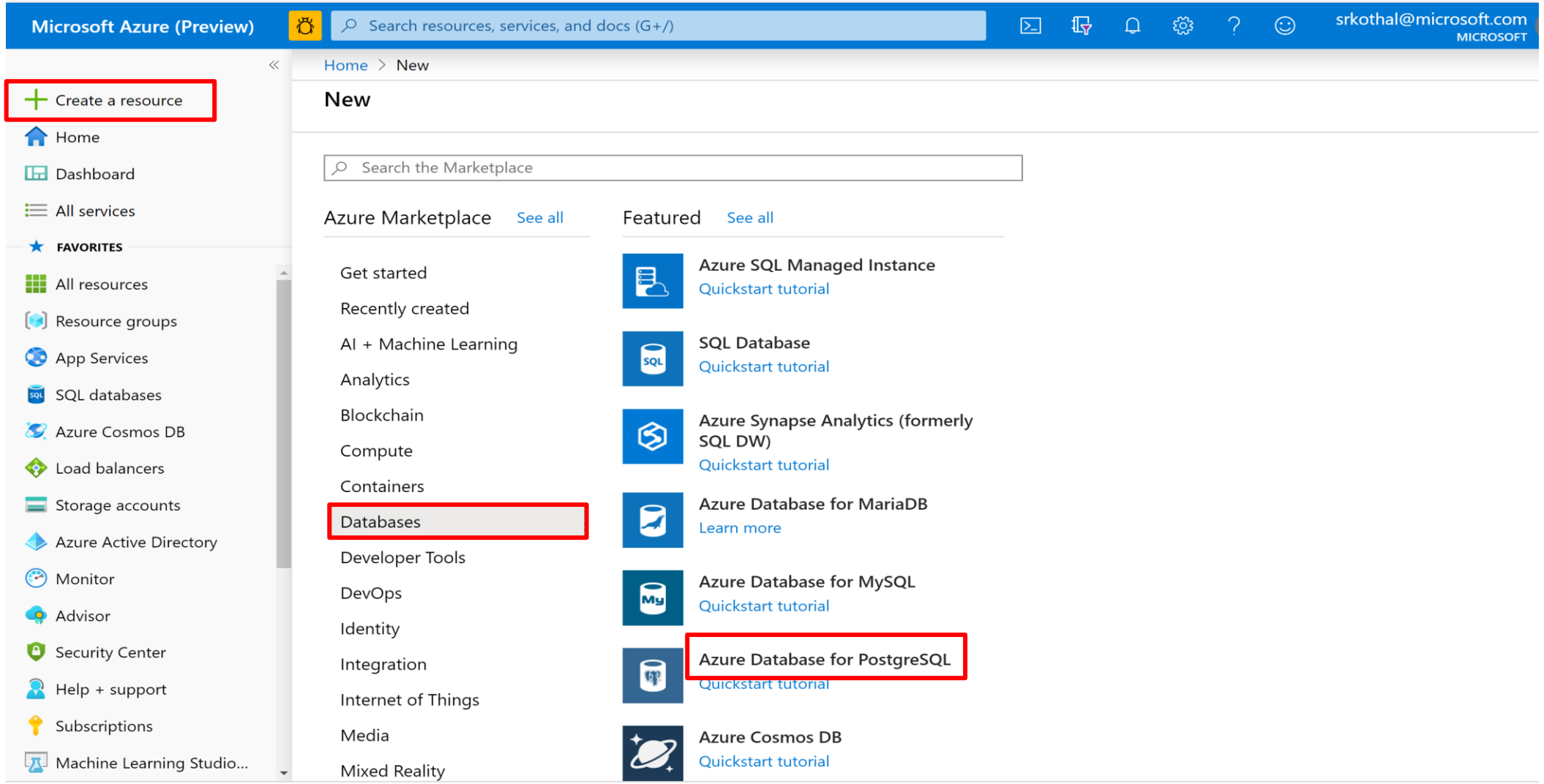
Cisco
Azure DB for PostgreSQL Backup & Restore Demo

Sridhar Kothalanka (CSA-Data&AI)

