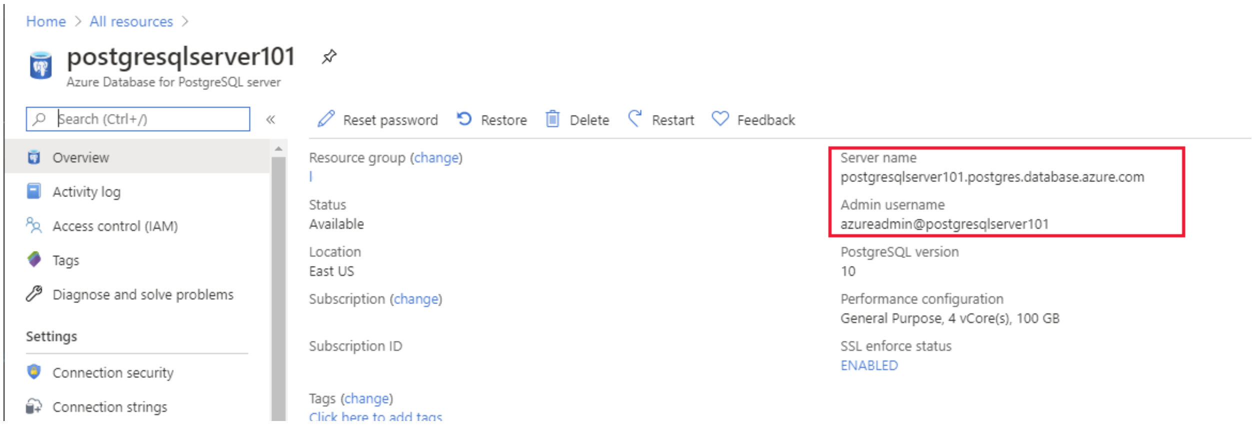
Query relational data in Azure Database for PostgreSQL

PostgreSQL provides many tools you can use to connect to a PostgreSQL database and run queries. These tools include the **pgAdmin** graphical user interface and the **psql** command-line utility. There are a large number of third-party utilities you can use as well.

In this reading, you'll see how to connect to a PostgreSQL database running in Azure Database for PostgreSQL from the command line using **psql**, and from Azure Data Studio.

Retrieve connection information for Azure Database for PostgreSQL

To connect to a PostgreSQL database, you require the name of the server, and the credentials for an account that has access rights to connect to the server. You can find the server name and the name of the default administrator account on the **Overview** page for the Azure Database for PostgreSQL instance in the Azure portal. Contact your administrator for the password.



Resource group (change)

Status
Available

Location
East US

Subscription (change)

Subscription ID

Tags (change)
Click here to add tags

Server name
postgresqlserver101.postgres.database.azure.com

Admin username
azureadmin@postgresqlserver101

PostgreSQL version
10

Performance configuration
General Purpose, 4 vCore(s), 100 GB

SSL enforce status
ENABLED

As with Azure SQL Database, you must open the PostgreSQL firewall to enable client applications to connect to the service. For detailed information, see Firewall rules in Azure Database for PostgreSQL - Single Server

Use psql to query a database

The **psql** utility is available in the Azure Cloud Shell. You can also run it from a command prompt on your desktop computer, but you must download and install the psql client. You can find the psql client on the postgresql.org website.

To connect to Azure Database for PostgreSQL using **psql**, perform the following operations:

1. Run the following command. Make sure to replace the server name and admin name with the values from the Azure portal.

Bash

```
psql --host=<server-name>.postgres.database.azure.com --username=<admin-user>@<server-name> --dbname=postgres
```

Enter your password when prompted.

Note

postgres is the default management database created with Azure Database for PostgreSQL. You can create additional databases using the **CREATE DATABASE** command from psql.

2. If your connection is successful, you'll see the prompt postgres=>.

3. You can create a new database with the following SQL command:

```
CREATE DATABASE "Adventureworks";
```

Note

postgres is the default management database created with Azure Database for PostgreSQL. You can create additional databases using the **CREATE DATABASE** command from psql.

