

RDS

1) APARTADO A

- 1) Crea una instancia (servidor-xyz y sustituye xyz por tus iniciales) de una base de datos de tipo MariaDB o MySQL conforme prefieras dentro de AWS RDS.

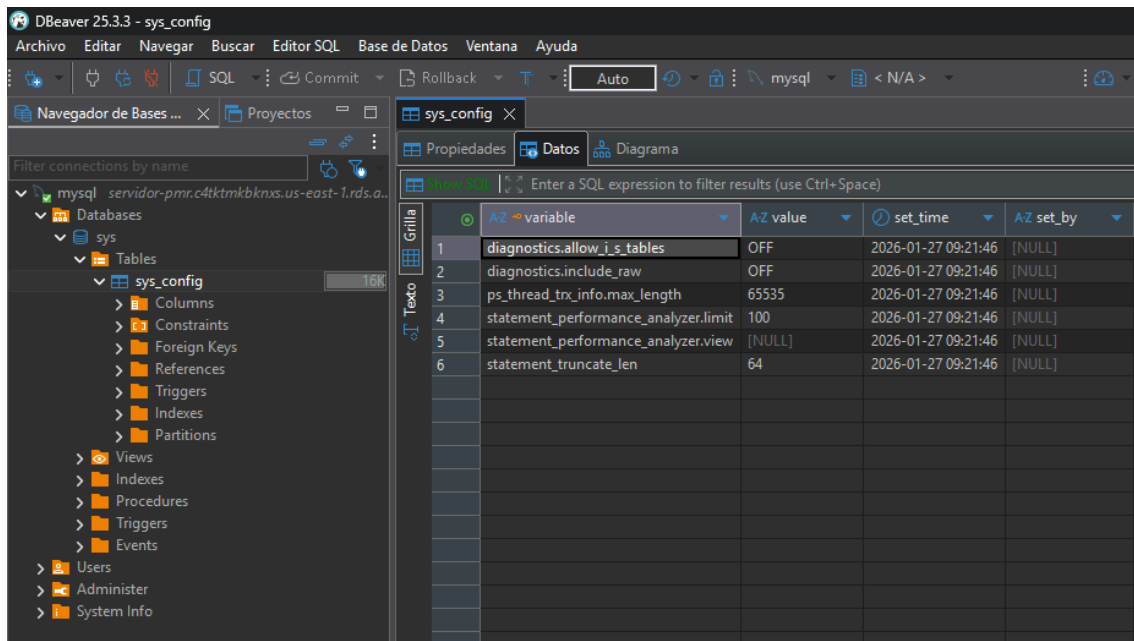
Creación de base de datos servidor-pmr

Es posible que el lanzamiento de la base de datos tarde unos minutos. Puede utilizar la configuración de servidor-pmr para mientras terminamos de crear su base de datos.

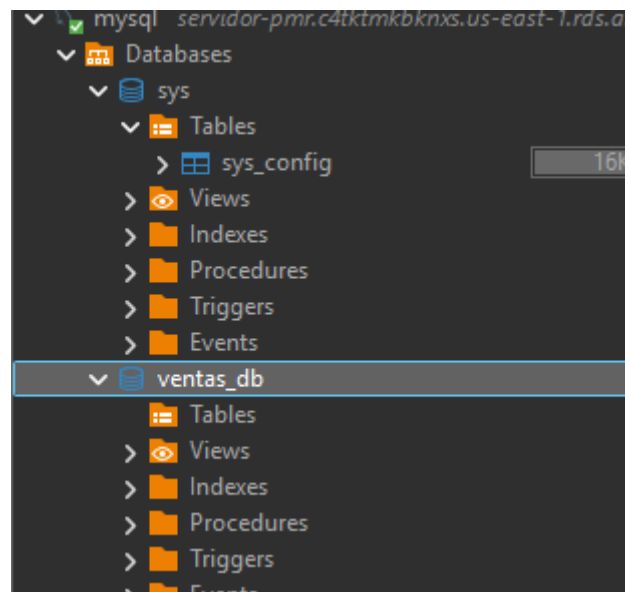
Bases de datos (2)

<input type="text"/> Filtrar por bases de datos				
Identificador de base de datos	Estado	Rol	Motor	Orden de
instituto	Backin...	Instancia	MariaDB	SECOND
servidor-pmr	Creando	Instancia	MySQL Co...	SECOND

- 2) Conéctate a la instancia RDS (necesitas que el acceso sea público y abrir el puerto necesario en el grupo de seguridad) bien sea mediante DBeaver o bien desde un terminal y crea en ella una base de datos llamada ventas_db.