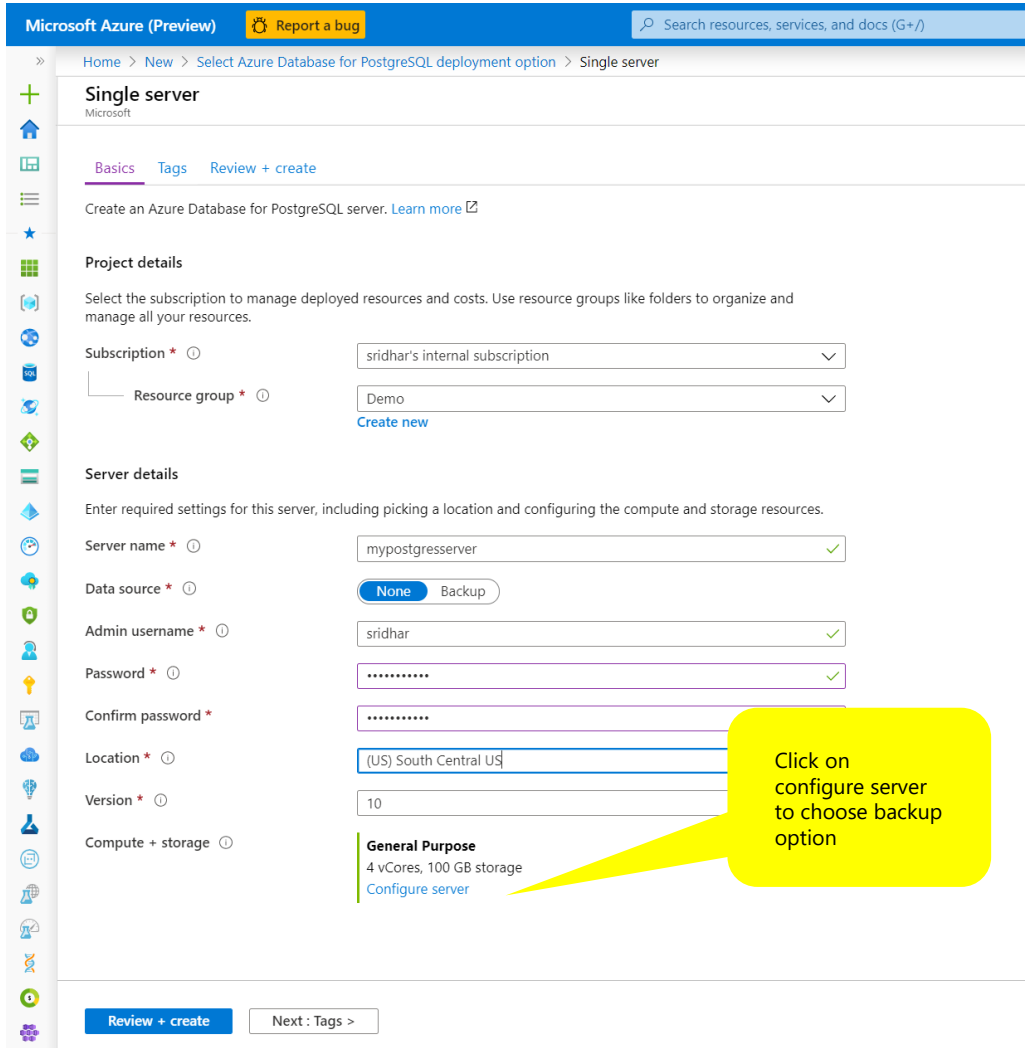
Jan 2020

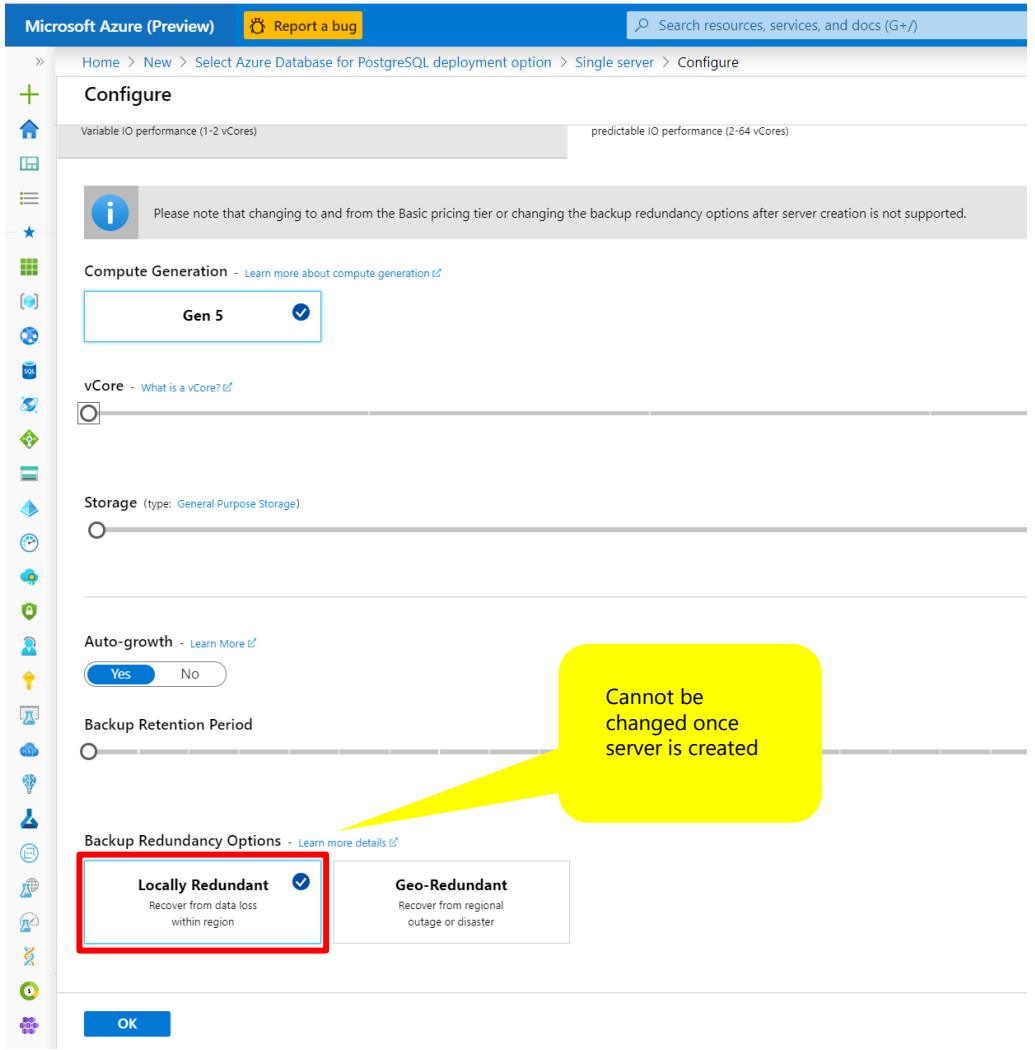
Single server backup and restore using “Locally Redundant” backup option

