

ACTIVIDAD 1 - REDIS - CONFIGURACIÓN MAESTRO- ESCLAVO (MASTER-REPLICA)

Cristóbal Suárez Abad

ADMINISTRACIÓN DE SISTEMAS GESTORES DE BASES DE DATOS - 2º ASIR

Índice

MAESTRO-REPLICA.....	2
1) Levanta dos nodos Maestro-Esclavo con el docker-compose adjunto.....	2
2) Visualiza la configuración en cada caso.....	3
3) Instala Redis-Insight como cliente de Redis.....	4
4) Crea una conexión tanto para el Maestro como el esclavo.....	5
5) Verificación de Sincronización(Añade pantallazos y explicaciones):	6
6) Prueba de Integridad: Intenta escribir un dato directamente en el esclavo. ¿Qué ocurre? Explica por qué. (Añade pantallazos y explicaciones)	8
REPLICACIÓN EN CADENA.....	9
Haz una prueba de escritura en el Maestro(A).....	13
Verifica que el dato llega a la Réplica Intermedia (B).	14
Verifica que el dato llega a la Réplica Final (C).	15
Ejecuta INFO REPLICATION en todos los nodos, explica que ves y la diferencia entre nodos.	16

MAESTRO-REPLICA

- 1) Levanta dos nodos Maestro-Esclavo con el docker-compose adjunto.

```
C:\Users\UsuarioASIR\proyectos\sgbd_tema04_actividad_01>docker compose up -d
time="2025-12-02T10:01:12+01:00" level=warning msg="C:\\\\Users\\\\UsuarioASIR\\\\proyectos\\\\sgbd_tema04_actividad_01\\\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 10/10
  ✓ redis-master Pulled
    ✓ d75b3bec998 Pull complete
    ✓ 4f4fb700ef54 Pull complete
    ✓ 60c57c0072ef Pull complete
    ✓ f637881d1138 Pull complete
    ✓ 232f7549c9b0 Pull complete
    ✓ c70aae7b5e0d Pull complete
    ✓ fc4343b4accd Pull complete
    ✓ 380e8aa8b1fd Pull complete
  ✓ redis-replica Pulled
[+] Running 3/3
  ✓ Network sgbt_tema04_actividad_01_redis-net Created
  ✓ Container redis-master Started
  ✓ Container redis-replica Started
  8.3s
  3.9s
  0.2s
  3.9s
  5.0s
  3.6s
  5.2s
  5.0s
  5.2s
  0.3s
  0.1s
  0.8s
  0.9s
C:\Users\UsuarioASIR\proyectos\sgbd_tema04_actividad_01>
```

```
C:\Users\UsuarioASIR\proyectos\sgbd_tema04_actividad_01>docker ps
CONTAINER ID   IMAGE      COMMAND       CREATED     STATUS      PORTS          NAMES
c770229cc222   redis:7-alpine "docker-entrypoint.s..."  24 seconds ago Up 22 seconds  0.0.0.0:6380->6379/tcp, [::]:6380->6379/tcp   redis-replica
bde673c2254c   redis:7-alpine "docker-entrypoint.s..."  24 seconds ago Up 23 seconds  0.0.0.0:6379->6379/tcp, [::]:6379->6379/tcp   redis-master
```

