

# Laboratorio de desafíos: migración de datos a las bases de datos de Amazon RDS


En este laboratorio de desafíos vamos migrar los datos de una base de datos alojada en una instancia de EC2 en una base de datos relacional RDS.


## Tarea 1: Crear la base de datos relacional RDS


Primero que nada necesitaremos crear una base de datos en la que migraremos los datos en un futuro. La base de datos necesitará las siguientes características:


Opciones del motor

Tipo de motor [Información](#)

☐ Aurora (MySQL Compatible)  


☐ Aurora (PostgreSQL Compatible)  


☐ MySQL  


☒ MariaDB  


Plantillas

Elija una plantilla de ejemplo para adaptarla a su caso de uso.

☐ Producción  
Utilice los valores predeterminados para disfrutar de una alta disponibilidad y de un rendimiento rápido y constante.

☒ Desarrollo y pruebas  
Esta instancia se ha diseñado para su uso en desarrollo, fuera de un entorno de producción.

☐ Capa gratuita  
Utilice el nivel gratuito de RDS para desarrollar nuevas aplicaciones, probar aplicaciones existentes o adquirir experiencia práctica con Amazon RDS. [Información](#)

Identificador de instancias de bases de datos [Información](#)

Escriba un nombre para la instancia de base de datos. El nombre del pertenecientes a su cuenta de AWS en la región de AWS actual.

CafeDatabase

Nombre de usuario maestro

Escriba un ID de inicio de sesión pa

admin

1 a 16 caracteres alfanuméricos. El

password: Caf3DbPassw0rd!

- ☐ Clases estándar (incluye clases m)
- ☐ Clases optimizadas para memoria (incluye clases r y
- ☒ Clases con ráfagas (incluye clases t)

db.t3.micro

2 vCPUs 1 GiB RAM Red: 2085 Mbps

## Disponibilidad y durabilidad

### Implementación Multi-AZ [Información](#)

- ☐ Crear una instancia en espera (recomer  
Crea una instancia en espera en otra zona de  
minimizar los picos de latencia durante las co
- ☒ No crear una instancia en espera

### Grupo de subredes de la base de datos [Inf](#)

Elija el grupo de subred de DB. El grupo de subred  
seleccionada.

lab-db-subnet-group

2 Subredes, 2 Zonas de disponibilidad

### Zona de disponibilidad [I](#)

us-east-1a

### Tipo de almacenamiento [Información](#)

Los volúmenes de almacenamiento SSD de IOPS aprovis

SSD de uso general (gp3)

El rendimiento se escala independientemente del alm;

### Almacenamiento asignado [Información](#)

20

Mínimo: 20 GiB Máximo: 5144 GiB

### Nube privada virtual (VPC) [Información](#)

Elija la VPC. La VPC define el entorno de red virtual

Lab VPC (vpc-0136dc0188304e1a1)

