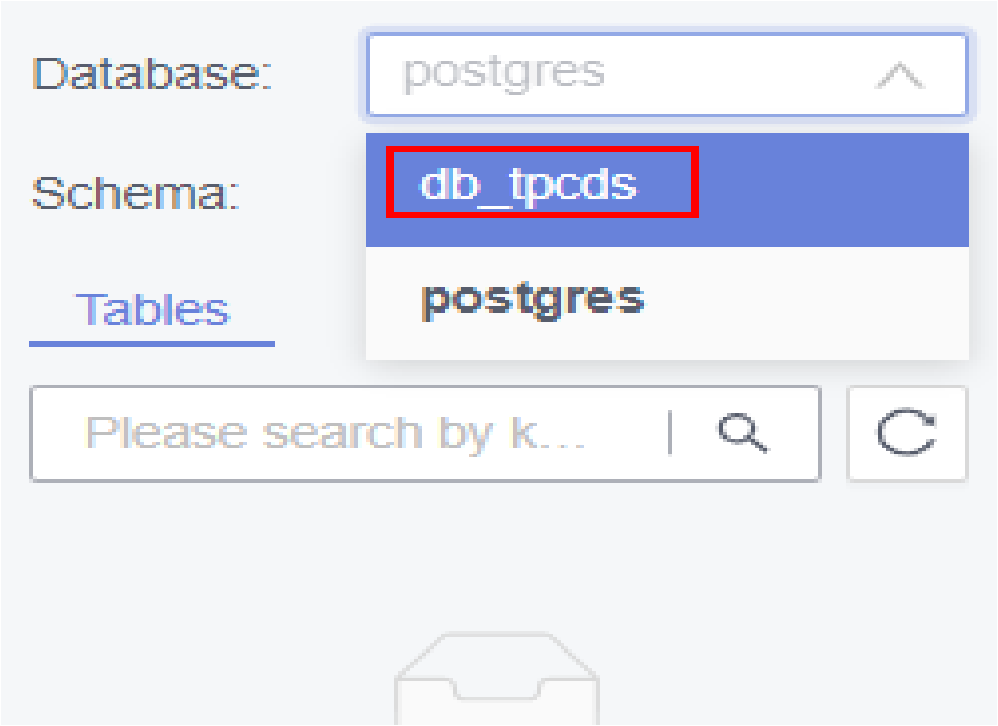
If information similar to the following is displayed, the creation is successful.

Figure 1-7 Creating a database



Switch to the newly created database in the upper left corner.

Figure 1-8 Switching to the new database



Step 3 Create a table.

- Run the following command to create a schema:

  CREATE SCHEMA *myschema*;
- Create a table named **mytable** that has only one column. The column name is **firstcol** and the column type is integer.

  CREATE TABLE myschema.mytable *(firstcol int)*;
- Insert data to the table.

  INSERT INTO myschema.mytable values (100);
- View data in the table.

  SELECT * FROM myschema.mytable;

Note:

- By default, new database objects, such as the **mytable** table, are created in the *$user* schema. For more information about schemas, see **Creating and Managing Schemas**.
- For details about how to create a table, see **Creating and Managing Tables**.
- In addition to the created tables, a database contains many system catalogs. These system catalogs contain information about instance installation as well as the queries and processes running in GaussDB. You can collect information about the database by querying the system catalogs. For details about querying system catalogs, see **Querying a System Catalog**.