Create an Azure DB for PostgreSQL resource

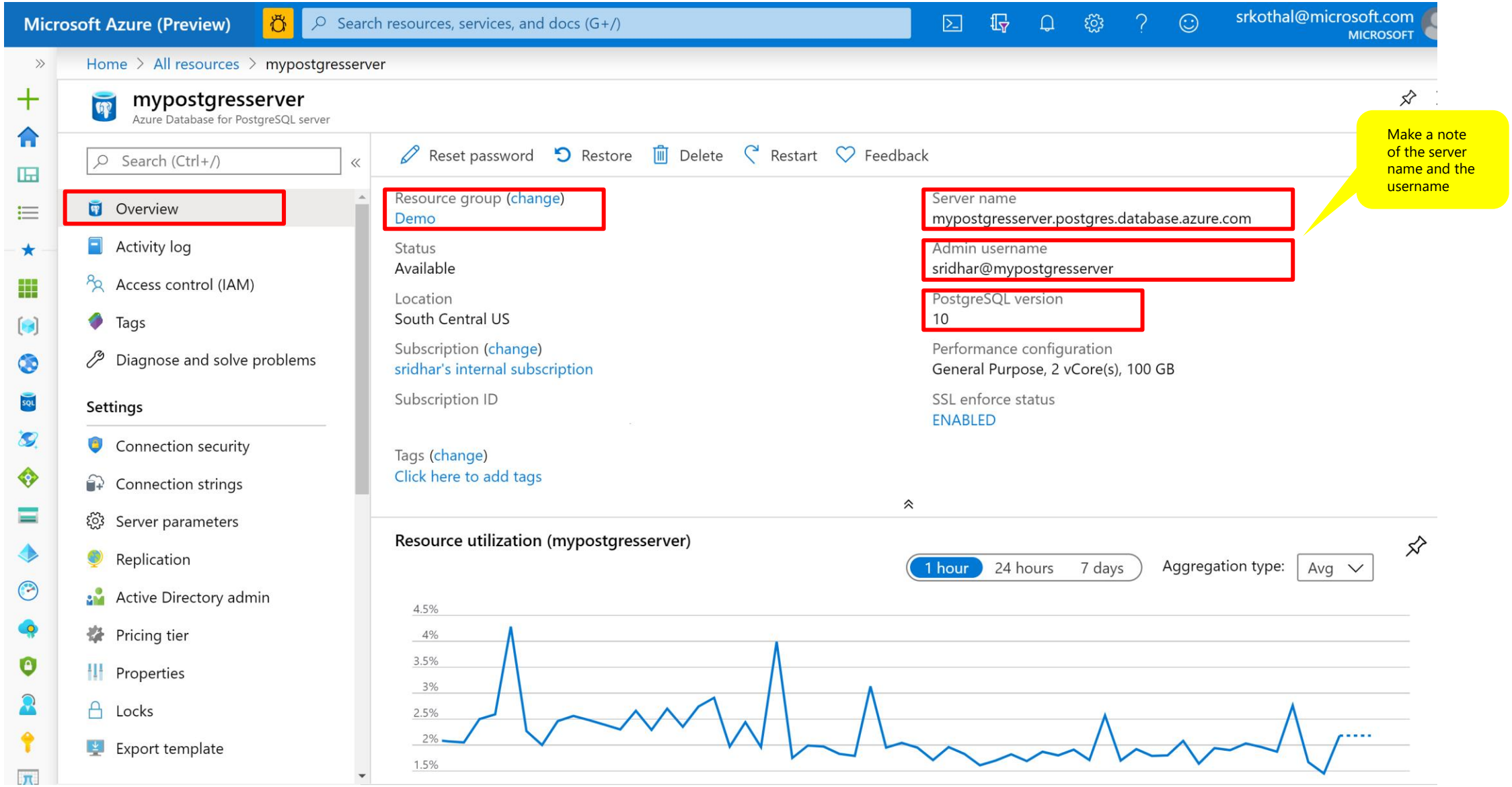


Pick the locally redundant server option

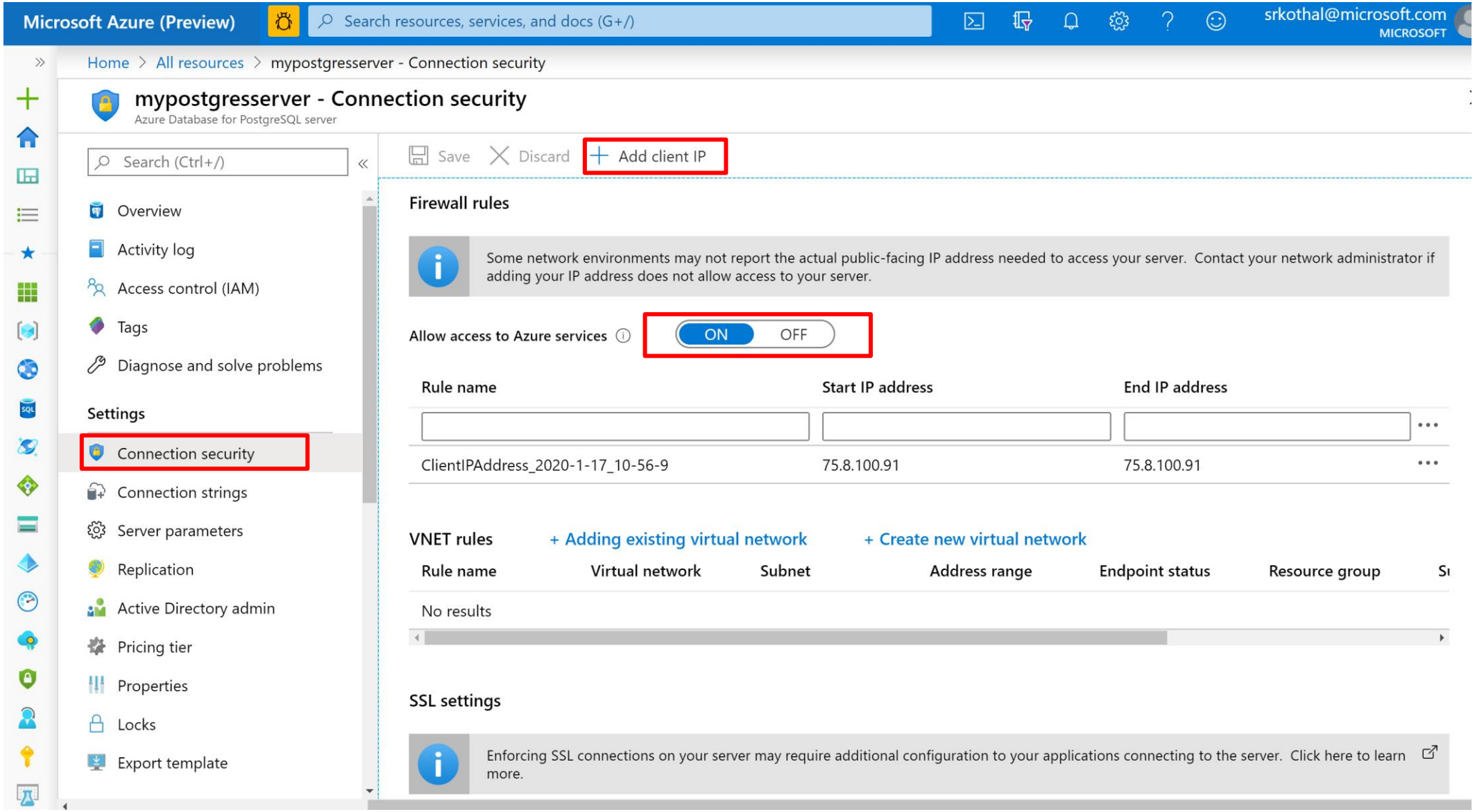




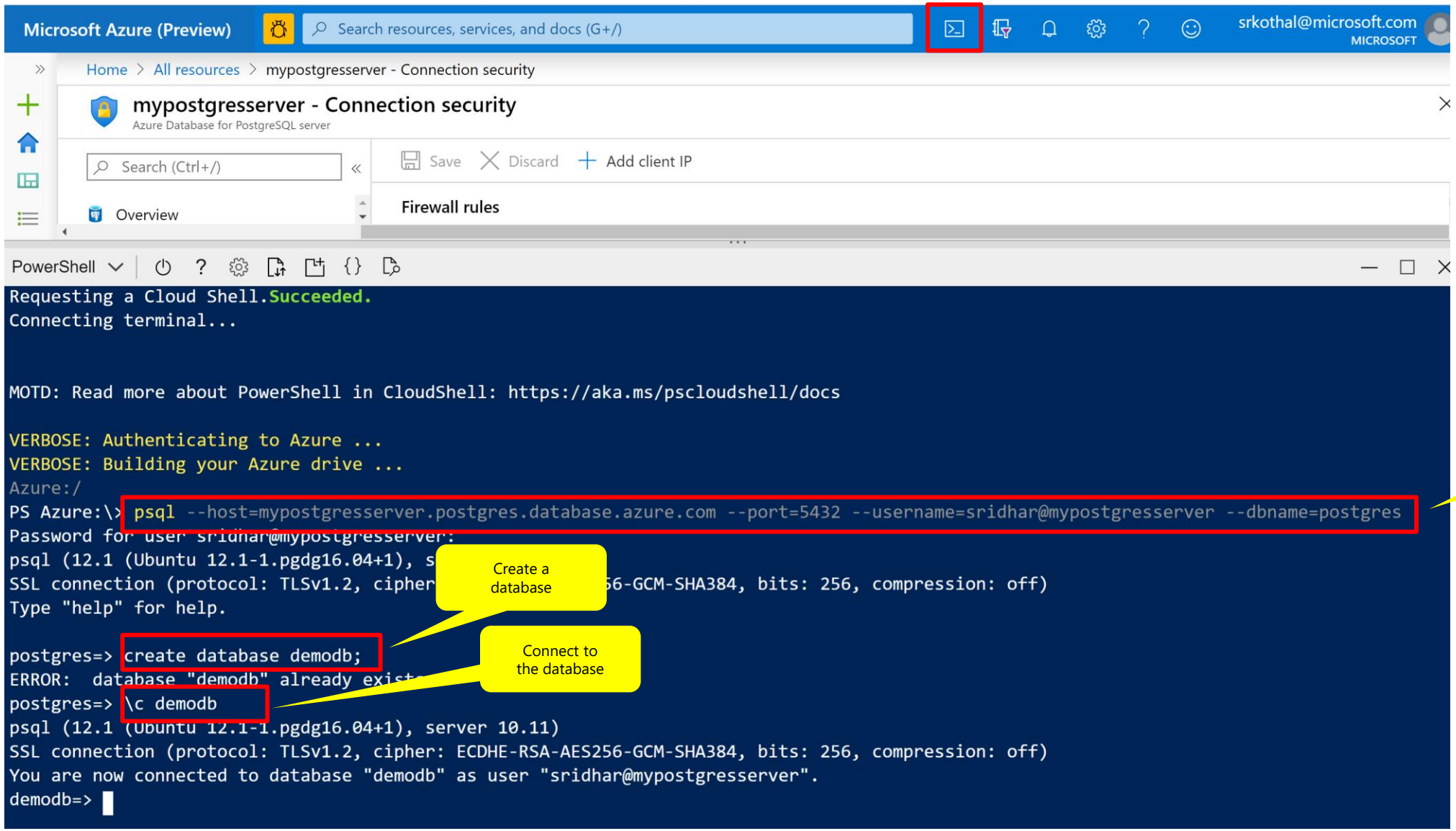
Note Server name & username



Configure the server-level firewall rule



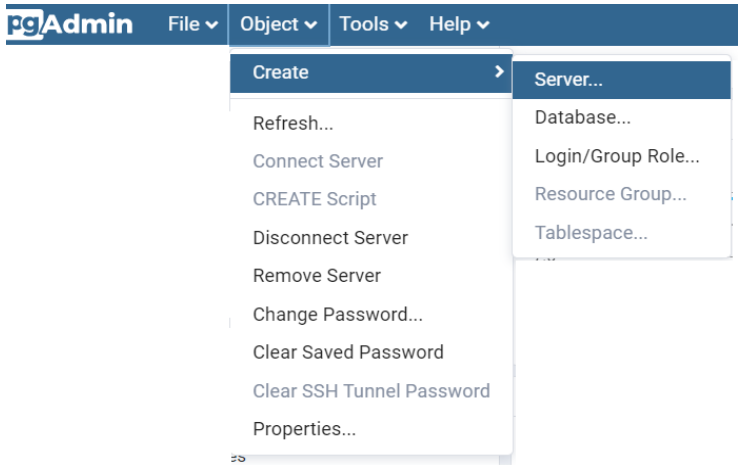
Connect to PostgreSQL server using Cloud Shell



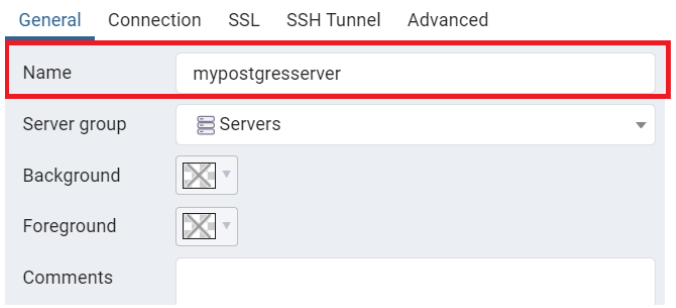
Connect using pg_admin

1. Launch pg_admin and add a server using the tool menu (Object -> Create -> Server…)
2. Input the server name in the “general” tab
3. Input the connection string, port, database name and username in “Connection” tab