Creamos la data base



- 3) Cárgala con los datos del script create_db.sql que se adjunta. Contiene información sobre un comercio (clientes, productos, pedidos, etc.).

<mysql> create_db

ventas_db

Propiedades

Diagrama

Database Name:

ventas_db

Default Collation:

utf8n

Default Charset:

utf8mb4

SQL Path:

Tables

Views

Indexes

Procedures

Triggers

Events

Source

Table Name	Engine	Auto Increment	Data Length	Partitioned	Description
categories	InnoDB	59	16K	[]	
customers	InnoDB	12.436	1,5M	[]	
departments	InnoDB	8	16K	[]	
order_items	InnoDB	172.199	7,5M	[]	
orders	InnoDB	68.884	3,5M	[]	
products	InnoDB	1.346	240K	[]	

2) APARTADO B

Realiza las siguientes consultas:

- 1) ¿De qué años tenemos datos de ventas?

```
SELECT year(o.order_date) FROM orders AS o
GROUP BY year(o.order_date);
```

resultados 1		Estadísticas 1	
Enter a SQL expression to filter results (use Ctrl+ Space)			
123	year(o.order_date)		
1		2.013	
2		2.014	

2) Ventas totales por categoría en un año (elige tú el año) de los existentes.

```
SELECT c.category_name, count(o.order_id) AS 'Total' FROM categories AS c
LEFT JOIN products AS p ON c.category_id = p.product_category_id
LEFT JOIN order_items AS oi ON oi.order_item_id = p.product_id
LEFT JOIN orders AS o ON oi.order_item_order_id = o.order_id
WHERE year(o.order_date) = '2013'
GROUP BY c.category_name;
```

AZ category_name	123 Total
Soccer	24
Baseball & Softball	24
Basketball	24
Lacrosse	24
Tennis & Racquet	24
Hockey	24
More Sports	24
Cardio Equipment	24
Strength Training	24

3) Clientes con más de 5 órdenes completas para ese año

```
SELECT c.customer_lname, count(o.order_id) as 'Total' FROM customers AS c
LEFT JOIN orders AS o ON o.order_customer_id = c.customer_id
WHERE year(o.order_date) = '2013'
GROUP BY c.customer_id
HAVING count(o.order_id) >= 5;
```

AZ customer_lname	123 Total
Rodriguez	5
Smith	6
James	5
Smith	8
Smith	5
Smith	5
Hayes	7
Smith	5
Butler	5
Smith	5
Wright	6

4) Los 3 productos más vendidos por departamento

```
SELECT department_name, product_id, product_name, cantidad_total
FROM (
  SELECT
    d.department_name,
    p.product_id,
    p.product_name,
    SUM(oi.order_item_quantity) AS cantidad_total,
    ROW_NUMBER() OVER (
      PARTITION BY d.department_id
      ORDER BY SUM(oi.order_item_quantity) DESC
    ) AS rn
  FROM departments d
  JOIN categories c
    ON c.category_department_id = d.department_id
  JOIN products p
    ON p.product_category_id = c.category_id
  JOIN order_items oi
    ON oi.order_item_product_id = p.product_id
```

```

GROUP BY
    d.department_id,
    d.department_name,
    p.product_id,
    p.product_name
) AS productos_rn
WHERE rn <= 3
ORDER BY department_name, cantidad_total DESC;