Step 4 In the **db_tpcds** database, run the following statement as user **root** to grant all permissions of the **db_tpcds** database to user **joe**:

GRANT ALL ON DATABASE db_tpcds **TO** joe;

GRANT USAGE ON schema myschema **TO** joe;

GRANT ALL ON TABLE myschema.mytable **TO** joe;

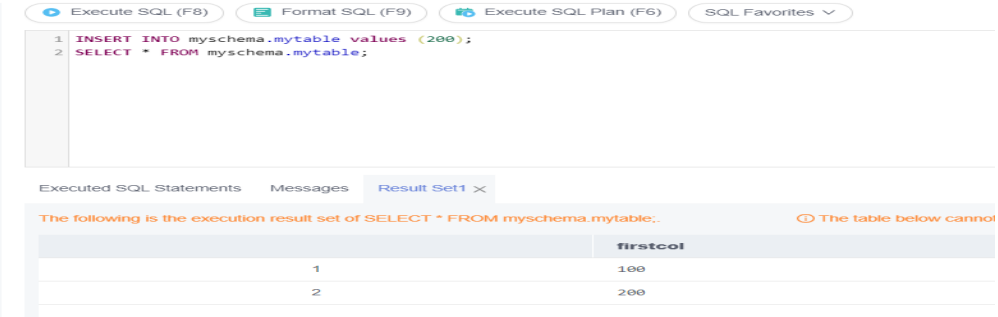
Step 5 Log in to the **db_tpcds** database as user **joe**.

Step 6 After login, insert data into the table and verify the data.

INSERT INTO myschema.mytable values (200);

SELECT * FROM myschema.mytable;

Figure 1-9 Inserting and verifying data



----End

Follow-up Operations

After logging in to the instance, you can create alarm rules, set automated backup policies, create databases, and migrate data. For details, see:

- **Creating an Alarm Rule**
- **Setting an Automated Backup Policy**
- **Creating a Database Using SQL**
- **Creating a Database Through an API**
- **Managing GaussDB Databases Through DAS**
- **Migrating Data to GaussDB**

2 Buying and Connecting to the GaussDB Instance Using gsql

This section describes how to use the gsql client to connect to a GaussDB instance you have bought on the GaussDB management console. For more methods of connecting to an instance, see **Connecting to a GaussDB Instance**.

- **Step 1: Buy an Instance**
- **Step 2: Buy an ECS**
- **Step 3: Obtain the Driver Package**
- **Step 4: Connect to the Database**
  - **Non-SSL connection**
  - **SSL connection**

Buying an Instance

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner and select a region and project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **GaussDB**.

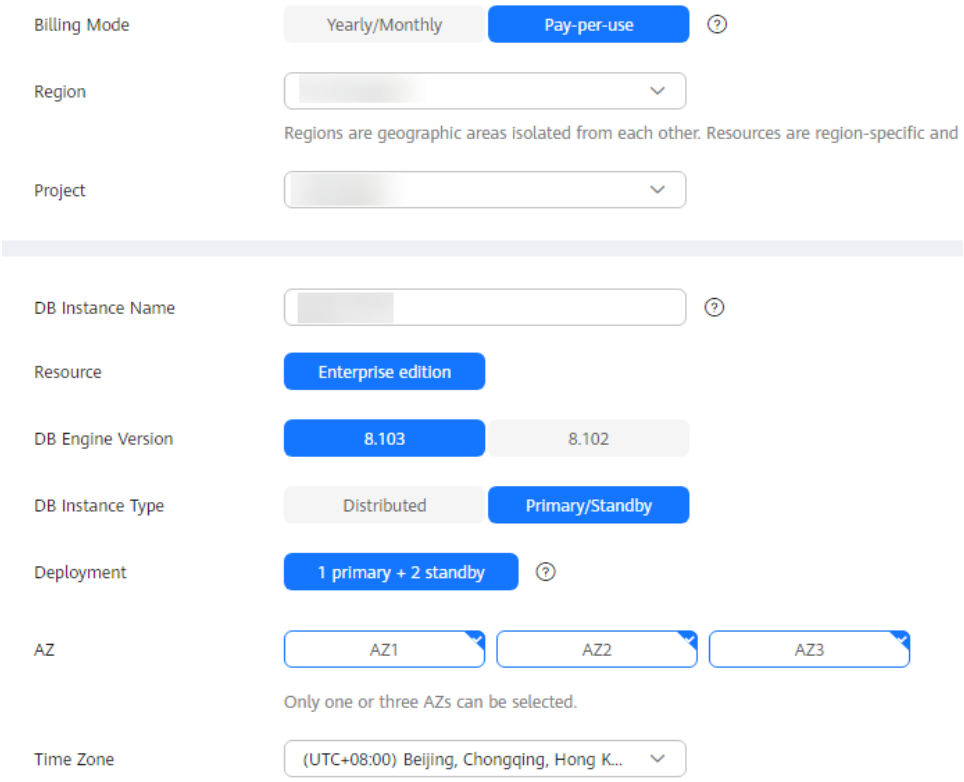
Step 4 On the **Instances** page, click **Buy DB Instance**.