2) Visualiza la configuración en cada caso.

```
version: '3.8'

services:
  # -----
  # ACTIVIDAD 1 y 3: EL NODO MAESTRO (MASTER)
  #
  redis-master:
    image: redis:7-alpine
    container_name: redis-master
    ports:
      - "6379:6379"
    networks:
      - redis-net
    command: redis-server --appendonly yes

  # -----
  # ACTIVIDAD 1: EL PRIMER ESCLAVO (REPLICACION DIRECTA)
  #
  redis-replica:
    image: redis:7-alpine
    container_name: redis-replica
    ports:
      - "6380:6379"
    networks:
      - redis-net
    depends_on:
      - redis-master
    command: redis-server --replicaof redis-master 6379

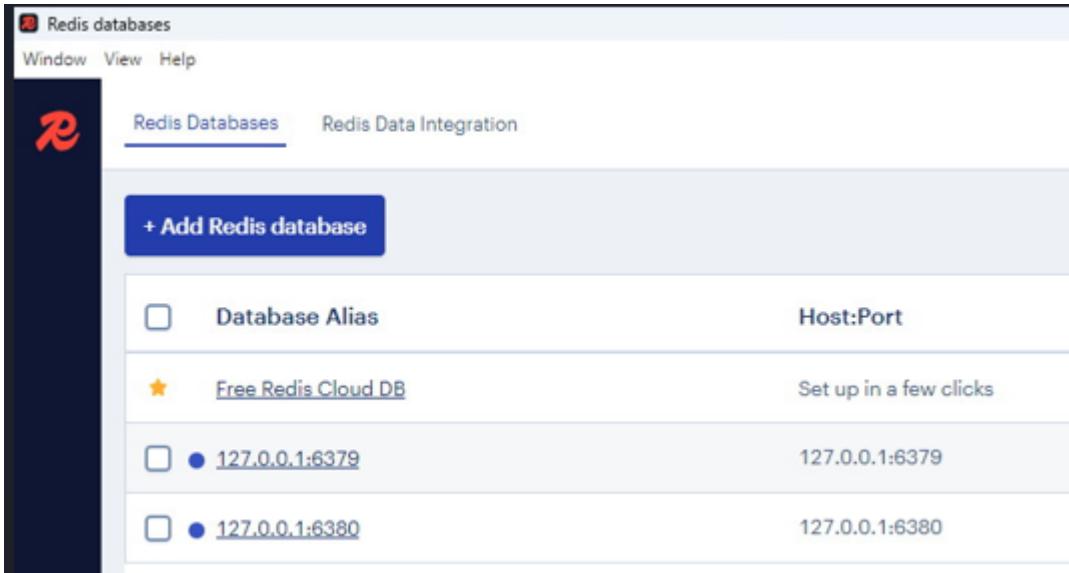
networks:
  redis-net:
    driver: bridge
```

3) Instala Redis-Insight como cliente de Redis.



4) Crea una conexión tanto para el Maestro como el esclavo

Al abrir Redis-Insight se configuran automáticamente las conexiones. Más adelante veremos como se hace una conexión.



The screenshot shows the Redis-Insight application window. At the top, there's a menu bar with 'Redis databases' (highlighted), 'Window', 'View', and 'Help'. Below the menu is a navigation bar with 'Redis Databases' (highlighted in blue) and 'Redis Data Integration'. On the left side, there's a dark sidebar with a red 'R' logo. In the main content area, there's a blue button labeled '+ Add Redis database'. Below it, there's a table with two columns: 'Database Alias' and 'Host:Port'. The first row shows a yellow star icon next to a link 'Free Redis Cloud DB' and the text 'Set up in a few clicks'. The second and third rows show entries for '127.0.0.1:6379' and '127.0.0.1:6380' respectively, both with a blue dot icon next to the host:port.

Database Alias	Host:Port
Free Redis Cloud DB	Set up in a few clicks
127.0.0.1:6379	127.0.0.1:6379
127.0.0.1:6380	127.0.0.1:6380

5) Verificación de Sincronización(Añade pantallazos y explicaciones):

- Conéctate al Maestro y crea la clave-valor de ejemplo

The screenshot shows the Redis Browser interface at the URL 127.0.0.1:6379. The left sidebar has icons for databases, keys, and monitoring. The main area shows a table of keys:

Total: 1	Last refresh: < 1 min	Columns
HASH plato	No limit	104 B

On the right, the details for the 'plato' key are shown:

Key Size: 104 B Length: 1 TTL: No limit

Field Value TTL

Papas fritas con huevos revueltos muy rico No Limit

- Conéctate al Esclavo y lee la clave

The screenshot shows the Redis Browser application window. The title bar reads "127.0.0.1:6380 - Browser". The menu bar includes "Window", "View", and "Help". The main interface displays a list of databases, with "db0" selected. A search bar at the top right says "Filter by Key Name or Pattern". On the left, there's a sidebar with icons for "Edit", "New", and "Import". The main content area shows a table with one row:

Total: 1	Last refresh: now	Columns	
HASH	plato	No limit	104 B

On the right, a detailed view of the "plato" key is shown:

plato
Key Size: 104 B Length: 1 TTL: No limit
Last refresh: now

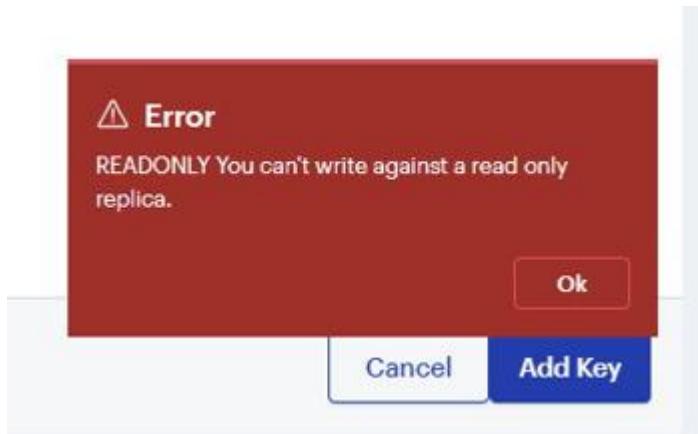
Below this, a table shows the field-value pairs:

Field	Value	TTL
Papas fritas con huevos revueltos	muy rico	No Limit

- 6) Prueba de Integridad: Intenta escribir un dato directamente en el esclavo. ¿Qué ocurre? Explica por qué. (Añade pantallazos y explicaciones)

Nos da un error que indica que la Réplica es solo de lectura y por lo tanto no se puede escribir en ella.

La Réplica está diseñada para ser una copia del maestro. No tiene sentido poder escribir datos en ella de manera manual. Solo debe copiar la información que se contiene en el maestro.



REPLICACIÓN EN CADENA

El nodo esclavo se encuentra en una red geográfica distinta y queremos distribuir los datos a una tercera oficina sin sobrecargar al Maestro original. Configuraremos una topología en cascada: A → B → C.

- Añade un tercer servicio al docker-compose.yaml llamado redis-replica-chained (Puerto 6381).
- Configura este tercer nodo para que replique **del nodo réplica** (redis-replica), NO del maestro.

Modificamos el docker-compose.yaml que se nos entregó previamente e introducimos lo siguiente:

```
redis-replica-chained:
  image: redis:7-alpine
  container_name: redis-replica-chained
# Le decimos que para la conexión con el host use el 6381
  ports:
    - "6381:6379"
# Misma red que los otros. Si no, no podrán conectarse
  networks:
    - redis-net
# Debe esperarse a “redis-replica” para iniciarse
  depends_on:
    - redis-replica
# Le decimos explícitamente que su maestro es 'redis-master'
  command: redis-server --replicaof redis-replica 6379
```

Imagen del archivo docker-compose.yaml modificado:

```
version: '3.8'

services:
    # -----
    # ACTIVIDAD 1 y 3: EL NODO MAESTRO (MASTER)
    # -----
    redis-master:
        image: redis:7-alpine
        container_name: redis-master
        ports:
            - "6379:6379"
        networks:
            - redis-net
        command: redis-server --appendonly yes

    # -----
    # ACTIVIDAD 1: EL PRIMER ESCLAVO (REPLICA DIRECTA)
    # -----
    redis-replica:
        image: redis:7-alpine
        container_name: redis-replica
        ports:
            - "6380:6379"
        networks:
            - redis-net
        depends_on:
            - redis-master
        # Le decimos explicitamente que su maestro es 'redis-master'
        command: redis-server --replicaof redis-master 6379

    # -----
    # ACTIVIDAD 2: Topología en cascada: MAESTRO -> ESCLAVO 1 -> ESCLAVO 2
    # -----
    redis-replica-chained:
        image: redis:7-alpine
        container_name: redis-replica-chained
        ports:
            - "6381:6379"
        networks:
            - redis-net
        depends_on:
            - redis-replica
        # Le decimos explicitamente que su maestro es 'redis-master'
        command: redis-server --replicaof redis-replica 6379

networks:
    redis-net:
        driver: bridge
```