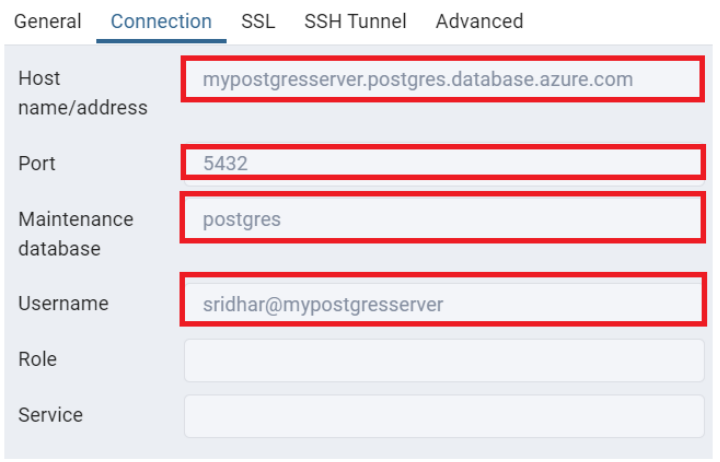
1



2

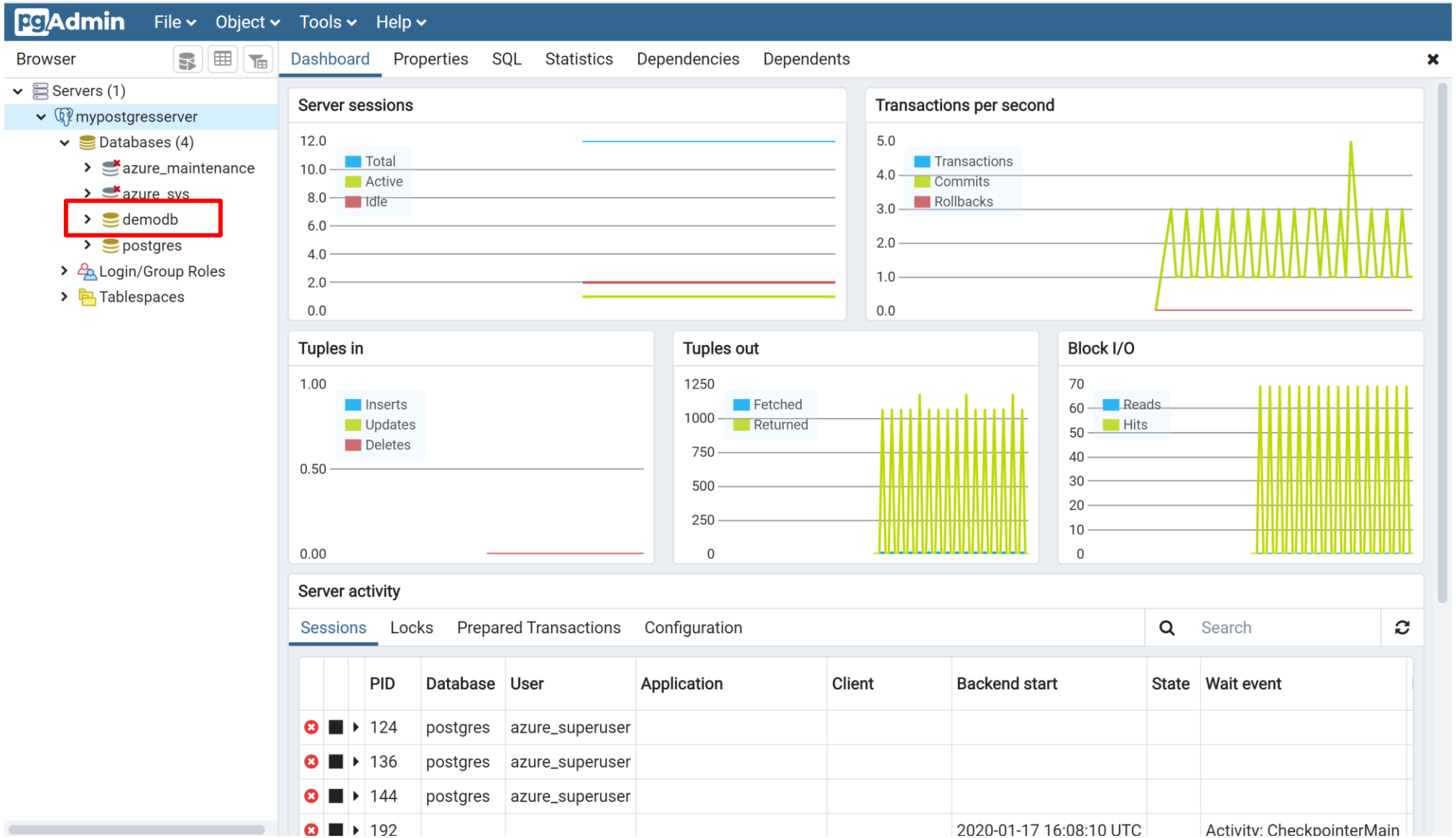


3

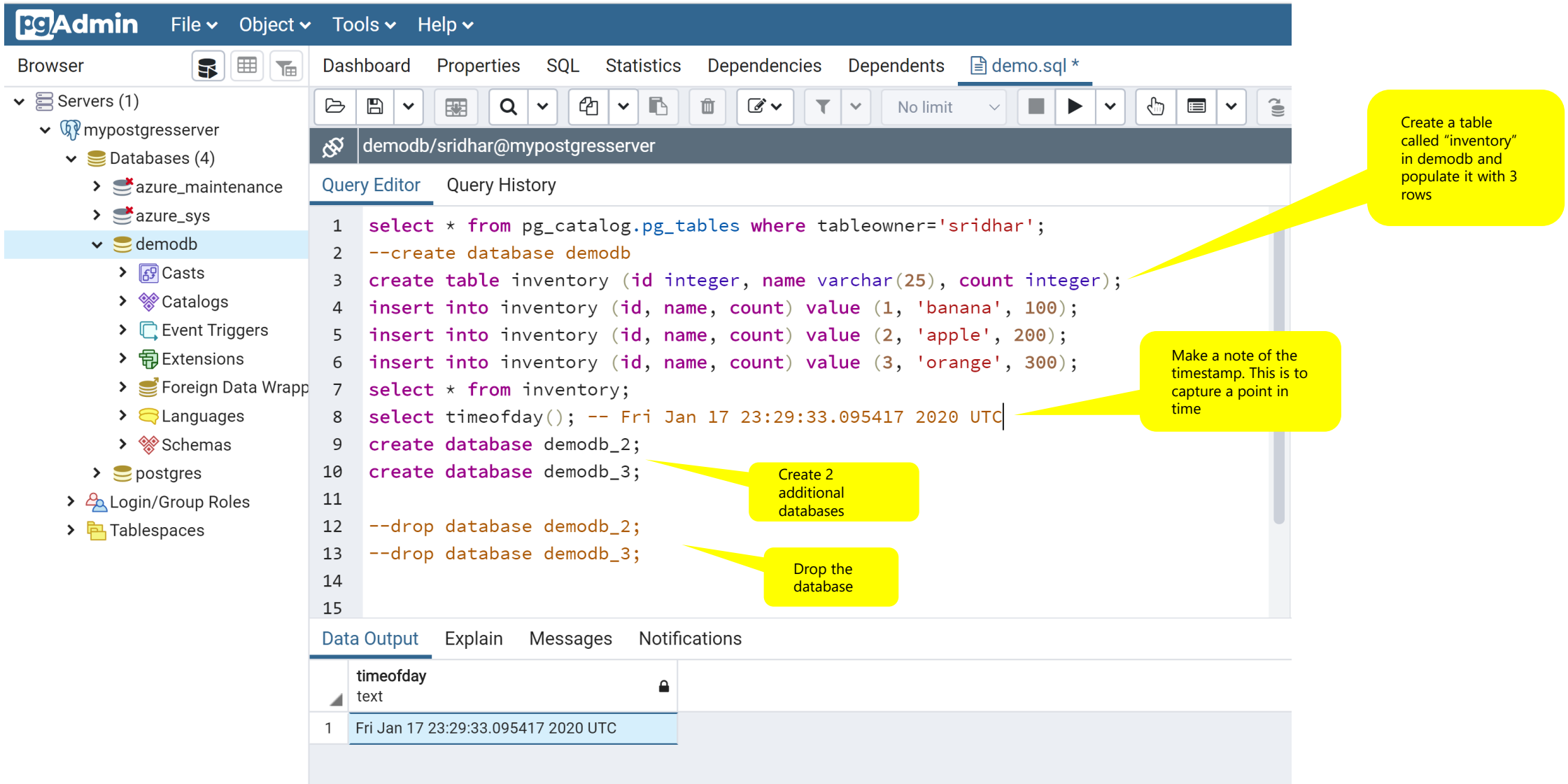


Download pg_Admin and client from PostgreSQL website
https://www.enterprisedb.com/downloads/postgres-postgresql-downloads

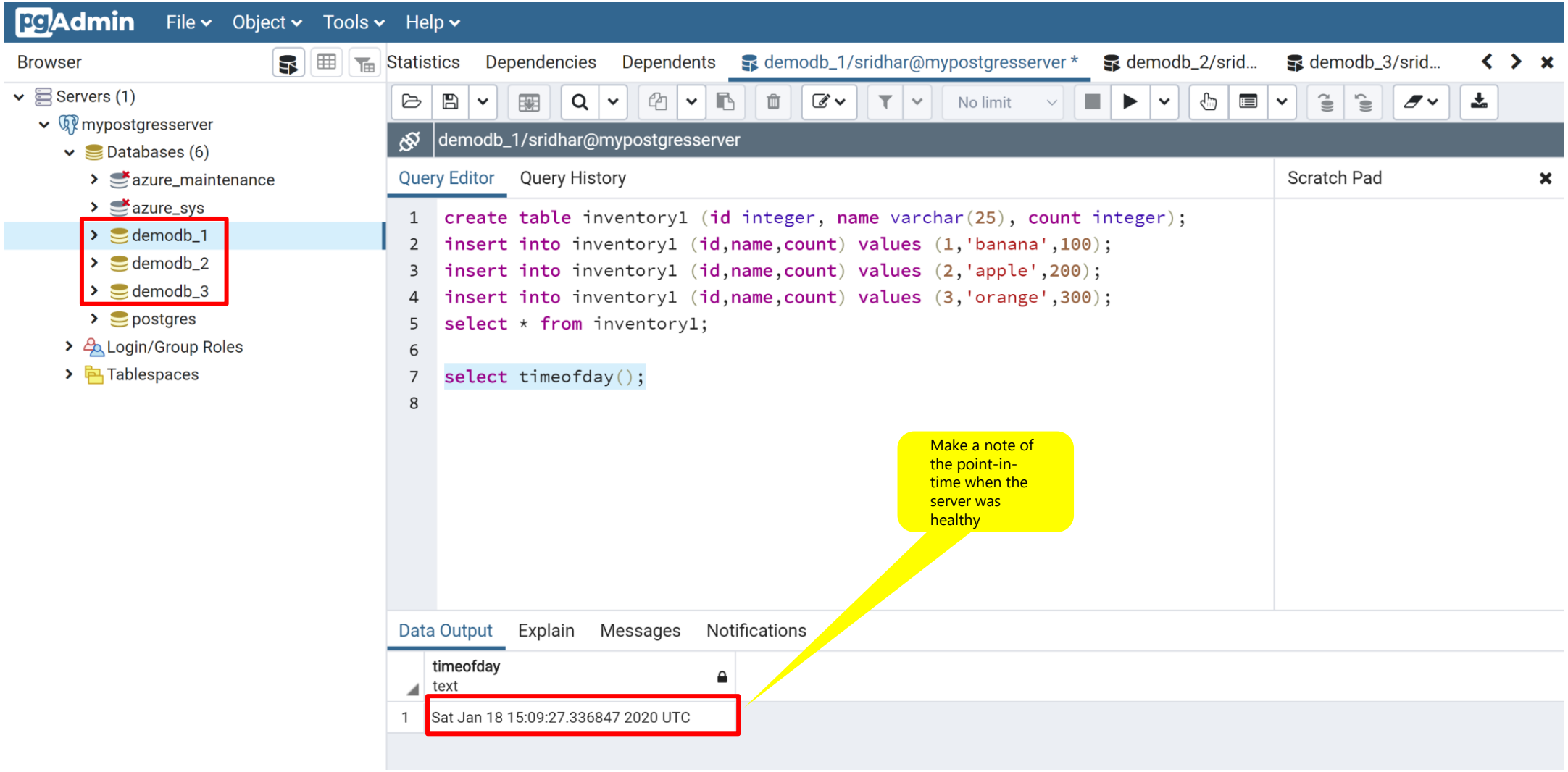
Open pg_admin client



Populate the server with databases and a table



Populate the server with databases and tables



Simulate some unexpected changes

- State of DB at Sat Jan 18 15:09:27.336847 2020 UTC
  - Database: demodb_1
    - Table: inventory1
  - Database: demodb_2
    - Table: inventory2
  - Database: demodb_3
    - Table: inventory3

```
15  select * from inventory1;
```