Step 5 On the displayed page, set **Billing Mode**, enter **DB Instance Name**, select **Edition Type**, **DB Engine Version**, **DB Instance Type**, and **Deployment Model**, and complete other configurations.

- **Billing Mode**: **Pay-per-use**
- **Region** and **Project**: Retain the default settings.
- **DB Instance Name**: Enter a name that is easy to identify.
- **Edition Type**: **Enterprise edition**
- **DB Engine Version**: **8.103**
- **DB Instance Type**: **Primary/Standby**
- **Deployment Model**: **1 primary + 2 standby**

- **AZ**: **AZ1**
- **Time Zone**: Retain the default value.

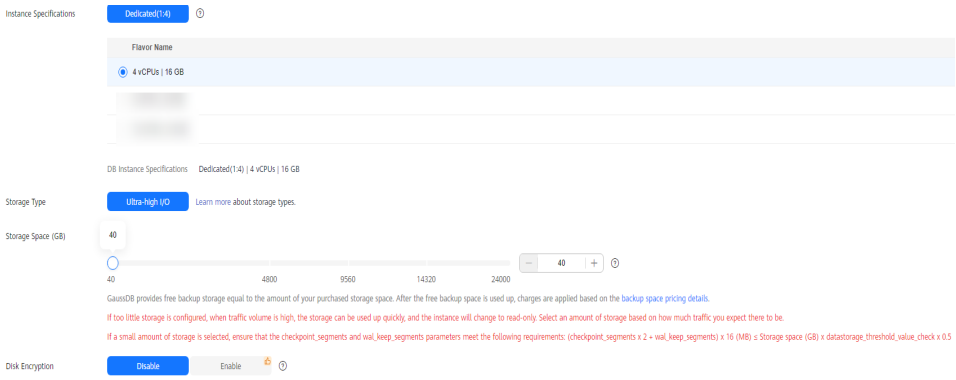
Figure 2-1 Billing mode and basic information



Step 6 Configure instance specifications.

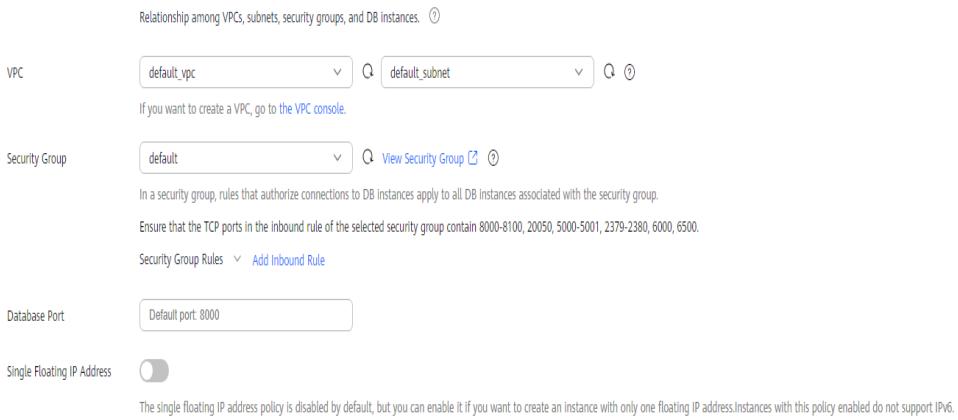
- **Instance Specifications**: **Dedicated(1:4)** and **4 vCPUs | 16 GB**
- **Storage Type**: **Ultra-high I/O**
- **Storage Space**: **40 GB**
- **Disk Encryption**: **Disable**

Figure 2-2 Specifications and storage



Step 7 Retain the default settings for the network information.