3 Subredes, 2 Zonas de disponibilidad

Cada vez que crea la VPC con un grupo de subredes de

### Grupos de seguridad de VPC existentes

Elegir una o más opciones

dbSG X

### ▼ Configuración adicional

### Puerto de la base de datos [Infor](#)

Puerto TCP/IP que la base de datos usa

3306

## Supervisión

☐ Activar la monitorización mejorada

Activar las métricas de monitorización mejo

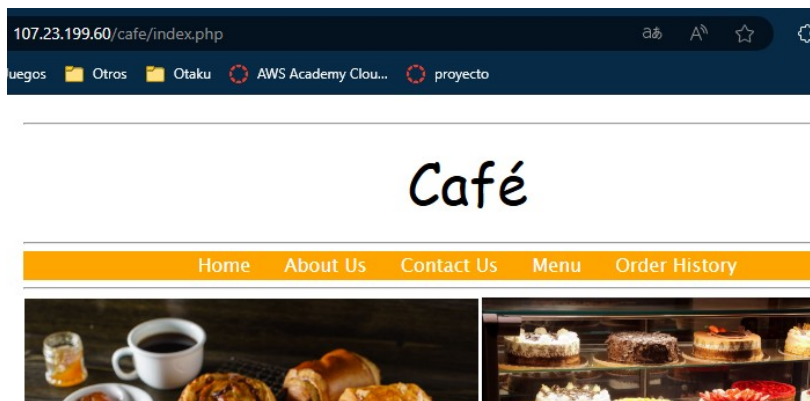
El resto lo dejamos como predeterminado y le damos a crear base de datos.(Tened en cuenta que esta configuración está hecha de esta forma debido a la instancia ya creada y al esquema de subredes, en nuestro proyecto podemos/debemos hacerlo diferente).

# Tarea 2: analizar la implementación de la aplicación existente de la cafetería

Para llevar a cabo esta tarea, necesitaremos conectarnos a la IP de la instancia EC2, la cual se encuentra en los detalles de la instancia **CafeServer**:



Una vez conocida, la ponemos en el navegador y le agregamos /cafe.



Vamos a probar que la app funciona haciendo un pedido en la pestaña “menu”:

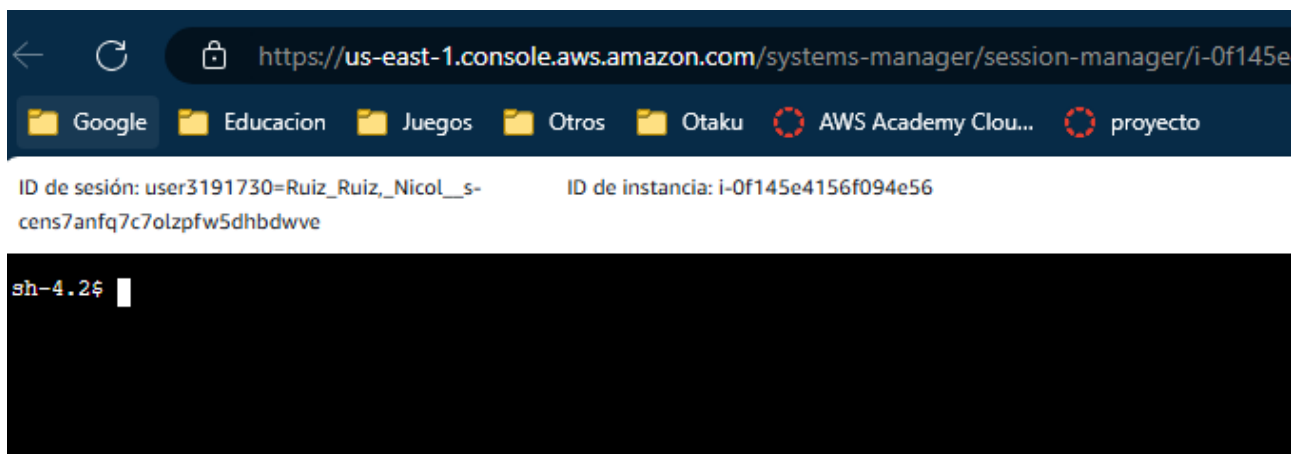
Thank for your order! It will be available for pickup within 15 minutes. Your order number and details are shown below.

Order Number: 25	Date: 2024-04-16	Time: 09:29:53	Total Amount: \$64.50
Item	Price	Quantity	Amount
Croissant	\$1.50	10	\$15.00
Donut	\$1.00	12	\$12.00
Chocolate Chip Cookie	\$2.50	15	\$37.50

© 2020, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Vemos que podemos pedir perfectamente, y estos valores se guardan en el historial de pedidos de la propia página. Esta base de datos es la que vamos a migrar a la RDS.

Vamos a conectarnos a la instancia EC2 utilizando Session Manager, para ellos abrimos el servicio en la consola AWS, elegimos nuestra instancia EC2 y le damos a conectar.