4. Inside psql, you can run the command \c Adventureworks to connect to the database.

5. You can create tables and insert data using CREATE and INSERT commands, as shown in the following examples:

```
CREATE TABLE PEOPLE(NAME TEXT NOT NULL, AGE INT NOT NULL);

INSERT INTO PEOPLE(NAME, AGE) VALUES ('Bob', 35);

INSERT INTO PEOPLE(NAME, AGE) VALUES ('Sarah', 28);

CREATE TABLE LOCATIONS(CITY TEXT NOT NULL, STATE TEXT NOT NULL);

INSERT INTO LOCATIONS(CITY, STATE) VALUES ('New York', 'NY');

INSERT INTO LOCATIONS(CITY, STATE) VALUES ('Flint', 'MI');
```

6. You can retrieve the data you just added using the following SQL commands:

```
SELECT * FROM PEOPLE;

SELECT * FROM LOCATIONS;
```

7. Other psql commands include:

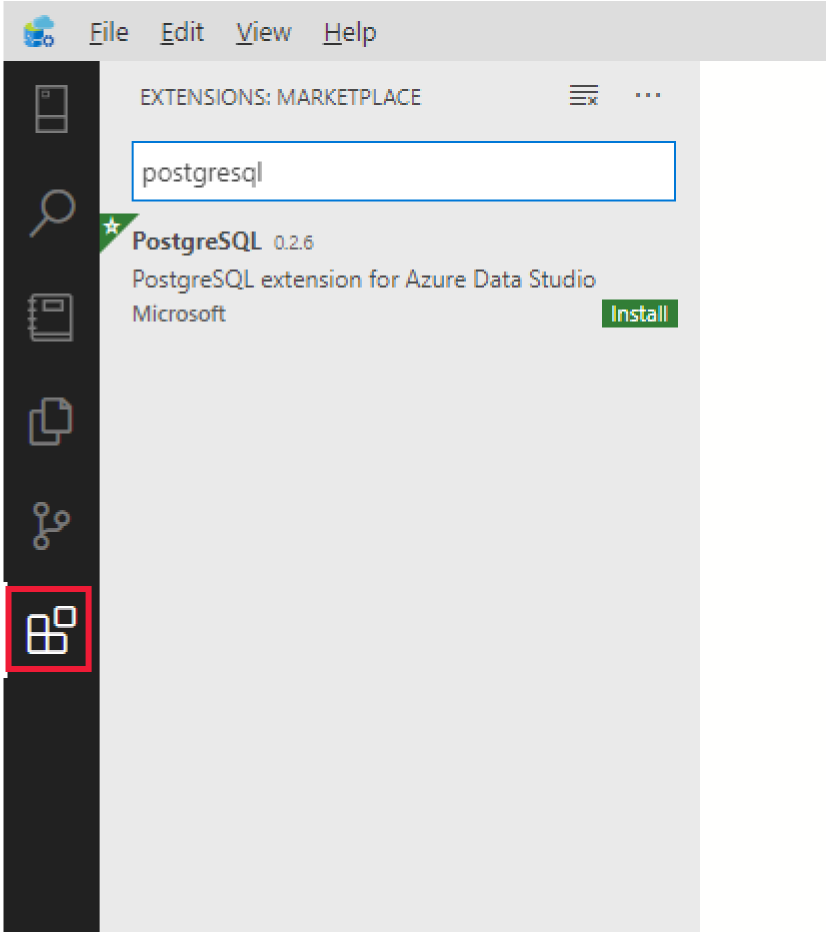
- \l to list databases.
- \dt to list the tables in the current database.

8. You can use the \q command to quit psql.

Connect to PostgreSQL database using Azure Data Studio

To connect to Azure Database for PostgreSQL from Azure Data Studio, you must first install the PostgreSQL extension.