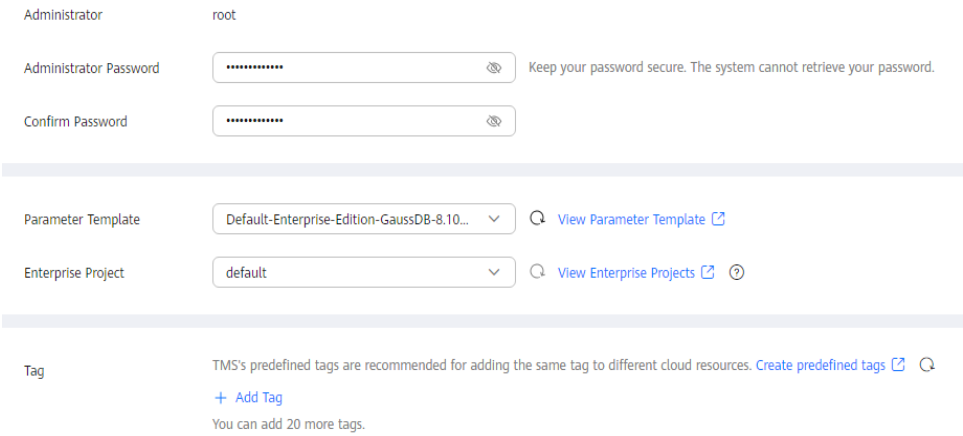
Figure 2-3 Network configuration



Step 8 Configure the instance password and enterprise project.

- **Administrator Password**: Set a strong password to improve security.
- **Confirm Password**: Enter the administrator password again.
- **Enterprise Project**: **default**
- Retain the default values for other parameters.

Figure 2-4 Database configuration



Step 9 Click **Next**, confirm the instance information, and click **Submit**.

Step 10 Go to the instance list.

If status of the instance becomes **Available**, the instance has been created.

----End

Buying an ECS

GaussDB provides the gsql tool to help you connect to the database using the command-line interface (CLI). You need to create an ECS in advance for installing the gsql tool.

1. **Log in to the management console**.
2. Click **Buy ECS**.
3. Configure basic settings and click **Next: Configure Network**.
   - **Region**: Select the region of the GaussDB instance to be connected.
   - **Billing Mode**: **Pay-per-use**
   - **AZ**: Select the AZ of the GaussDB instance to be connected.
   - **Image**: **EulerOS** and **EulerOS 2.5 64bit(40 GiB)**
   - Retain the default values for other parameters.