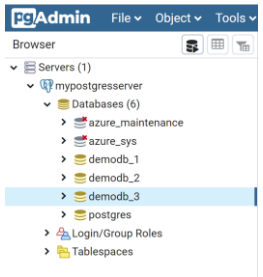
Data Output | Explain | Messages | Notifications

	id integer	name character varying (25)	count integer
1	1	banana	100
2	2	apple	200
3	3	orange	300

```
15  select * from inventory2;
```

Data Output | Explain | Messages | Notifications

	id integer	name character varying (25)	count integer
1	1	banana	100




```
15  select * from inventory3;
```

Data Output | Explain | Messages | Notifications

	id integer	name character varying (25)	count integer
1	1	banana	100
2	2	apple	200
3	3	orange	300

- Current state of DB
  - Database: demodb_1
    - Table: inventory1
  - Database: demodb_2
    - Table: inventory2
  - Database: demodb_3
    - Table: inventory3

Database is good to use

Inventory table should have only 1 row but has multiple rows

Database got accidentally deleted

```
15  select * from inventory1;
```

Data Output | Explain | Messages | Notifications

	id integer	name character varying (25)	count integer
1	1	banana	100
2	2	apple	200
3	3	orange	300

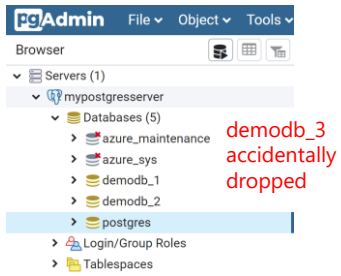
demodb_1 inventory1 table is good

```
18  select * from inventory2;
```

Data Output | Explain | Messages | Notifications

	id integer	name character varying (25)	count integer
1	1	banana	100
2	2	banana	100
3	3	banana	100
4	4	banana	100

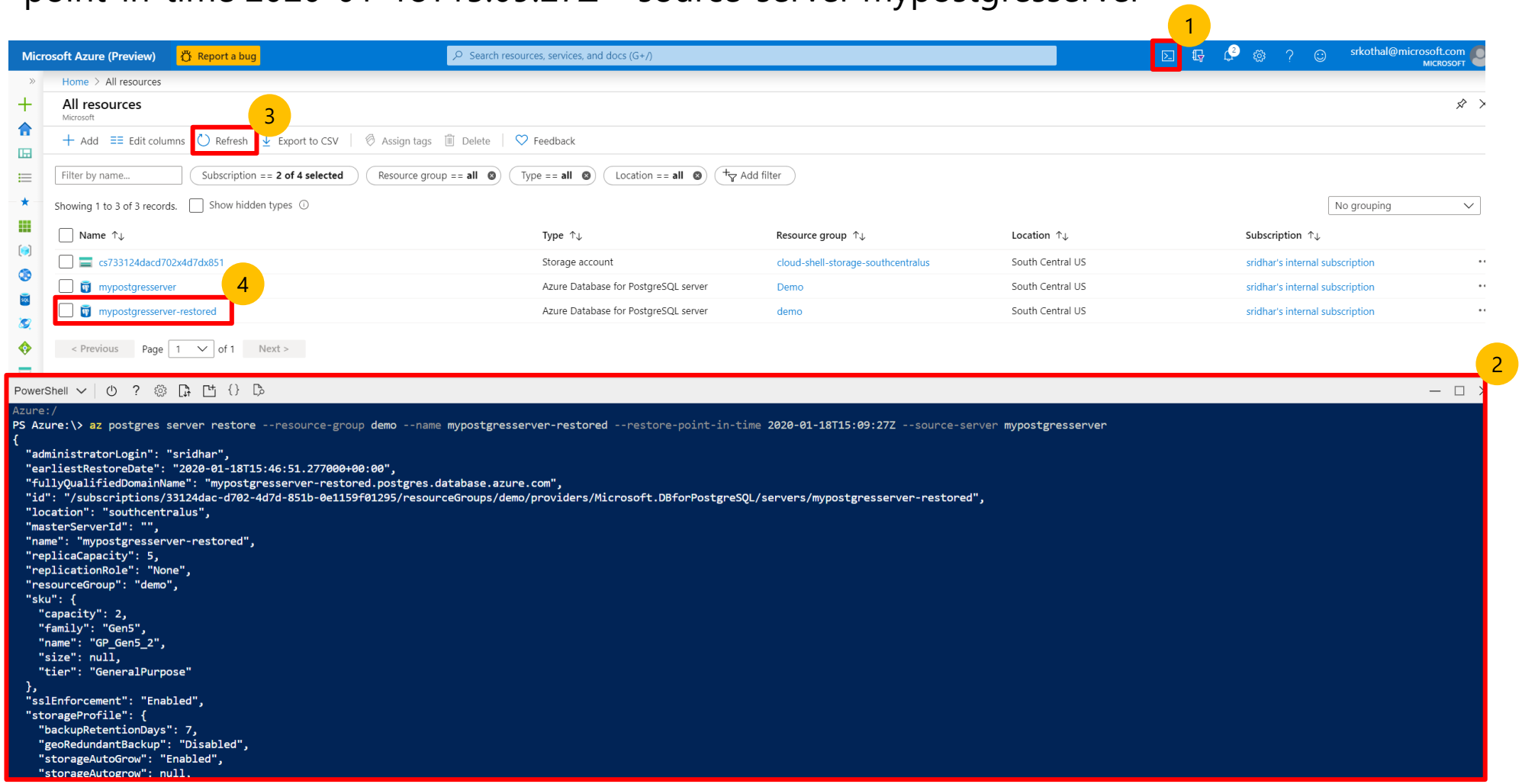
demodb_2 inventory2 table has duplicates