Lo levantamos con “docker compose up -d”.

```
C:\Users\UsuarioASIR\proyectos\sgbd_tema04_actividad_01>docker compose up -d
time="2025-12-02T10:16:27+01:00" level=warning msg="C:\\\\Users\\\\UsuarioASIR\\\\proyectos\\\\sgbd_tema04_actividad_01\\\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
✓ Container redis-master      Running
✓ Container redis-replica    Running
✓ Container redis-replica-chained Started

C:\Users\UsuarioASIR\proyectos\sgbd_tema04_actividad_01>docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED        STATUS        PORTS          NAMES
dee7933f62ff   redis:7-alpine "docker-entrypoint.s..."  11 seconds ago  Up 11 seconds  0.0.0.0:6381->6379/tcp, [::]:6381->6379/tcp   redis-replica-chained
c770229cc222   redis:7-alpine "docker-entrypoint.s..."  15 minutes ago Up 15 minutes  0.0.0.0:6380->6379/tcp, [::]:6380->6379/tcp   redis-replica
bde673c2254c   redis:7-alpine "docker-entrypoint.s..."  15 minutes ago Up 15 minutes  0.0.0.0:6379->6379/tcp, [::]:6379->6379/tcp   redis-master
```

En este caso en Redis-Insight no se autoconfigura la conexión. Tendremos que hacerlo de forma manual.

Tan sencillo como darle a “Add Redis database” e indicar la IP (local en nuestro caso) y el Puerto.

The screenshot shows the Redis Databases page in Redis-Insight. On the left, there is a list of existing databases with their connection URLs: "Free Redis Cloud DB" (redis://default@127.0.0.1:6381), "127.0.0.1:6380" (redis://127.0.0.1:6380), and "127.0.0.1:6379" (redis://127.0.0.1:6379). A modal dialog box titled "Add Database" is open in the center. It contains a "Connection URL" input field with the value "redis://default@127.0.0.1:6381". Below the input field are "Test Connection" and "Connection Settings" buttons, and a large blue "Add Database" button. At the bottom of the dialog, there is a link to "Get started with Redis Cloud account" and three options: "Add databases" (cloud icon), "New database" (FREE icon), and "Import from file".

Haz una prueba de escritura en el Maestro(A)

The screenshot shows the Apache Ignite web interface for a database named 'db0'. The main table lists a single key-value pair:

Total:	1
HASH	plato
	No limit
	104 B

The detailed view for the 'plato' key shows the following information:

Key Size: 128 B Length: 2 TTL: No limit

Field Value TTL

| Papas fritas con huevos revueltos | muy rico | No Limit |
| guiso | tambien esta rico | No Limit |

Verifica que el dato llega a la Réplica Intermedia (B).

Databases / 127.0.0.1:6380 ▾ db0 🔍 ⓘ

0.43 % | 0 | 1 MB | 1 | 3 | ⏪

All Key Types | Filter by Key Name or Pattern

Total: 1 Last refresh: <1 min

HASH plato Columns

Key Size: 128 B Length: 2 TTL: No limit

Last refresh: <1 min

Unicode | Show TTL | Add Fields

Field	Value	TTL
Papas fritas con huevos revueltos	muy rico	No Limit
guiso	tambien esta rico	No Limit

Verifica que el dato llega a la Réplica Final (C).

The screenshot shows the Cloudant NoSQL Database interface. The top navigation bar includes 'Databases / 127.0.0.1:6381' (with a dropdown arrow), 'db0' (with a refresh icon), and an info icon. On the right are icons for 'Cloud sign in', a gear, a question mark, and a bulk actions button. Below the navigation is a search bar labeled 'Filter by Key Name or Pattern'.