```

	AZ department_name	123 product_id	AZ product_name	123 cantidad_total
1	Apparel	365	Perfect Fitness Perfect Rip Deck	73.698
2	Apparel	403	Nike Men's CJ Elite 2 TD Football Cleat	22.246
3	Apparel	364	Total Gym 1400	36
4	Fan Shop	1.014	O'Brien Men's Neoprene Life Vest	57.803
5	Fan Shop	1.004	Field & Stream Sportsman 16 Gun Fire Safe	17.325
6	Fan Shop	1.073	Pelican Sunstream 100 Kayak	15.500
7	Fitness	116	Nike Men's Comfort 2 Slide	991
8	Fitness	44	adidas Men's F10 Messi TRX FG Soccer Cleat	939
9	Fitness	135	Nike Dri-FIT Crew Sock 6 Pack	883
10	Footwear	191	Nike Men's Free 5.0+ Running Shoe	36.680
11	Footwear	276	Under Armour Women's Ignite Slide	919
12	Footwear	172	Nike Women's Tempo Shorts	907
13	Golf	502	Nike Men's Dri-FIT Victory Golf Polo	62.956
14	Golf	627	Under Armour Girls' Toddler Spine Surge Runni	31.735
15	Golf	565	adidas Youth Germany Black/Red Away Match Soc	969
16	Outdoors	885	Team Golf St. Louis Cardinals Putter Grip	958
17	Outdoors	728	LJJA Women's Eyelet Sleeveless Golf Polo	946
18	Outdoors	804	Glove It Women's Imperial Golf Glove	941

5) Productos que nunca se han vendido

```

SELECT p.product_name, count(o.order_id) AS cuenta FROM products AS p
LEFT JOIN order_items AS oi ON oi.order_item_product_id = p.product_id
LEFT JOIN orders AS o ON oi.order_item_order_id = o.order_id
GROUP BY p.product_id
HAVING cuenta = 0;

```

	AZ product_name	123 cuenta
1	Quest Q64 10 FT. x 10 FT. SI	0
2	Under Armour Men's Highl	0
3	Under Armour Men's Reneç	0
4	Under Armour Men's Reneç	0
5	Riddell Youth Revolution Sp	0
6	Jordan Men's VI Retro TD Fi	0
7	Schutt Youth Recruit Hybri	0
8	Nike Men's Vapor Carbon E	0
9	Nike Adult Vapor Jet 3.0 Re	0
10	Under Armour Men's Highl	0
11	Fitness Gear 300 lb Olympic	0
12	Under Armour Men's Highl	0
13	Under Armour Men's Reneç	0
14	Quik Shade Summit SX170	0
15	Under Armour Kids' Highlig	0
16	Riddell Youth 360 Custom f	0
17	Under Armour Men's Highl	0
18	Reebok Men's Full Zip Trair	0

6) Promedio de ventas por categoría por año

```

SELECT
    c.category_id,
    c.category_name,
    YEAR(o.order_date) AS anyo,
    AVG(oi.order_item_subtotal) AS promedio
FROM orders o
JOIN order_items oi
    ON oi.order_item_order_id = o.order_id
JOIN products p
    ON p.product_id = oi.order_item_product_id
JOIN categories c
    ON c.category_id = p.product_category_id
GROUP BY
    c.category_id,
    c.category_name,
    YEAR(o.order_date)
ORDER BY
    anyo,
    c.category_name;

```

	123 category_id	A-Z category_name	123 anyo	123 promedio
1	40	Accessories	2.013	74,2396689782
2	16	As Seen on TV!	2.013	299,9700000904
3	3	Baseball & Softball	2.013	145,3325220149
4	4	Basketball	2.013	349,9899902344
5	12	Boxing & MMA	2.013	198,002166748
6	43	Camping & Hiking	2.013	299,9800109863
7	9	Cardio Equipment	2.013	294,1980860814
8	17	Cleats	2.013	180,1253641504
9	13	Electronics	2.013	101,9866304985
10	37	Electronics	2.013	127,4403408372
11	45	Fishing	2.013	399,9800109863
12	11	Fitness Accessories	2.013	120,6308569908
13	26	Girls' Apparel	2.013	127,4233885732
14	32	Golf Apparel	2.013	74,6164433489
15	34	Golf Bags & Carts	2.013	169,9900054932
16	36	Golf Balls	2.013	53,3252967923
17	35	Golf Gloves	2.013	108,6761786264
18	33	Golf Shoes	2.013	209,9273504274
19	7	Hockey	2.013	81,1893554688
20	44	Hunting & Shooting	2.013	121,5814502525
21	46	Indoor/Outdoor Games	2.013	150,8967769227
22	38	Kids' Golf Clubs	2.013	257,8459335819
23	5	Lacrosse	2.013	119,4906160146
24	18	Men's Footwear	2.013	129,9900054932
25	30	Men's Golf Clubs	2.013	158,9460020752

3) APARTADO C

- 1) Crea una instancia de bases de datos llamada servidor2-xyz, pero en esta ocasión utilizando únicamente comandos de AWS CLI.