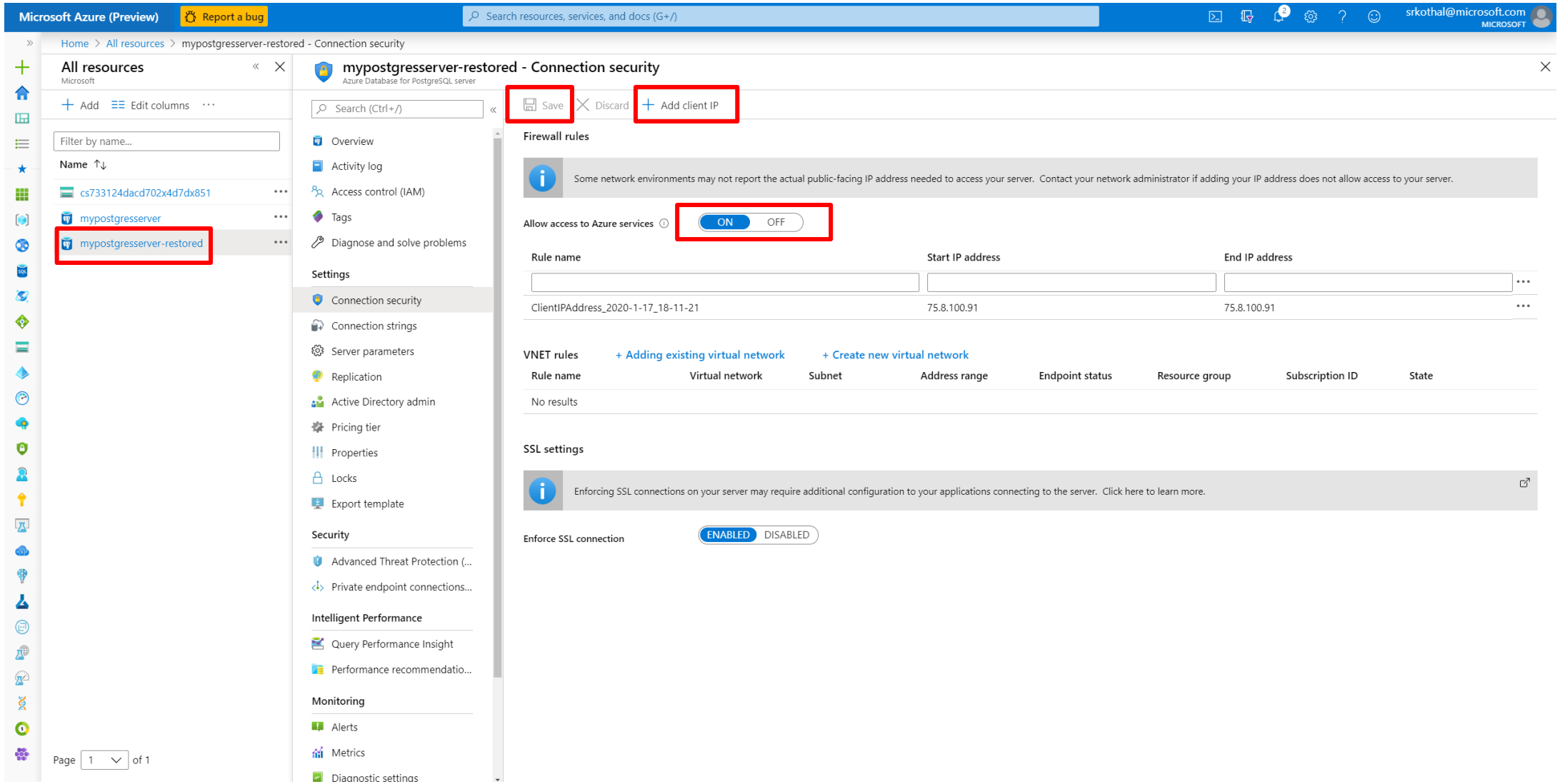
demodb_3 accidentally dropped

Restore the server to a point-in-time

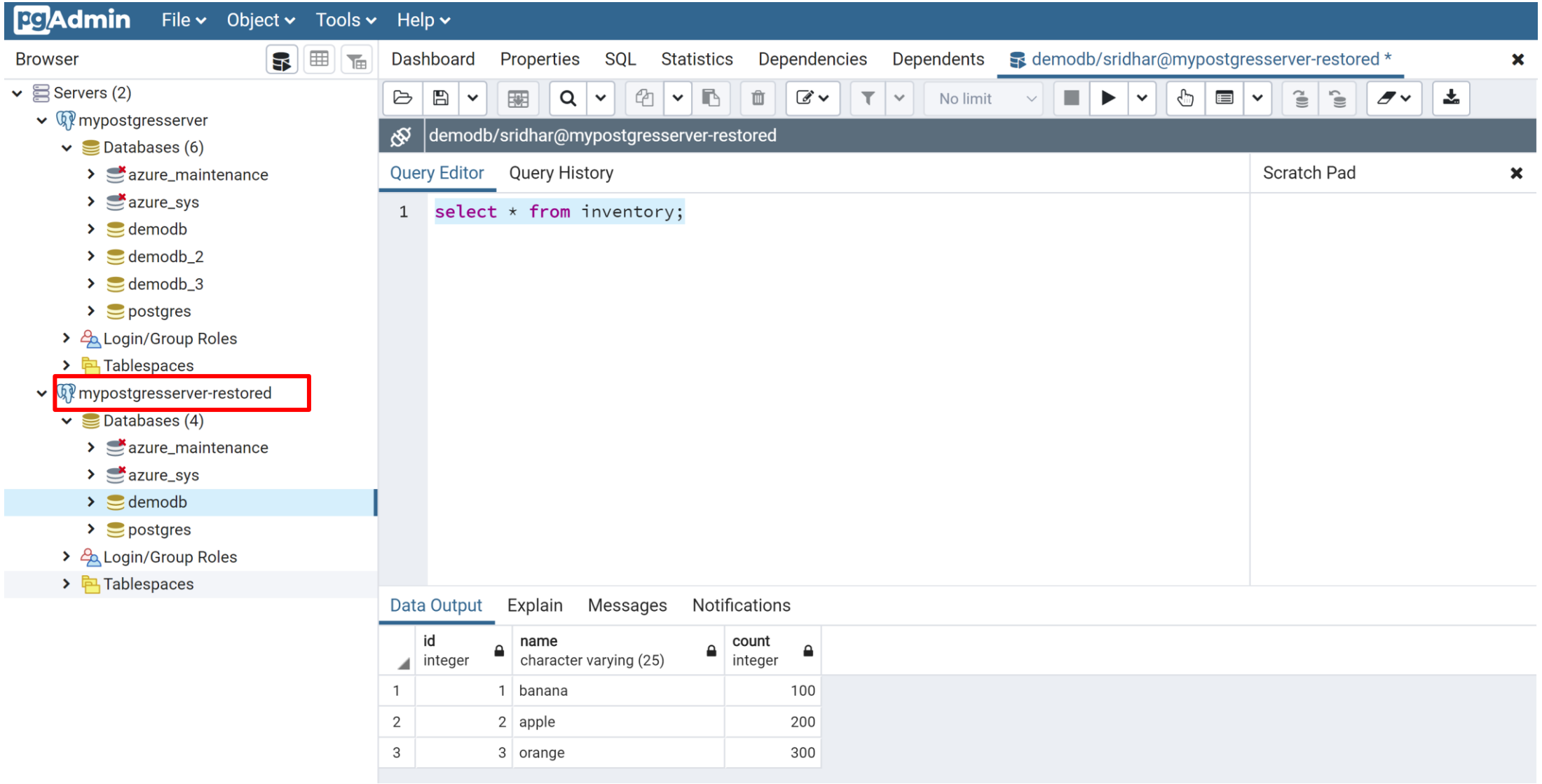
az postgres server restore --resource-group demo --name mypostgresserver-restored --restore-point-in-time 2020-01-18T15:09:27Z --source-server mypostgresserver