The main area displays a table with one row:

Total: 1	Last refresh: now	Columns
1	Now	1 Column

The single column is labeled 'HASH' and has a value 'plato'. To the right, there is a detailed view of the 'plato' key:

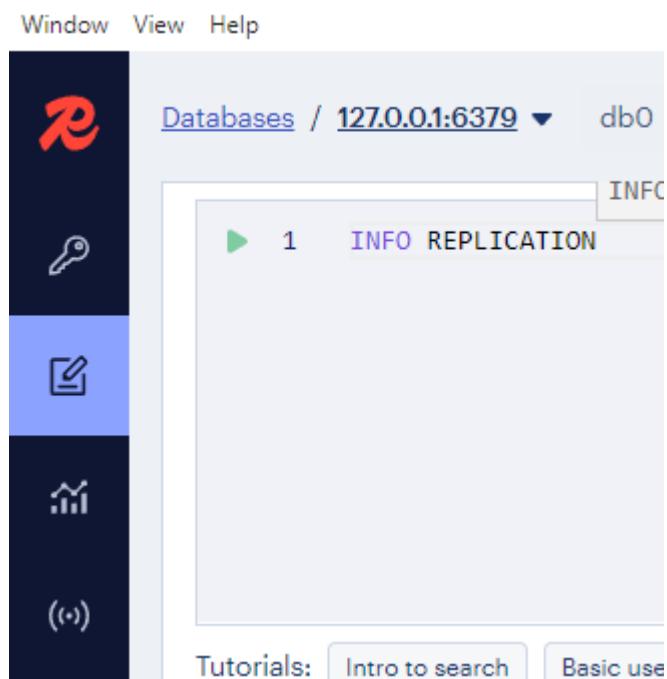
Key Size: 128 B Length: 2 TTL: No limit Last refresh: now

Below this is a table with three columns: 'Field', 'Value', and 'TTL'. It contains two rows:

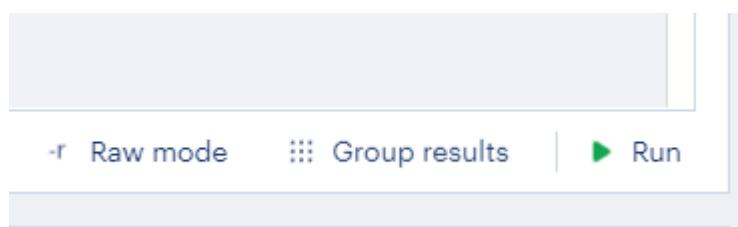
Field	Value	TTL
Papas fritas con huevos revueltos	muy rico	No Limit
guiso	tambien esta rico	No Limit

Ejecuta INFO REPLICATION en todos los nodos, explica que ves y la diferencia entre nodos.

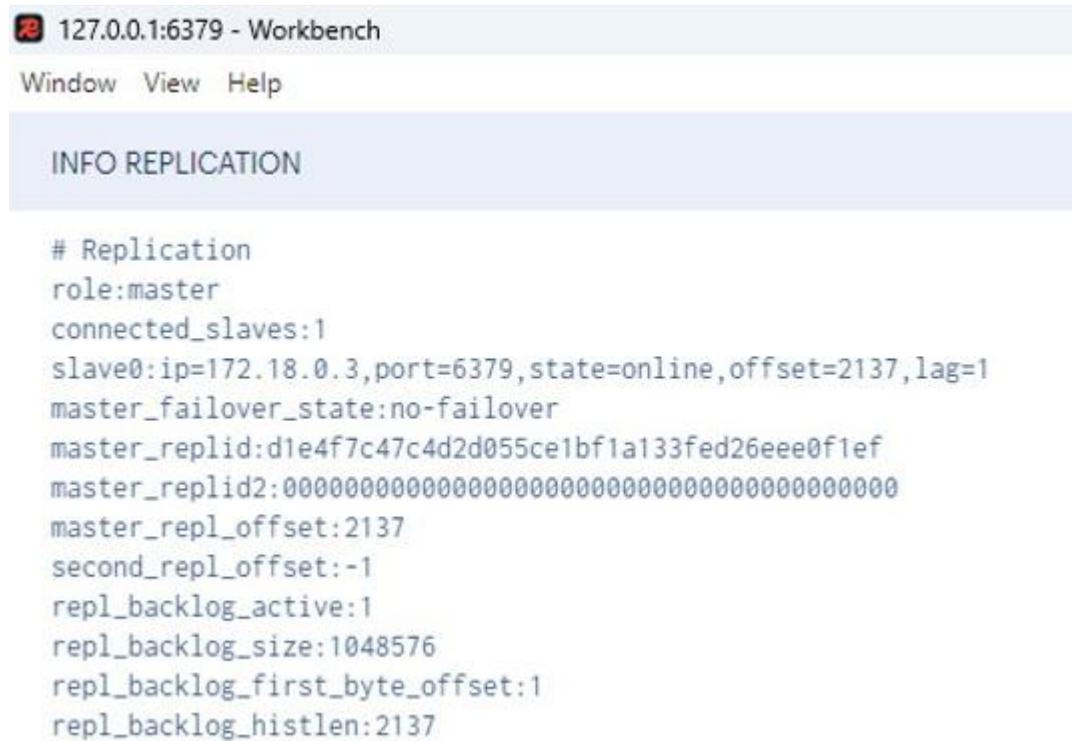
Seleccionamos la base de datos que queremos y nos vamos al panel de la izquierda, donde seleccionaremos “Workbench”. Introducimos “INFO REPLICATION” y luego nos vamos al botón verde de PLAY (No es el que aparece en la imagen de abajo) y lo pulsamos.