Grupo de seguridad:

```
C:\Users\Mañana>aws ec2 create-security-group --group-name rds-public --description "Para servidor2-pmr"
{
  "GroupId": "sg-04aafb574b3528c6f",
  "SecurityGroupArn": "arn:aws:ec2:us-east-1:017319017365:security-group/sg-04aafb574b3528c6f"
}
```

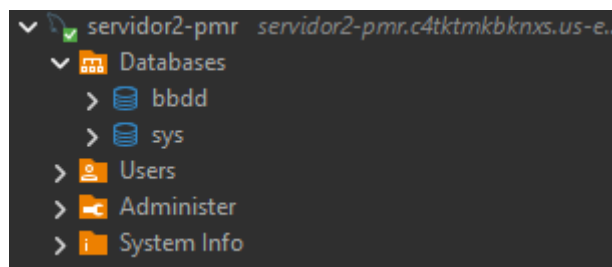
Instancia:

```
aws rds create-db-instance --db-instance-identifier servidor2-pmr --db-
instance-class db.t4g.micro --engine mysql --master-username admin --master-
user-password 12345678 --publicly-accessible --allocated-storage 20 --vpc-
security-group-ids sg-04aafb574b3528c6f --db-name bbdd --port 3306 --region
us-east-1
```

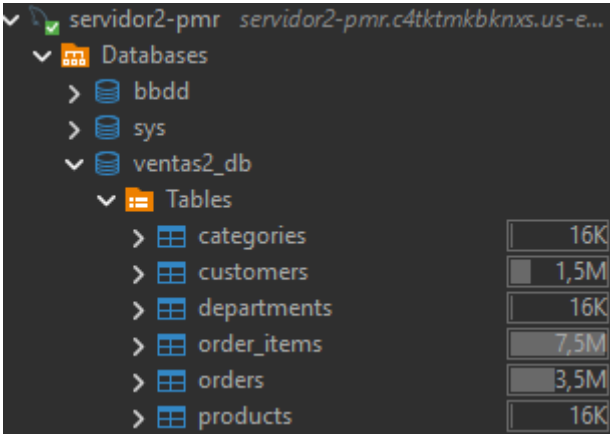
```
C:\Users\Mañana>aws rds create-db-instance --db-instance-identifier servidor2-pmr
--allocated-storage 20 --vpc-security-group-ids sg-04aafb574b3528c6f --db-name bbdd
{
  "DBInstance": {
    "DBInstanceIdentifier": "servidor2-pmr",
    "DBInstanceClass": "db.t4g.micro",
    "Engine": "mysql",
    "DBInstanceStatus": "creating",
    "MasterUsername": "admin",
    "DBName": "bbdd",
    "AllocatedStorage": 20,
    "PreferredBackupWindow": "07:51-08:21",
    "BackupRetentionPeriod": 1,
    "DBSecurityGroups": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-04aafb574b3528c6f",
        "Status": "active"
      }
    ],
    "DBParameterGroups": [
      {
        "DBParameterGroupName": "default.mysql8.0",
        "ParameterApplyStatus": "in-sync"
      }
    ],
    "DBSubnetGroup": {
      "DBSubnetGroupName": "default",
      "DBSubnetGroupDescription": "default",
      "VpcId": "vpc-0505ac2ec259b71c6",
      "SubnetGroupStatus": "Complete",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-0177f8d8cc2003f82",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1f"
          },
          "SubnetOutpost": {},
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-0d0ac8ca3bfcba18c",
          "SubnetAvailabilityZone": {