Firewall settings on the restored database

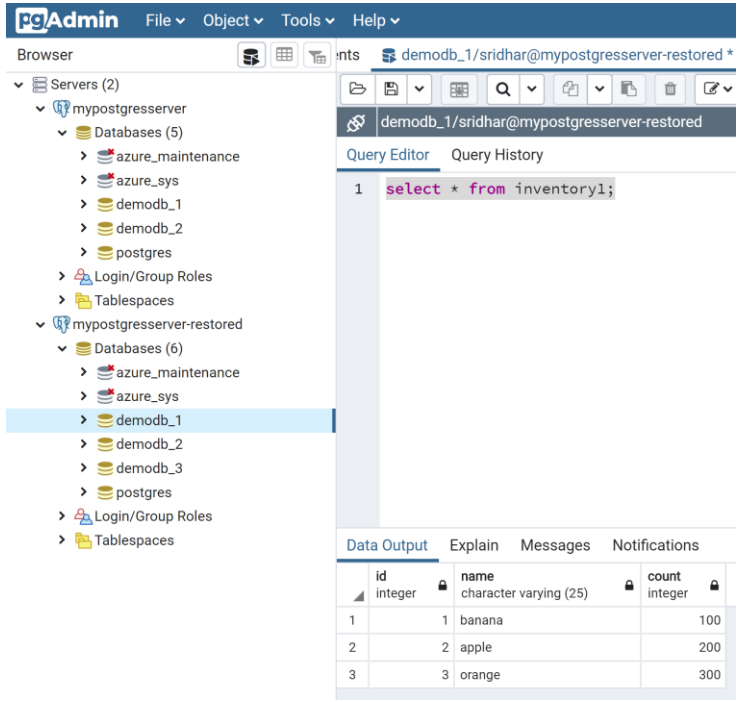


Connect to the restored database

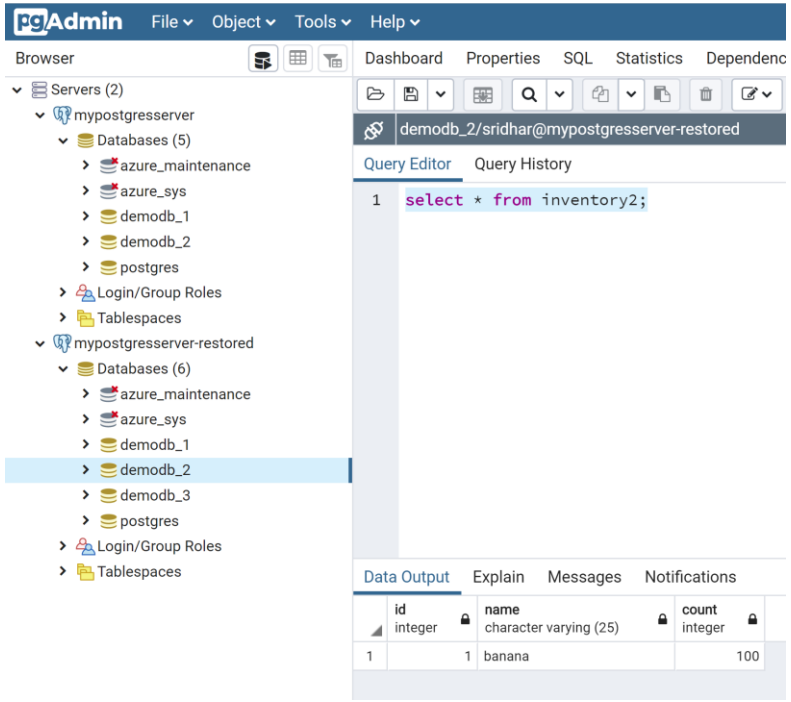


Verify data in restored server data is good

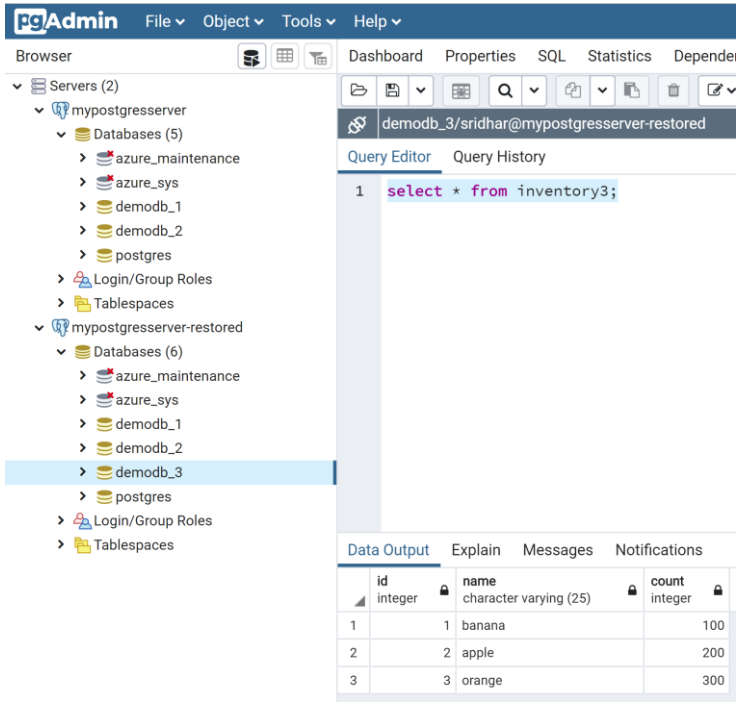
Server: mypostgresserver-restored



demodb_1



demodb_2



demodb_3

Restore the inventory2 table to demodb_2

- Export the inventory2 table from demodb_2 database from mypostgresserver-restored server
  - `pg_dump --host=mypostgresserver-restored.postgres.database.azure.com --username=sridhar@mypostgresserver-restored --dbname=demodb_2 --file=demodb_2.sql --table=inventory2;`
- Drop the table “inventory1” from mypostgresserver server
  - `drop inventory2 from demodb_2`
- Import the inventory2 table to original server
  - `psql --file=demodb_2.sql --host=mypostgresserver.database.windows.net --port=5432 --username=sridhar@mypostgresserver --dbname=demodb_2 --table=inventory2;`

```
PS Azure:\> pg_dump --host=mypostgresserver-restored.postgres.database.azure.com --username=sridhar@mypostgresserver-restored --dbname=demodb_2 --file=demodb_2.sql --table=inventory2;
Password:
Azure:/
PS Azure:\> ls
clouddrive  demodb_2.sql
Azure:/
PS Azure:\> psql --host=mypostgresserver.postgres.database.azure.com --port=5432 --username=sridhar@mypostgresserver --dbname=postgres
Password for user sridhar@mypostgresserver:
psql (12.1 (Ubuntu 12.1-1.pgdg16.04+1), server 10.11)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=> \c demodb_2
psql (12.1 (Ubuntu 12.1-1.pgdg16.04+1), server 10.11)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
You are now connected to database "demodb_2" as user "sridhar@mypostgresserver".
demodb_2=> drop table inventory2;
DROP TABLE
demodb_2=> \q
Azure:/
PS Azure:\> psql --file=demodb_2.sql --host=mypostgresserver.database.windows.net --port=5432 --username=sridhar@mypostgresserver --dbname=demodb_2 --table=inventory2;
Password for user sridhar@mypostgresserver:
```

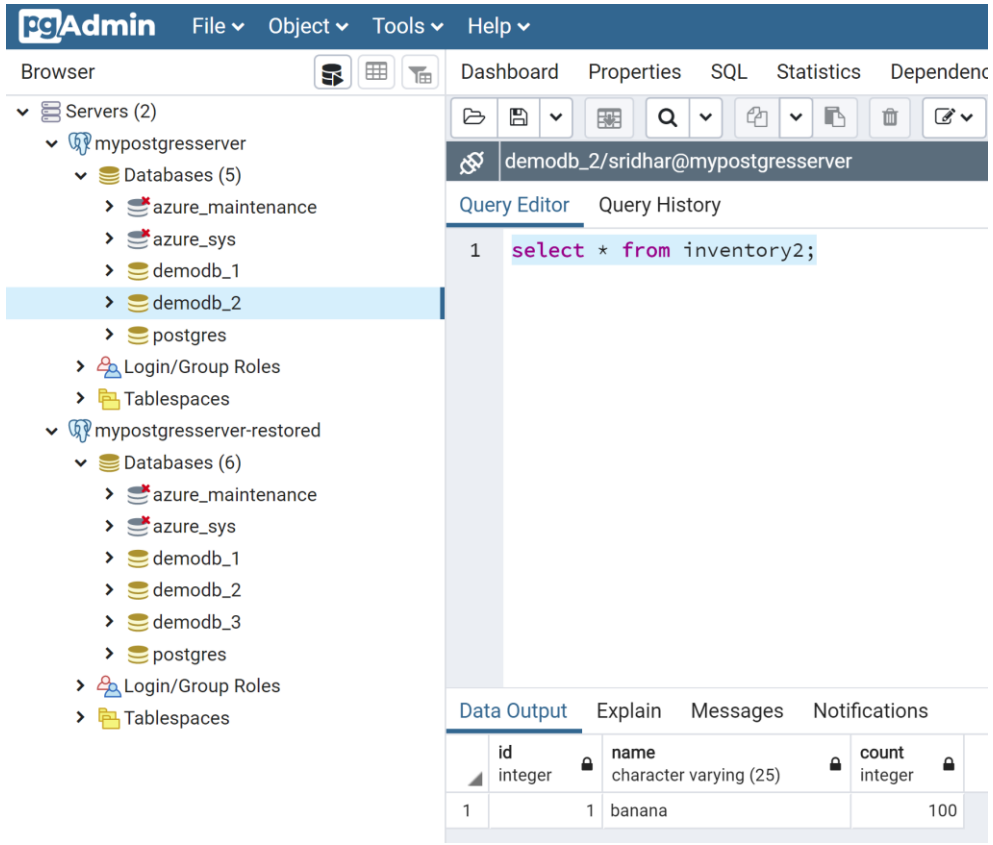
Restore the demodb_3 database

- Export the demodb_3 database from mypostgresserver-restored server
  - pg_dump --host=mypostgresserver-restored.postgres.database.azure.com --username=sridhar@mypostgresserver-restored --dbname=demodb_3 --file=demodb_3.sql;
- Create an empty database demodb_3 in mypostgresserver
  - create database demodb_3;
- Import the demodb_3 to mypostgresserver server
  - psql --file=demodb_3.sql --host=mypostgresserver.database.windows.net --port=5432 --username=sridhar@mypostgresserver --dbname=demodb_3;

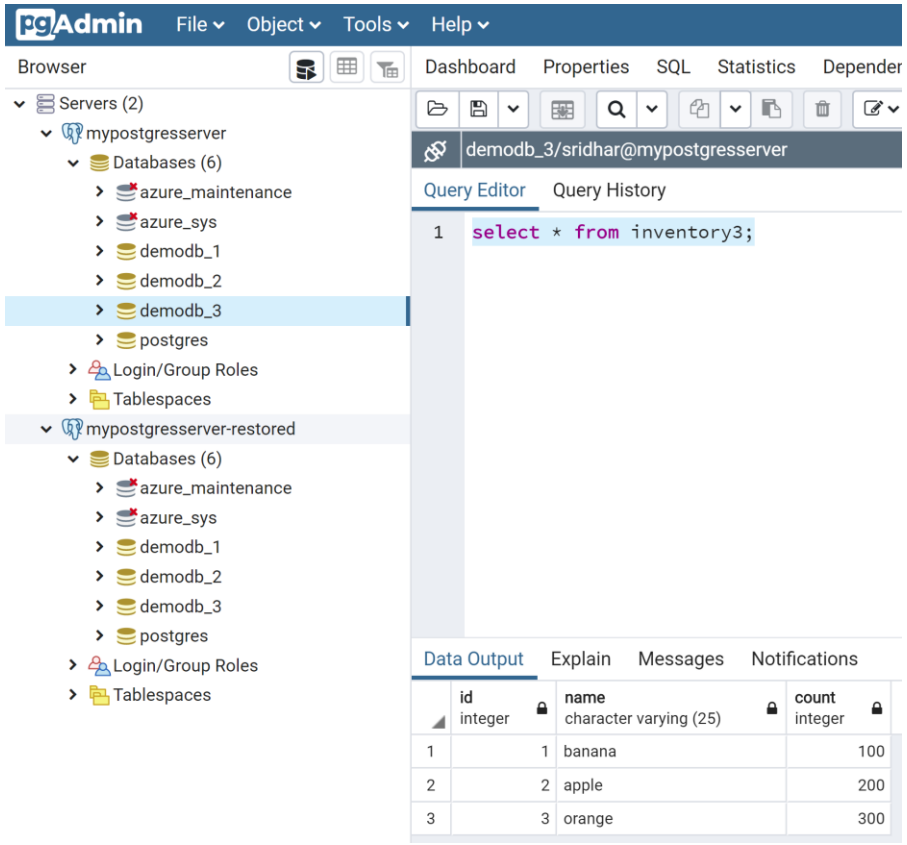
```
PS Azure:\> pg_dump --host=mypostgresserver-restored.postgres.database.azure.com --username=sridhar@mypostgresserver-restored --dbname=demodb_3 --file=demodb_3.sql;
Password:
Azure:/
PS Azure:\> ls
clouddrive  demodb_2.sql  demodb_3.sql
Azure:/
PS Azure:\> psql --host=mypostgresserver.postgres.database.azure.com --port=5432 --username=sridhar@mypostgresserver --dbname=postgres;
Password for user sridhar@mypostgresserver:
psql (12.1 (Ubuntu 12.1-1.pgdg16.04+1), server 10.11)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=> create database demodb_3;
CREATE DATABASE
postgres=> \q
Azure:/
PS Azure:\> psql --file=demodb_3.sql --host=mypostgresserver.database.windows.net --port=5432 --username=sridhar@mypostgresserver --dbname=demodb_3;
Password for user sridhar@mypostgresserver:
```

