

Ejecutar Redis Server

Redis se instaló en la máquina virtual siguiendo la secuencia indicada en la [página oficial](#).

Se debe abrir una terminal y ejecutar el siguiente comando para levantar el *server* Redis:

```
$ src/redis-server
```

Una vez el servicio de Redis esté disponible, existen dos alternativas para hacer consultas:

Redis CLI

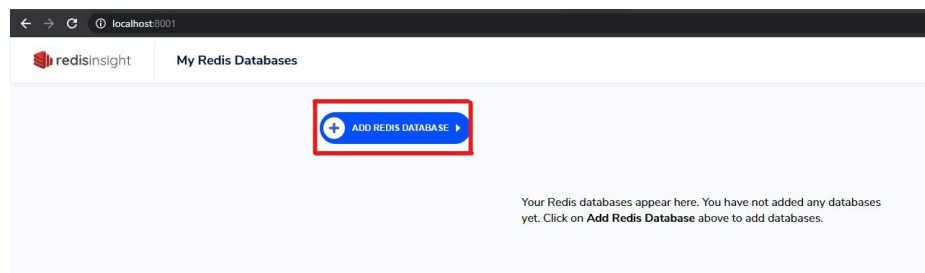
Para iniciar se debe abrir una terminal distinta a la del *server* y ejecutar:

```
$ src/redis-cli
```

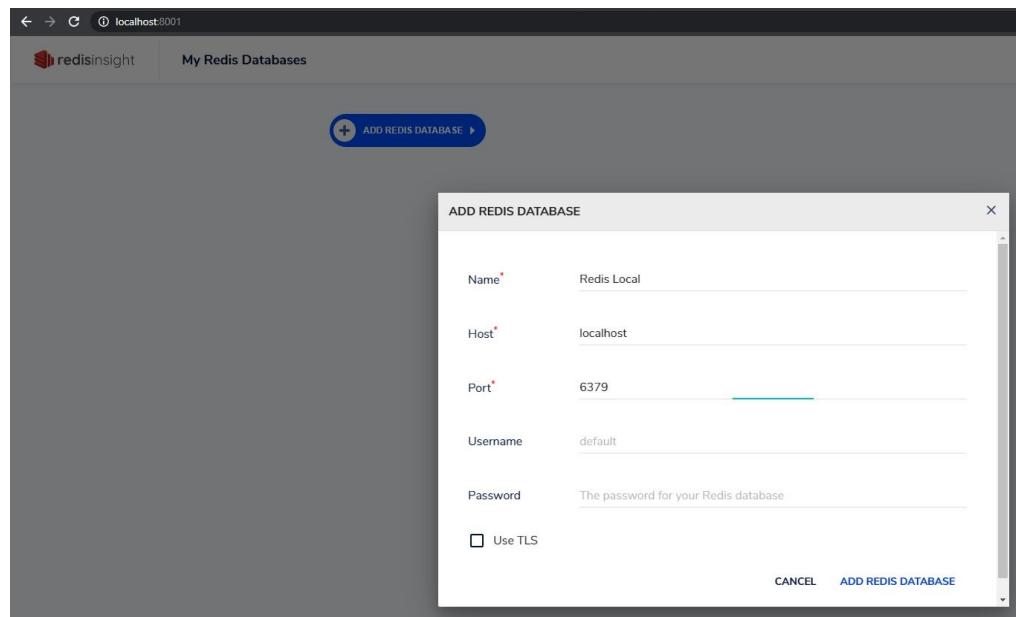
Luego de realizarse la conexión, ya se pueden realizar consultas a la base de datos.