Figure 2-5 Basic settings

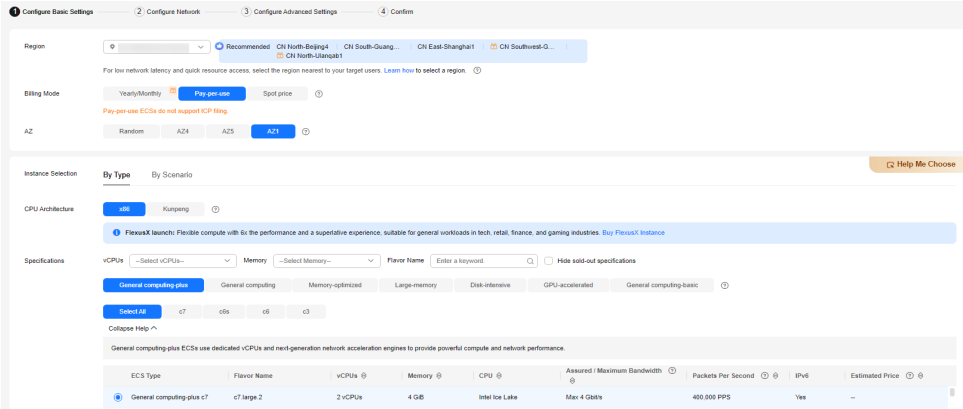
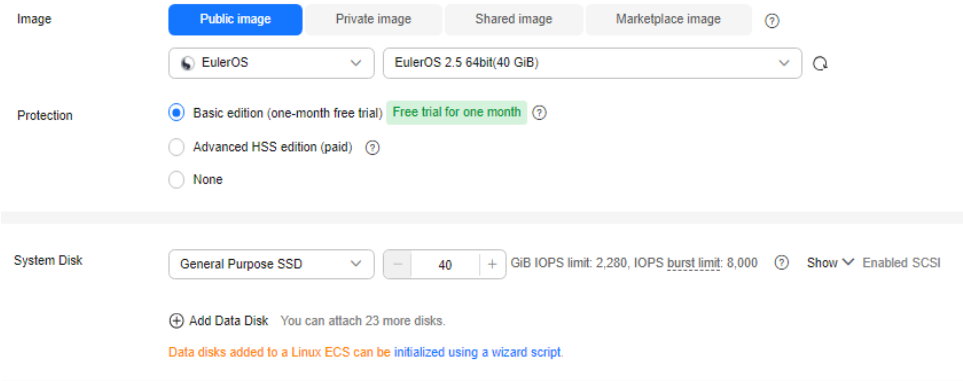
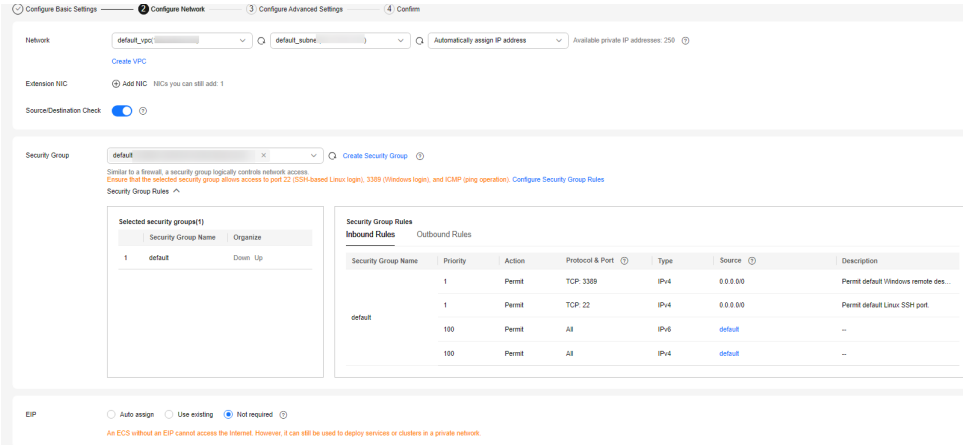