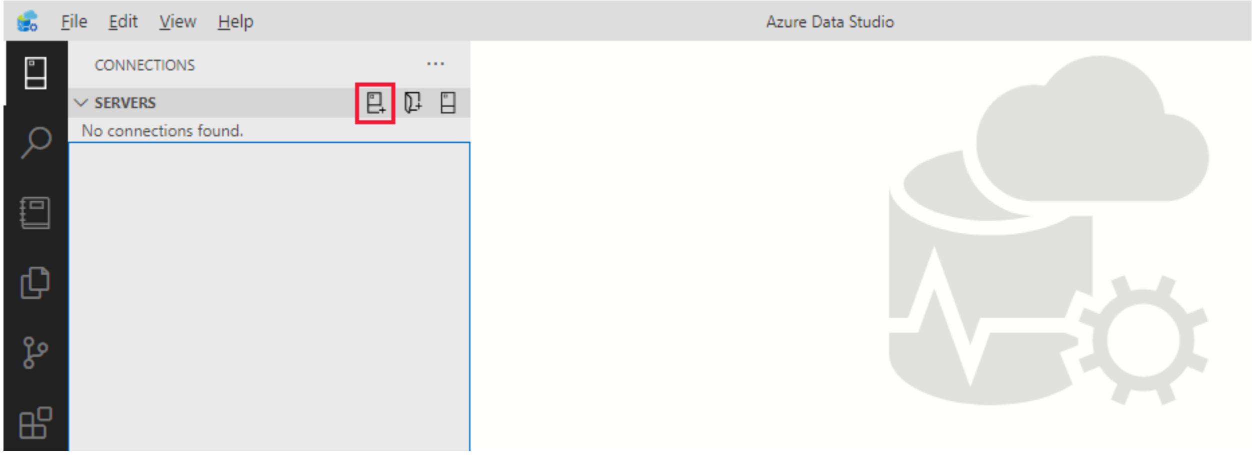
1. On the **Extensions** page, search for **postgresql**.



2. Select the **PostgreSQL** extension, and then select **Install**.