Redis Insight

Una vez que se inicie el programa, se debe ingresar la información de conexión al Redis Server iniciado anteriormente:



Completar con los siguientes datos:

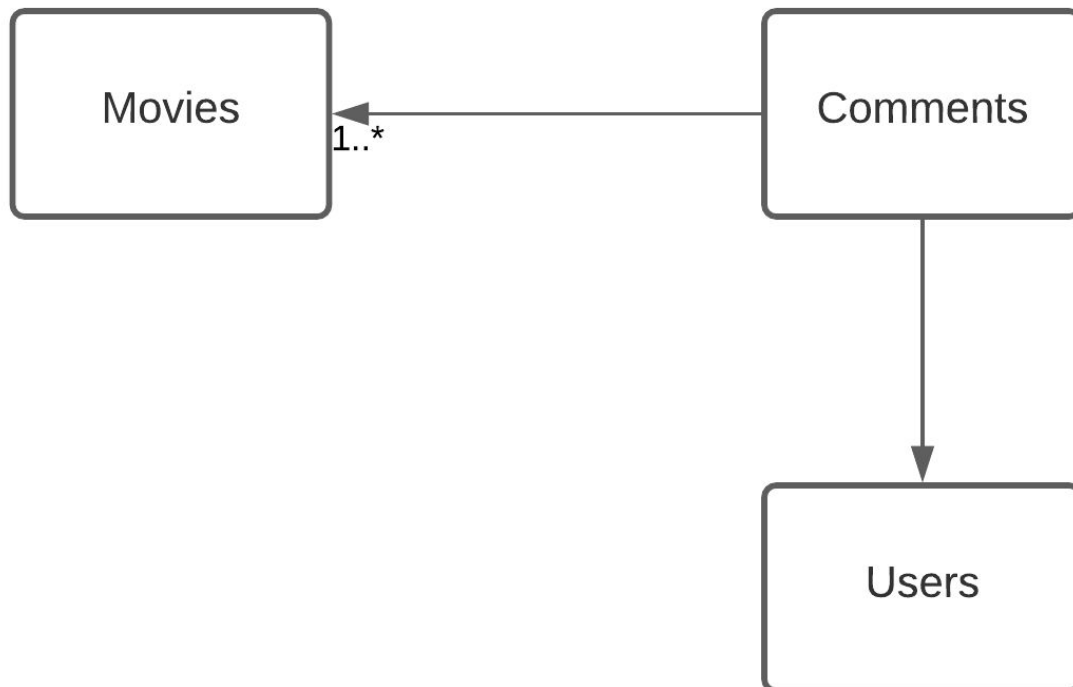




Contexto

Se deberá crear una plataforma que será utilizada como foro de reseñas de películas. El arquitecto de solución decidió que Redis sea un componente crucial para esta plataforma. Por lo tanto, creó los siguientes ejercicios para que el equipo se pusiera a tono con esta tecnología.

El modelo transaccional respeta el diagrama que se muestra a continuación:





Ejercicios

1. Crear tres claves que contengan la propiedad “name” en los siguientes usuarios:

- User 1 → Julia.
- User 2 → Pedro.
- User 3 → Jorge.

Consultar las claves creadas.

```
SET users:1:name Julia
SET users:2:name Pedro
SET users:3:name Jorge
GET users:1:name Julia
GET users:2:name Pedro
GET users:3:name Jorge
```

2. Si se desea crear un contador, ¿qué opciones son correctas?

- Opción A:

```
$ SET movie:1:views 1
OK
$ SET movie:1:views 2
OK
$ GET movie:1:views
"2"
```

- Opción B:

```
$ INCR movie:1:views
(integer) 1
$ INCR movie:1:views
(integer) 2
$ GET movie:1:views
"2"
```

- Opción C:

```
$ HINCRBY movie:1 views 1
(integer) 1
$ HINCRBY movie:1 views 1
(integer) 2
$ HGET movie:1 views
"2"
```

- Opción D:

```
$ INCR movie:1:views
(integer) 1
```



```
$ SET movie:1:views 2
OK
$ GET movie:1:views
"2"
```

- Opción E:

No implementar contadores en Redis.

Opción B y C.