Figure 2-6 Selecting an image



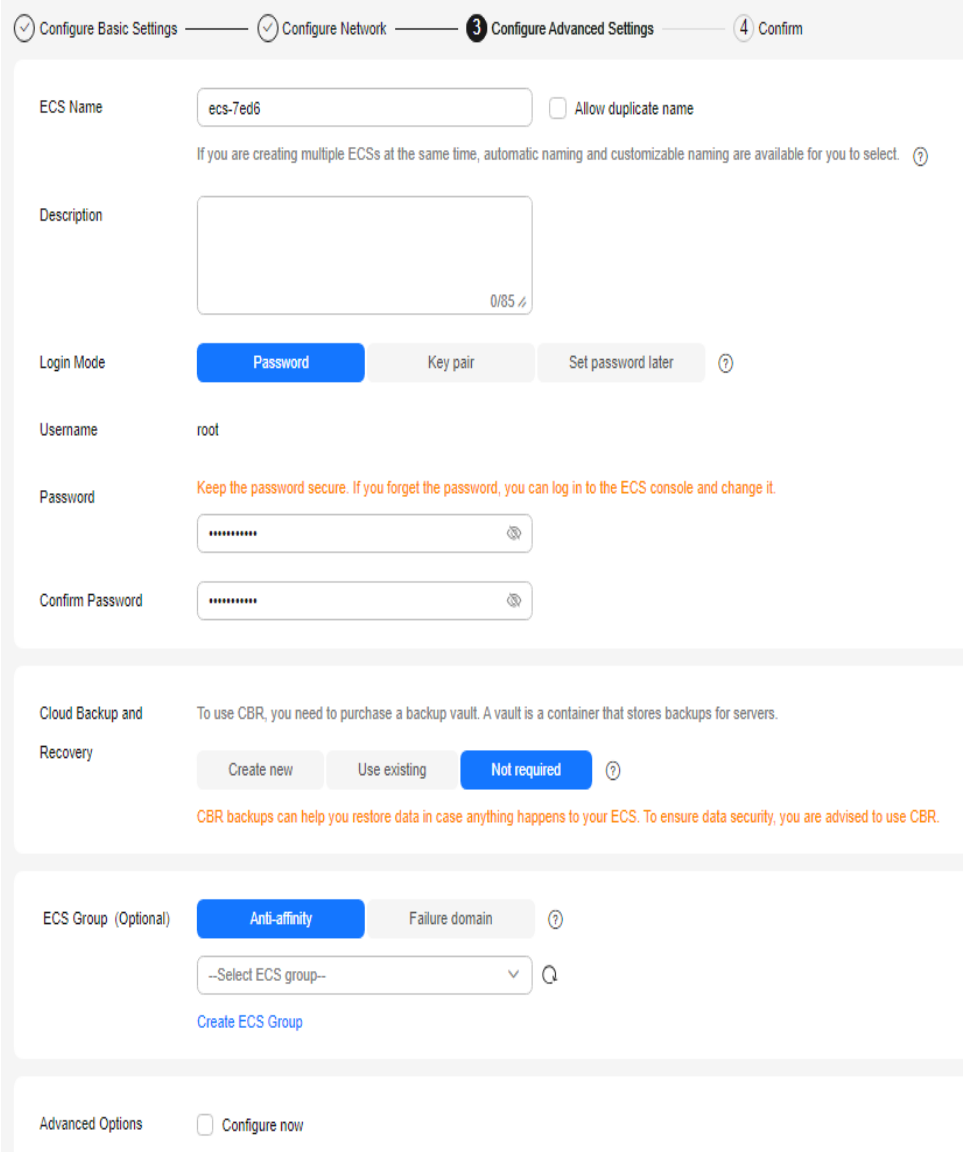
4. Configure the ECS network and click **Next: Configure Advanced Settings**.
   - **Network**: Select the VPC of the GaussDB instance to be connected.
   - **Security Group**: Select the security group of the GaussDB instance to be connected.
   - **EIP**: **Do Not Use**

Figure 2-7 Network configuration



5. Configure a password for the ECS, set other parameters as required, and click **Next: Confirm**.
   - **ECS Name**: Enter a name that is easy to identify.
   - **Password**: Set a strong password to improve security.
   - **Confirm Password**: Enter the password again.
   - Retain the default values for other parameters.

Figure 2-8 Advanced settings



6. Confirm the configuration and click **Apply Now**.
7. View the purchased ECS.

Obtaining the Driver Package

Download particular packages listed in **Table 2-1** based on the version of your instance.

Table 2-1 Driver package download list

Version	Download Address
8.x	**Driver package** **Verification package for the driver package**

Version	Download Address
3.*x*	**Driver package** **Verification package for the driver package**
2.*x*	**Driver package** **Verification package for the driver package**

To prevent a software package from being tampered with during transmission or storage, download the corresponding verification package and perform the following steps to verify the software package:

1. Upload the software package and verification package to the same directory on a Linux VM.
2. Run the following command to verify the integrity of the software package:

   cat *GaussDB_driver.zip.sha256* **| sha256sum --check**

   If **OK** is displayed in the command output, the verification is successful.