You can then use the extension to connect to PostgreSQL:

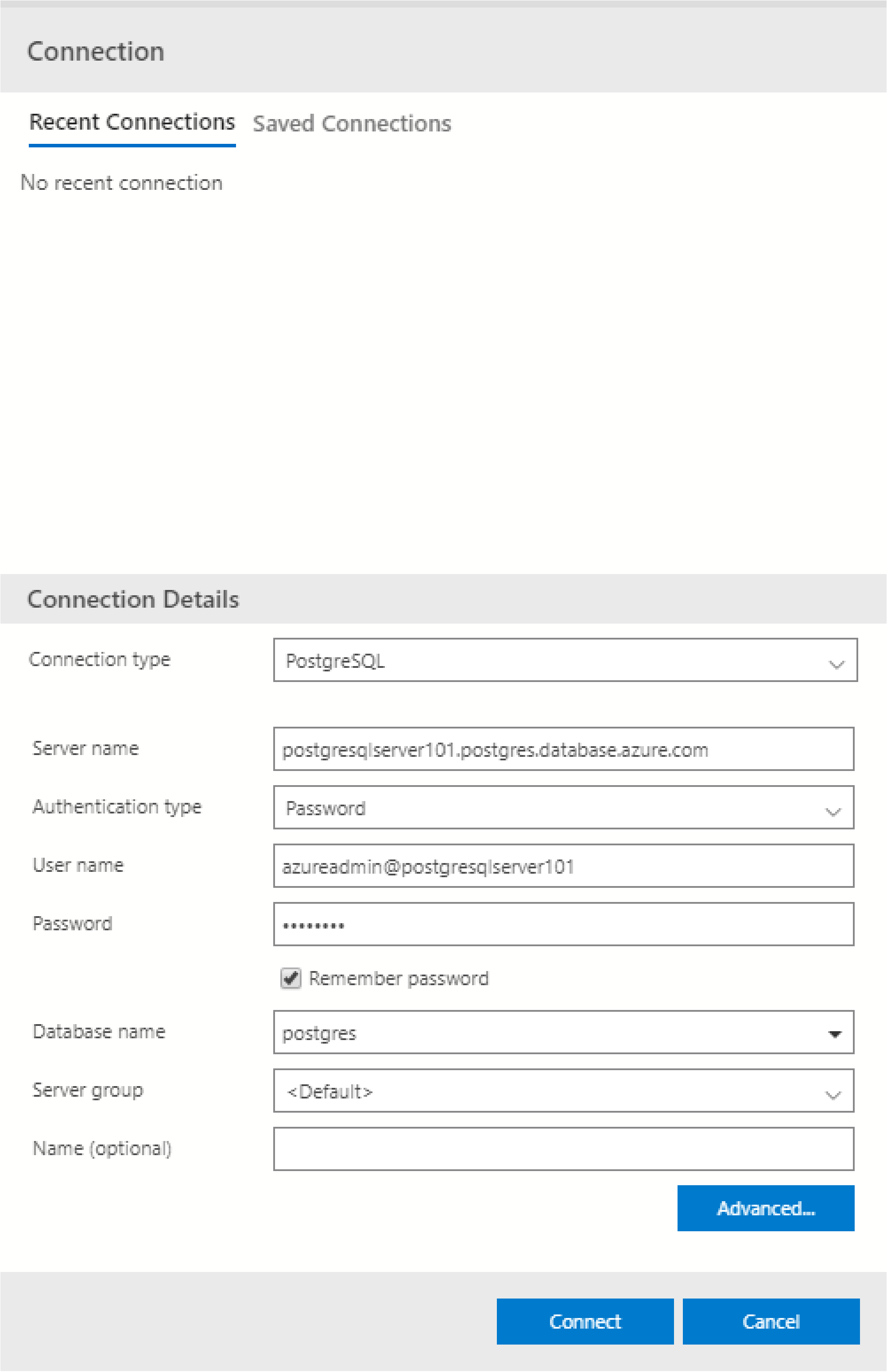
1. In Azure Data Studio, go to the **SERVERS** sidebar, and select **New Connection**.