Es este botón.



Resultado en **redis-master**:



The screenshot shows a MySQL Workbench interface with the title bar "127.0.0.1:6379 - Workbench". Below the title bar is a menu bar with "Window", "View", and "Help". The main area is titled "INFO REPLICATION". The content of the replication information is as follows:

```
# Replication
role:master
connected_slaves:1
slave0:ip=172.18.0.3,port=6379,state=online,offset=2137,lag=1
master_failover_state:no-failover
master_replid:d1e4f7c47c4d2d055ce1bf1a133fed26eee0f1ef
master_replid2:000000000000000000000000000000000000000000000000000000000000000
master_repl_offset:2137
second_repl_offset:-1
repl_backlog_active:1
repl_backlog_size:1048576
repl_backlog_first_byte_offset:1
repl_backlog_histlen:2137
```

Se indica cuál es su rol (“master”), cuantos esclavos tiene conectado (1), datos del esclavo: IP, puerto, estado y lag (lo que se tarda en sincronizar).

Resultado en redis-replica:

The screenshot shows the Redis Workbench interface with the title bar "127.0.0.1:6380 - Workbench" and menu items "Window", "View", "Help". The main window displays the output of the "INFO REPLICATION" command. The output is as follows:

```
# Replication
role:slave
master_host:redis-master
master_port:6379
master_link_status:up
master_last_io_seconds_ago:4
master_sync_in_progress:0
slave_read_repl_offset:2179
slave_repl_offset:2179
slave_priority:100
slave_read_only:1
replica_announced:1
connected_slaves:1
slave0:ip=172.18.0.4,port=6379,state=online,offset=2179,lag=1
master_failover_state:no-failover
master_replid:d1e4f7c47c4d2d055ce1bf1a133fed26eee0f1ef
master_replid2:000000000000000000000000000000000000000000000000000000000000000
master_repl_offset:2179
second_repl_offset:-1
replog_backlog_active:1
replog_backlog_size:1048576
replog_backlog_first_byte_offset:1
replog_backlog_histlen:2179
```

Se indica su rol ("slave"). El nombre, puerto y estado de su master. Cuanto fue la última operación I/O con el master (hace 4 segundos). Si se está produciendo una sincronización en estos instantes. Su prioridad (100) y se indica que es una réplica en la que solo se puede leer ("slave-read-only= 1") ➔ ¿Quiero esto decir que hay réplicas en las que se puede escribir?

También se indica que tiene esclavos (1) y los datos de este, de igual manera a como aparecían en cuando hicimos esta consulta en el master.

Resultado en redis-replica-chained:

Se indica su rol y el nombre, puerto y estado de su master (redis-replica). También se indica su prioridad y que es una réplica de solo escritura.