Debería verse tal que así, esto es literalmente una consola de comandos de linux abierta desde el navegador y sin entorno gráfico. Una vez dentro ponemos los siguientes comandos para iniciar sesión como el usuario de la instancia:

```
sh-4.2$ bash
[ssm-user@cafeserver bin]$ sudo su
[root@cafeserver bin]# su ec2-user
[ec2-user@cafeserver bin]$ whoami
ec2-user
[ec2-user@cafeserver bin]$ cd
[ec2-user@cafeserver ~]$ cd
[ec2-user@cafeserver ~]$ pwd
/home/ec2-user
[ec2-user@cafeserver ~]$
```

## Tarea 3: trabajar con la base de datos en la instancia de EC2

Desde esta terminal, podemos apreciar que la instancia tiene una base de datos en la que trabaja localmente:

```
[ec2-user@cafeserver ~]$ service mariadb status
Redirecting to /bin/systemctl status mariadb.service
● mariadb.service - MariaDB 10.2 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; vendor preset: enabled)
   Drop-In: /usr/lib/systemd/system/mariadb.service.d; want: mariadb.service
          └─tokudb.conf
   Active: active (running) since Sat 2024-05-25 12:17:27 CEST; 1min 17s ago
   Process: 2717 ExecStartPost=/usr/libexec/mysqld
```

Vamos a volver a la consola AWS donde abrimos **Systems Manager** y a la izquierda vamos a darle a Parameter Store. Vemos y copiamos la contraseña de la base de datos: Re:Start!9

Volvemos a la terminal e iniciamos sesión en la base de datos con los datos que podemos sacar aquí:

```
[ec2-user@cafeserver ~]$ mysql --version
mysql Ver 15.1 Distrib 10.2.38-MariaDB, for Linux (x86_64)
[ec2-user@cafeserver ~]$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g
Your MariaDB connection id is 15
Server version: 10.2.38-MariaDB MariaDB Server
```

Estando aquí dentro vamos a revisar el pedido que hicimos anteriormente, el cual se almacena en la tabla order:

```
22 | 2024-04-16 09:29:53 | 33.50 |
23 | 2024-04-16 09:29:53 | 6.00 |
24 | 2024-04-16 09:29:53 | 35.00 |
25 | 2024-04-16 09:29:53 | 64.50 |
+-----+-----+-----+
```

Aquí podemos ver los pedidos que hicimos(select \* from `order`;) Salgamos de aquí escribiendo exit o pulsando ctrl+c y hacemos un dump de la base de datos(local):

```
[ec2-user@cafeserver ~]$ mysqldump --databases cafe_db -u root -p > CafeDbDump.sql
Enter password:
[ec2-user@cafeserver ~]$ ls
CafeDbDump.sql  set-key.sh  setup  www
[ec2-user@cafeserver ~]$
```

Hay lo tenemos. Vamos a responder una cuantas preguntas antes de seguir avanzando(para responder, debemos analizar nuestra base de datos RDS):

¿Dónde se ejecuta la instancia de RDS?

Se ejecuta en el nivel de zona de disponibilidad

Región y AZ  
us-east-1a

¿La instancia de RDS tiene asignada una dirección IP pública IPv4?

Tipo de red  
IPv4

No, no la tiene asignada.

¿Cuál es la etiqueta **Name** (Nombre) aplicada a la subred en la que se ejecuta la instancia de RDS?

Abrimos los 2 enlaces de las subredes y podemos apreciar que la private subnet 1 está en la misma zona de disponibilidad que la RDS

<input type="checkbox"/>	Name	ID d...	Estado	VPC	CIDR...	CIDR...	Direc...	Zona de disponibilidad
<input type="checkbox"/>	Private Subnet 1	<a href="#">subnet-...</a>	Available..	<a href="#">vpc-013...</a>	10.0.2.0...	-	506	us-east-1a

¿Cuántas reglas del grupo de seguridad se definen para la instancia de RDS?