Verify the data on mypostgresserver after restore



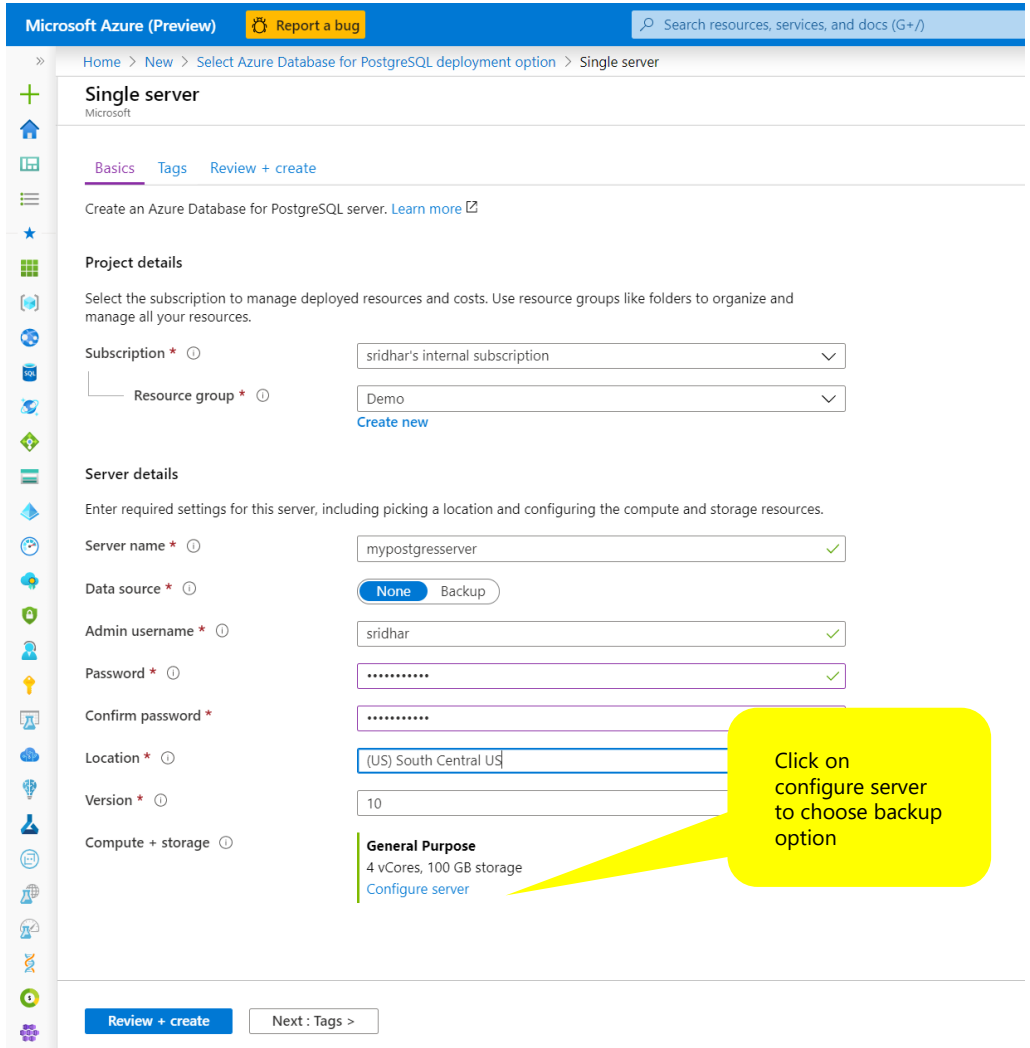
demodb_2

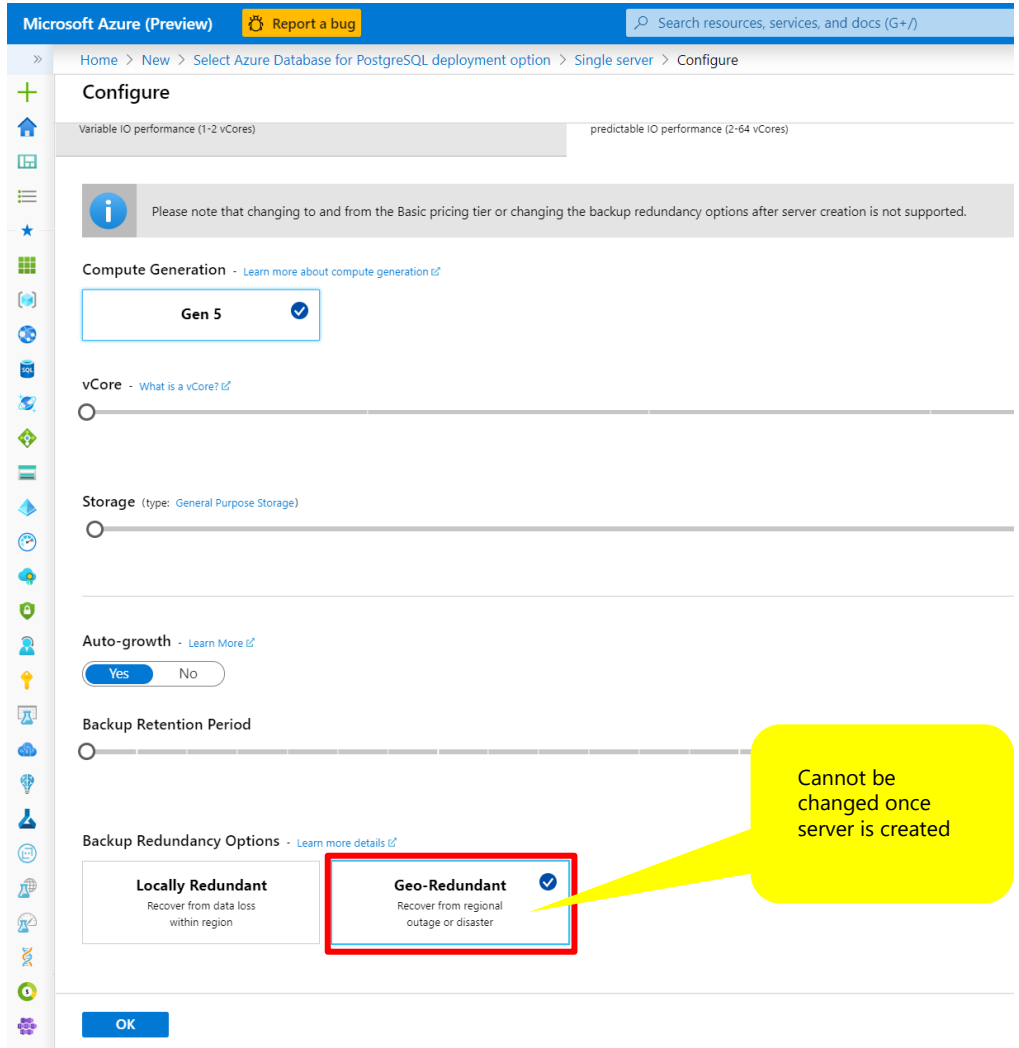


demodb_3

Single server backup and restore using “Geo Redundant” backup option

Pick the locally redundant server option





Restore the server to a point-in-time

- Restore the server in the same resource group as the backup
  - `az postgres server georestore --resource-group demo --name mypostgresserver-georestored --source-server mypostgresserver --location eastus`
- Restore the server in a different resource group from the backup
  - `az postgres server georestore --resource-group newresourcegroup --name mypostgresserver-georestored --source-server "/subscriptions/$<subscription ID>/resourceGroups/$<resource group ID>/providers/Microsoft.DBforPostgreSQL/servers/mypostgresserver" --location eastus`

Microsoft

Q&A

Connecting from a terminal