The SERVERS sidebar in Azure Data Studio. The user is creating a new connection

2. In the **Connection** dialog box, in the **Connection type** drop-down list box, select **PostgreSQL**.

3. Fill in the remaining fields using the server name, user name, and password for your PostgreSQL server.



Connection

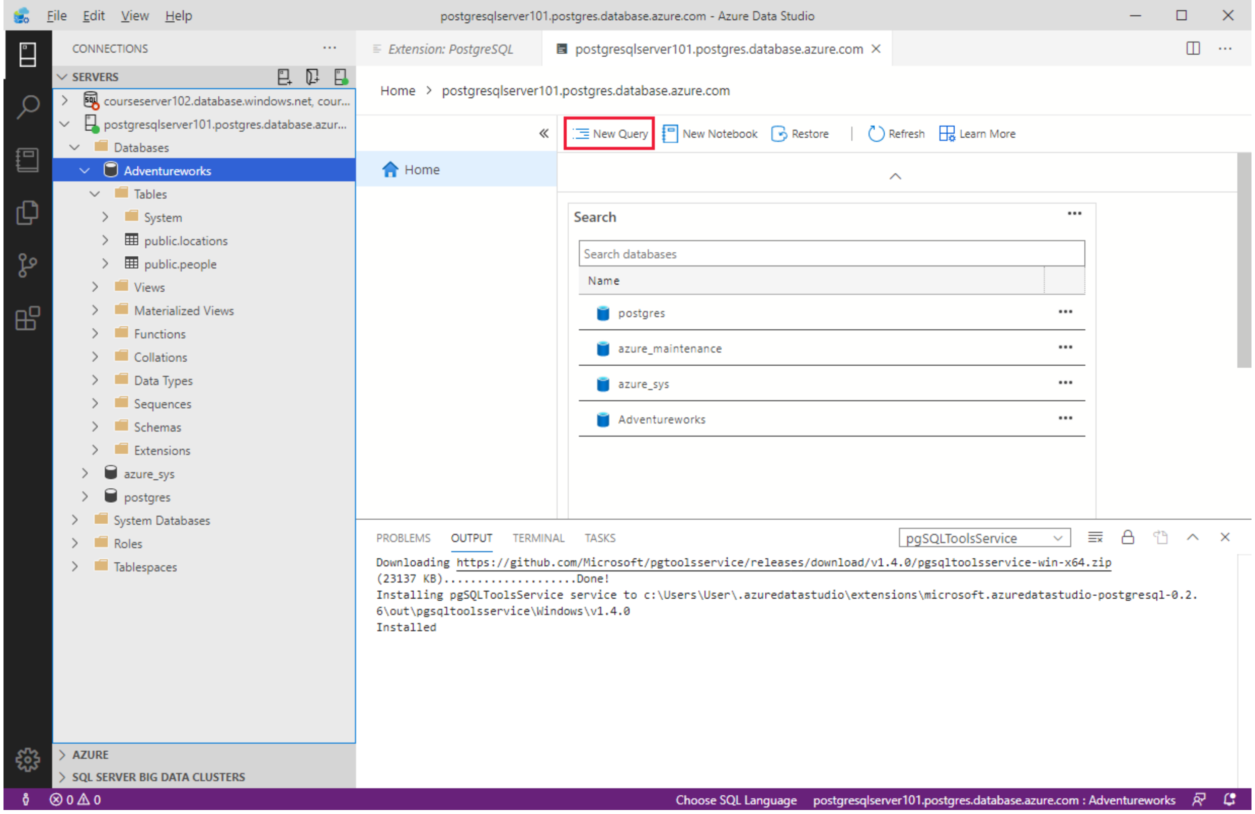
No recent connection

Connection Details

Connection type	PostgreSQL
Server name	postgresqlserver101.postgres.database.azure.com
Authentication type	Password
User name	azureadmin@postgresqlserver101
Password	••••••••
	☑ Remember password
Database name	postgres
Server group	<Default>
Name (optional)	

Setting	Description
Server Name	The fully qualified server name from the Azure portal.
User name	The user name you want to sign in with. This must be in the format shown in the Azure portal, <username>@<hostname>.
Password	The password for the account you're logging in with.
Database name	Fill this if you want the connection to specify a database.
Server Group	This option lets you assign this connection to a specific server group you create.
Name (optional)	This option lets you specify a friendly name for your server.

4. Select **Connect** to establish the connection. After successfully connecting, your server opens in the **SERVERS** sidebar. You can expand the **Databases** node to connect to databases on the server and view their contents. Use the **New Query** command in the toolbar to create and run queries.



```
Downloading https://github.com/Microsoft/pgtoolsservice/releases/download/v1.4.0/pgsqltoolsservice-win-x64.zip
(23137 KB)....................Done!
Installing pgSQLToolsService service to c:\Users\User\.azuredatastudio\extensions\microsoft.azuredatastudio-postgresql-0.2.
6\out\pgsqltoolsservice\Windows\v1.4.0
Installed
```

The following example adds a new table to the database and inserts four rows.

```
);
```

```
-- Create a new table called 'customers'

CREATE TABLE customers(

    customer_id SERIAL PRIMARY KEY,

    name VARCHAR (50) NOT NULL,

    location VARCHAR (50) NOT NULL,

    email VARCHAR (50) NOT NULL

);
```

```
-- Insert rows into table 'customers'
INSERT INTO customers
    (customer_id, name, location, email)
VALUES
    ( 1, 'Orlando', 'Australia', ''),
    ( 2, 'Keith', 'India', 'keith0@adventure-works.com'),
    ( 3, 'Donna', 'Germany', 'donna0@adventure-works.com'),
    ( 4, 'Janet', 'United States','janet1@adventure-works.com');
```

5. From the toolbar, select **Run** to execute the query. As with Azure SQL, notifications appear in the **MESSAGES** pane to show query progress.

6. To query the data, enter a SELECT statement, and then click **Run**:

```
-- Select rows from table 'customers'
SELECT * FROM customers;
```

7. The results of the query should appear in the results pane.