   ```
   GaussDB_driver.zip: OK
   ```

Connecting to a Database

- **Non-SSL connection**
  a. Log in as user **root** to the ECS you have created.
  b. Upload the client tool package and configure gsql environment variables.
     i. Run the following command to create the **/tmp/tools** directory for storing the client tool package:
     ```
     mkdir /tmp/tools
     ```
     ii. Download the **GaussDB_driver.zip** driver package of the required version by referring to **Obtaining the Driver Package**, and upload it to the **/tmp/tools** directory of the created ECS.
     iii. Run the following commands to decompress the **GaussDB_driver.zip** driver package:
     ```
     cd /tmp/tools
     unzip GaussDB_driver.zip
     ```
     iv. Run the following commands to copy the decompressed **GaussDB-Kernel_***_EULER_64bit-Gsql.tar.gz** client tool package to the **/tmp/tools** directory:

     NOTE

     This section uses the gsql tool package suitable for the primary/standby instances running on Euler2.5_x86_64 as an example. The relative path of the tool package varies depending on where you decompressed it.

     ```
     cd /tmp/tools/GaussDB_driver/Centralized/Euler2.5_X86_64/
     cp GaussDB-Kernel_***_EULER_64bit-Gsql.tar.gz /tmp/tools
     ```
     v. Run the following commands to decompress the package:
     ```
     cd /tmp/tools
     tar -zxvf GaussDB-Kernel_***_EULER_64bit-Gsql.tar.gz
     ```


vi. Configure environment variables.

Run the following command to open the **~/.bashrc** file:

```
vim ~/.bashrc
```

Press **G** to move the cursor to the last line, press **i** to enter Insert mode, and type the following information. Then, press **Esc** to exit Insert mode, and run **:wq** to save the settings and exit.

```
export PATH=/tmp/tools/bin:$PATH
export LD_LIBRARY_PATH=/tmp/tools/lib:$LD_LIBRARY_PATH
```

Run the following command to make the environment variables take effect permanently:

```
source ~/.bashrc
```

c. Enter the password when prompted to connect to the database.

After an instance is created, a **postgres** database is generated by default. Database **postgres** is used an example.

```
gsql -d postgres -h 10.0.0.0 -U root -p 8000
Password for user root:
```

- *postgres* indicates the name of the database to be connected.
- *10.0.0.0* indicates the IP address of the instance. On the **Basic Information** page of the instance to be connected, obtain the IP address in the **Node List** area.
- *root* indicates the username used to log in to the database.
- *8000* indicates the port number of the database. On the **Basic Information** page of the instance to be connected, obtain the database port in the **Network Information** area.

For more information about gsql commands, see ***Tool Reference***.

- **SSL connection**

a. **Log in to the management console**.

b. Click in the upper left corner and select a region and project.

c. Click in the upper left corner of the page and choose **Databases** > **GaussDB**.

d. On the **Instances** page, click the name of the target instance. In the **Configuration** area on the **Basic Information** page, click next to the **SSL** field to download the root certificate or certificate bundle.

e. Upload the root certificate to the ECS or save it to the device to be connected to the GaussDB instance.

Import the root certificate to the Linux ECS. For details, see **How Can I Import the Root Certificate to a Windows or Linux OS?**

f. Connect to a GaussDB instance.