```

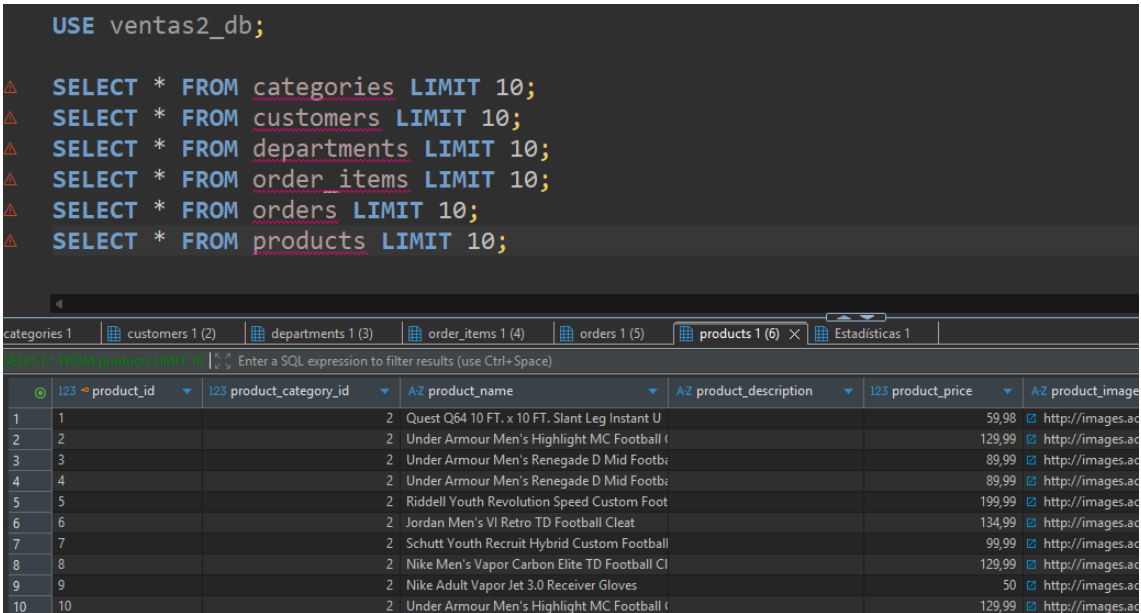
- 2) Conéctate a ella desde el cliente de terminal de MySQL que tienes instalado en la máquina EC2 con Ubuntu y MySQL que creamos en una práctica anterior.



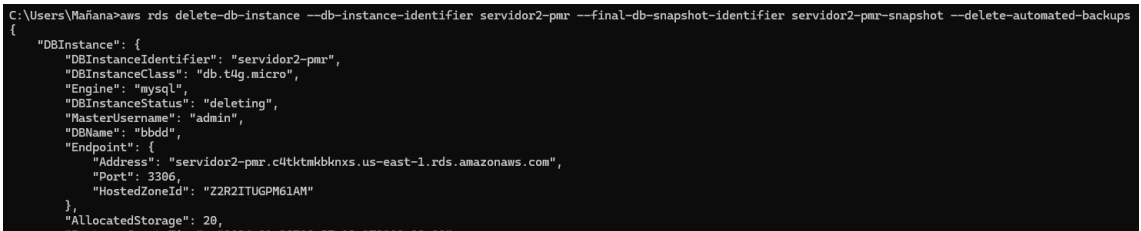
- 3) Desde el terminal anterior crea una base de datos ventas2_db y ejecuta el script create_db.sql del apartado A para poblarla de datos.



4) Comprueba con las consultas adecuadas que se crearon las tablas y se añadieron los registros.



5) Investiga con que comando de AWS CLI podemos borrar la instancia.



4) APARTADO D

- 1) A partir de esta instancia, crea una instantánea de forma manual que en el nombre contenga tus iniciales. Adjunta capturas de pantalla donde se vea la instancia creada, así como las tablas existentes (ya sea mediante el cliente de MySQL o DBeaver) y la instantánea.

Tomar instantánea de la base de datos

Preferencias

Para tomar una instantánea de base de datos, elija una base de datos y asigne un nombre a la instantánea de base de datos.

Tipo de instantánea

- ☒ Instancia de base de datos
☐ Clúster de base de datos

Instancia de base de datos

Identificador de instancia de base de datos. Es la clave única que identifica una instancia de base de datos.

servidor2-pmr

Nombre de instantánea

Identificador de la instantánea de la base de datos.

servidor2-pmr-instantanea

El identificador de la instantánea no distingue entre mayúsculas y minúsculas, pero se almacena todo en minúsculas, como en "mysnapshot". No puede ser nulo, vacío o estar en blanco. Debe contener de 1 a 255 caracteres alfanuméricos o guiones. El primer carácter debe ser una letra. No puede terminar con un guión ni contener dos guiones consecutivos.