Entramos en el grupo de seguridad y vemos que solo tiene una regla de salida.

Buscar

<

1

>

<input type="checkbox"/>	Name	ID de la regla del gr...	Versión de IP	Tipo	Protocolo	Intervalo de pu
<input type="checkbox"/>	-	sgr-0416fd06c9c352bc8	IPv4	Todo el tráfico	Todo	Todo

Una vez respondidas estas preguntas podemos continuar. Vamos a configurar una conexión de red desde la terminal a una nueva instancia en RDS. Pero primero que nada debemos permitir el acceso a través del puerto 3306, para ello editamos los grupos de seguridad de la RDS y añadimos esta regla de entrada:

sgr-097f6518654f0dc21

MySQL/Aurora

TCP

3306

Person...

Eliminar

sg-02ebbb240b283672f9

X

El origen es el grupo de seguridad de la instancia EC2. Ahora podremos iniciar sesión en la base de datos desde la terminal:

```
[ec2-user@cafeserver ~]$ mysql -u admin -p --host cafedatabase.cjxjnzebacc5.us-east-1.rds.amazonaws.com
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 72
Server version: 10.11.6-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

En esta base de datos aún no hay nada ya que no hemos importado los datos todavía, vamos a hacerlo.

# Tarea 5: importar los datos a la instancia de base de datos de RDS

Para ello usaremos este comando:

```
mysql -u admin -p --host cafedatabase.cjxjnzebacc5.us-east-1.rds.amazonaws.com < CafeDbDump.sql
```

Ahora mismo si volvemos a entrar, debemos ver la base de datos ya creada y los registros en las tablas:

```
Database changed
MariaDB [cafe_db]> show tables
-> ;
+-----+
| Tables_in_cafe_db |
+-----+
| order              |
| order_item         |
| product            |
| product_group      |
+-----+
4 rows in set (0.00 sec)

MariaDB [cafe_db]> select * from `order`;
+-----+-----+-----+
| order_number | order_date_time | amount |
+-----+-----+-----+
| 1           | 2024-04-16 09:29:53 | 20.00 |
| 2           | 2024-04-16 09:29:53 | 24.00 |
+-----+-----+-----+
```

En este punto ya tenemos lista la base de datos RDS, es hora de conectarla con la aplicación para que los registros se guarden en ella.

# Tarea 6: conectar la aplicación de la cafetería a la nueva base de datos

Para ello vamos a abrir y editar el almacén de parámetros, y vamos a poner que la url de la base de datos es el punto de acceso de esta misma:

Nombre

/cafe/dbUrl

ARN

arn:aws:ssm:us-east-1:840946746128:parameter/cafe/dbUrl

Capa

Standard

Tipo

String

Valor

cafedatabase.cjxjnzebacc5.us-east-1.rds.amazonaws.com

También tenemos que cambiar el nombre de usuario y la contraseña:

Valor

admin

Tipo

String


Valor

Caf3DbPassw0rd!

Si lo hemos hecho bien, podremos ver el menú perfectamente:

HomeMenuOrder History

Pastries




Croissant

\$1.50

Fresh, buttery and fluffy... Simply delicious!

Quantity:




Donut

\$1.00

We have more than half-a-dozen flavors!

Quantity:



Chocolate Chip Cookie

\$2.50

Made with Swiss chocolate with a touch of Madagascar vanilla

Quantity:




Y para comprobar que la instancia EC2 ya no usa esa base de datos, vamos a parar el servicio y volver a la página:

```
[ec2-user@cafeserver ~]$ sudo service mariadb stop
Redirecting to /bin/systemctl stop mariadb.service
[ec2-user@cafeserver ~]$
```


Sigue funcionando perfectamente:

[Home](#) [Menu](#) [Order History](#)


Pastries



**Croissant**  
\$1.50



**Donut**  
\$1.00



**Chocolate Chip Cookie**  
\$2.50