A Linux ECS is used in this example. Run the following command to set environment variables on the ECS:

```
export PGSSLMODE=<sslmode>
export PGSSLROOTCERT=<ca-file-directory>
```

gsql -h *<host>* **-p** *<port>* **-d** *<database>* ***-U*** *<user>*

Table 2-2 Parameter description

Parameter	Description
<host>	IP address of the DB instance. To obtain the IP address, click the instance name on the **Instances** page to go to the **Basic Information** page of the instance. The IP address can be found in the **IP Address** column of the **Node List** area.
<port>	Database port in use. The default value is **8000**. To obtain this parameter, go to the **Basic Information** page of the DB instance. The port number can be found in the **Database Port** field in the **Network Information** area.
<database>	Name of the database (**postgres** by default).
<user>	Username of the GaussDB database account. The default administrator is **root**.
<ca-file-directory>	Path of the CA certificate for SSL connection.
<sslmode>	SSL connection mode. Set it to **verify-ca** to use a CA to check whether the service is trusted.

For example, to connect to a **postgres** database through an SSL connection as user **root**, run the following commands on the ECS:

```
export PGSSLMODE="verify-ca"
export PGSSLROOTCERT="/home/Ruby/ca.pem"
```

gsql -d *postgres* **-h** *10.0.0.0* **-U** *root* **-p** *8000*

```
Password for user root:
```

For more information about gsql commands, see ***Tool Reference***.

g. Check the command output after you log in to the database. If information similar to the following is displayed, the SSL connection has been established.

```
SSL connection (cipher: DHE-RSA-AES256-GCM-SHA384, bits: 256)
```

Follow-up Operations

After logging in to the instance, you can create alarm rules, set automated backup policies, create databases, and migrate data. For details, see:

- **Creating an Alarm Rule**
- **Setting an Automated Backup Policy**
- **Creating a Database Using SQL**
- **Creating a Database Through an API**
- **Managing GaussDB Databases Through DAS**
- **Migrating Data to GaussDB**

3 GaussDB Common Practices

After buying and connecting to a GaussDB instance, you can use GaussDB as required by referring to the common practices provided in this section.

Table 3-1 Common best practices of GaussDB

Practice		Description
Development and design proposal	**Database Object Naming Conventions**	This practice describes the restrictions on naming database objects.
	Database Object Design	This practice describes the principles for designing databases, schemas, tables, fields, constraints, views, and joined tables.
	Tool Interconnection	This practice describes notes about configuring tools.
	SQL Compilation	This practice describes the principles for performing DDL operations, loading and unloading data, converting types, and performing queries.
Database usage	**Creating a Database Account**	This practice describes how to use CREATE USER and ALTER USER to create and manage database users, respectively. Data is not shared between users.
	Creating and Managing Databases	This practice describes how to use CREATE DATABASE and ALTER DATABASE to create and manage database users, respectively.

Practice		Description
	Creating and Managing Tablespaces	The administrator can use tablespaces to control the layout of disks where a database is installed. This practice describes how to use SQL syntax to create and manage tablespaces.
	Creating and Managing Tables	A table is created in a database and can be stored in different databases. Tables under different schemas in a database can have the same name. This practice describes how to use SQL syntax to create tables.
	Querying System Catalogs	In addition to the created tables, GaussDB contains many system catalogs. These system catalogs contain information about instance installation as well as the queries and processes running in GaussDB. This practice describes how to collect information about the databases by querying system catalogs.
Data migration	**Using DRS to Migrate Data from Oracle Database to GaussDB**	This practice describes how to use DRS to migrate data from Oracle databases to GaussDB.
Data backup	**Configuring an Automated Backup Policy**	This practice describes how to enable the system to automatically create backups for a DB instance during a backup window and save the backups based on the configured retention period.
	Creating a Manual Backup	This practice describes how to create manual backups for a DB instance. These backups can be used to restore data for improved reliability.
	Deleting a Manual Backup	You can delete manual backups for instances and tables to release storage space. This practice describes how to delete manual backups of an instance.
Data restoration	**Restoring a Backup File to a DB Instance**	This practice describes how to use an automated or manual backup to restore a DB instance to how it was when the backup was created. The restoration is at the instance level.

Practice		Description
	Restoring a DB Instance to a Specific Point in Time	This practice describes how to use an automated backup to restore instance data to a specified point in time.