Cancelar

Realizar instantánea

Tablas

ventas2_db						
Propiedades Diagrama						
Database Name:		ventas2_db			Default Collation:	
Default Charset:		utf8mb4			SQL Path:	
Tables	Table Name	Engine	Auto Increment	Data Length	Partitioned	Descript
Views	categories	InnoDB	59	16K	[]	
Indexes	customers	InnoDB	12.436	1,5M	[]	
Procedures	departments	InnoDB	8	16K	[]	
Triggers	order_items	InnoDB	172.199	7,5M	[]	
Events	orders	InnoDB	68.884	3,5M	[]	
Source	products	InnoDB	1.346	16K	[]	

- 2) A partir de la instantánea recién creada, restáurala en una nueva instancia (por ejemplo, servidor3-xyz) de tipo db.t4g.small, y tras conectarte mediante DBeaver, comprueba que tiene los datos ya cargados. Adjunta una captura de pantalla donde se vean las características de las dos instancias.

Se está restaurando la instantánea						
Restaurando servidor2-pmr-instantanea a servidor2-pmr-instantanea . Esto puede tardar varios minutos en completarse.						
ases de datos (3)						
Filtrar por bases de datos						
	Identificador de base de datos	Estado	Rol	Motor	Orden de implementa...	Región
<input type="radio"/>	servidor-pmr	✓ Dispon...	Instancia	MySQL Co...	SECOND	us-east
<input type="radio"/>	servidor2-pmr	✓ Dispon...	Instancia	MySQL Co...	SECOND	us-east
<input checked="" type="radio"/>	servidor2-pmr-instantanea	⌚ Creando	Instancia	MySQL Co...	SECOND	-

Servidor2-pmr:

RDS

servidor2-pmr

Modificar Acciones

Resumen

Identificador de base de datos
servidor2-pmr

CPU
5,52%

Estado
Disponible

Clase
db.t4g.micro

Rol
Instancia

Actividad actual
2 Conexiones

Motor
MySQL Community

Región y AZ
us-east-1f

Recomendaciones

Servidor2-pmr-instantanea:

servidor2-pmr-instantanea

Modificar Acciones

Resumen

Identificador de base de datos
servidor2-pmr-instantanea

CPU
5,44%

Estado
Disponible

Clase
db.t4g.micro

Rol
Instancia

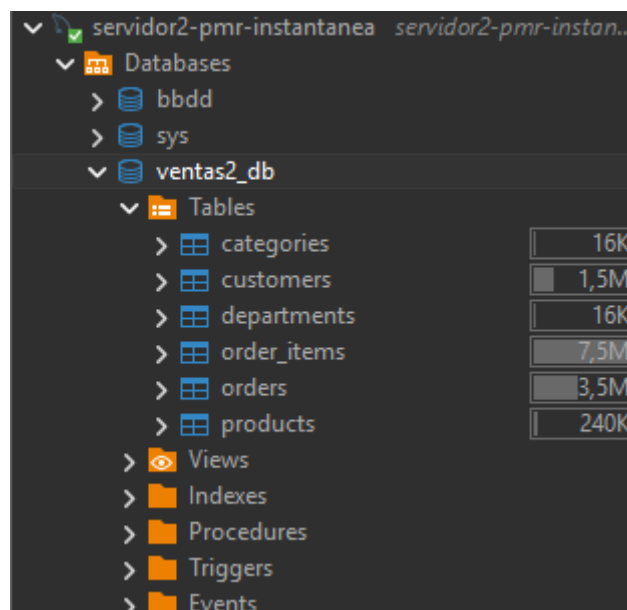
Actividad actual
0 Conexiones

Motor
MySQL Community

Región y AZ
us-east-1d

Recomendaciones

Tiene datos:



- 3) Finalmente, elimina ambas instancias para evitar gastos innecesarios en el laboratorio.

Bases de datos (3)

Filtrar por bases de datos

Identificador de base de datos	Estado
servidor-pmr	Disponible
servidor2-pmr	Eliminando
servidor2-pmr-instantanea	Eliminando