3. Crear una entidad “movie” con las siguientes propiedades:

- id: 213
- name: The Matrix
- rating: 8
- country: USA

Consultar la película creada. ¿Qué otra alternativa hay para crear esta entidad?

```
HMSET movies:213 name "The Matrix" rating 8 country USA
HGETALL movies:213
SET movies:213:name "The Matrix"
SET movies:213:rating 8
SET movies:213:country USA
```

4. Crear una lista de comentarios para la película creada en el punto 3:

- User 1: “Muy buena película”.
- User 2: “No me convencieron los efectos especiales”.
- User 3: “La segunda parte es mejor”.

```
RPUSH movies:213:comments "Muy buena película" "No me convencieron los efectos especiales" "La segunda parte es mejor"
```

5. Obtener los primeros dos comentarios de la lista creada en el punto anterior.

```
LRANGE movies:213:comments 0 1
```

6. Escribir una consulta que obtenga la cantidad de comentarios que tiene una película con cierto ID. Se debe respetar la estructura creada en el punto 4.

```
LLEN movies:213:comments
```



7. Crear una clave que sea utilizada para almacenar la dirección de correo electrónico (sin repeticiones) de los usuarios para recomendarles películas.

```
SADD users:emails asd@example.com qwe@example.com zxc@example.com
```

8. Para la estructura creada en el punto anterior, consultar si una de las direcciones ingresadas está incluida.

```
SISMEMBER users:emails asd@example.com
```

9. Se necesita almacenar el título de las películas ordenadas por año de manera ascendente. ¿Qué estructura es la correcta para este requerimiento? Mostrar la operación de escritura que utilizará con al menos cinco películas de distintos años.

```
ZADD movies:ordered 2008 "Wall-E" 1995 "Toy Story" 2020 Soul 2005  
Cars 2017 Coco
```

10. Obtener la película más antigua siguiendo con la estructura del punto anterior.

```
ZRANGE movies:ordered 0 0
```

11. Al parecer no se obtuvieron los derechos del estreno más reciente, por ende se requiere que se arme una consulta que elimine dicha película de la estructura planteada en el punto 7.

```
ZPOPMAX movies:ordered
```

12. Registrar las visitas de usuarios para una determinada película, para el análisis en patrones de los datos (*analytics*).

El valor que se guardará en la clave será la IP del usuario concatenado al valor devuelto por el comando "TIME" para lograr registros de visitas únicas:

```
$ TIME  
1) "1604689091" --> este valor  
2) "417905"
```

Ejemplo: "186.23.28.11-1604689091".

Agregar diez visitas y consultar el estado de la clave.



```
PFADD movies:213:views 186.23.28.11-1604689091
186.23.25.12-1604689092 ...
(10 valores únicos)
```

13. Se requiere crear usuarios de manera aislada y atómica. La secuencia de pasos debe ser la siguiente:

- Crear un usuario con las propiedades:
 - id
 - name
 - age
 - birthday
 - email
- Registrar la dirección de correo electrónico del usuario con la estructura creada en el punto 6.
- Incrementar el contador de usuarios activos.

```
MULTI
HMSET users:1 name John age 32 birthday "20/11/1990" email
jdoe@example.com
SADD users:emails jdoe@example.com
INCR activeUsers
EXEC
```

14. Por cuestiones de seguridad, la CDN (*Content Delivery Network*), que se utiliza en la plataforma para guardar las *thumbnails* de las películas, entrega una contraseña temporal que tiene siete días de vencimiento. Crear una clave que contenga esta contraseña y que, además, cumpla con el requerimiento de los siete días de vencimiento.

```
SET cdn:password unaClaveMuySecreta EX 604800
SETEX cdn:password 604800 unaClaveMuySecreta
```

15. Determinar si existe la clave creada en el punto anterior y eliminarla.

```
EXISTS cdn:password
DEL cdn:password
```

