


9 artículos



MSI Spatium M460 SSD 1TB M.2 NVMe PCIe Gen4x4

61,99€

Recibelo entre el lunes 10 y el miércoles 12 de marzo

−

1

+

Servicios disponibles

☐ Protégelo frente a daños con PcCare **por 5.99€** ⓘ


+1 año

+2 años

☐ Protégelo frente a daños 4 años y amplía 1 año la garantía **por 22.39€** ⓘ

☐ Soporte premium 365 + técnico a domicilio **por 34€** ⓘ

☐ Migración de sistema operativo **por 17.98€** ⓘ



Tecnoware Era Plus 1100 SAI Línea Interactiva 1100VA 770W con 2 Salidas Schuko

88,99€

Recibelo entre el viernes 7 y el lunes 10 de marzo

Vendido por Tecnoware

−

1

+

Servicios disponibles

☐ Amplia tu garantía con PcCare **por 12.29€** ⓘ


+1 año

+2 años

☐ Protégelo frente a daños con PcCare **por 6.29€** ⓘ

+1 año

+2 años



Forgeon Lichborne Pasta Térmica 4g 13.5W/mK

5,99€ ~~7,99€~~

Recibelo mañana

−

1

+

Servicios disponibles

☐ Soporte premium 365 + técnico a domicilio **por 34€** ⓘ



Nox Hummer ELEMENT Semi Torre ATX Cristal Templado USB-C Blanca y Madera

69,99€ ~~79,90€~~

Recibelo mañana

−

1

+

Servicios disponibles

☐ Protégelo frente a daños con PcCare **por 5.99€** ⓘ


+1 año

+2 años

☐ Protégelo frente a daños 4 años y amplía 1 año la garantía **por 22.39€** ⓘ

☐ Soporte premium 365 + técnico a domicilio **por 34€** ⓘ

☐ Tuning de ventiladores **por 7.99€** ⓘ



Corsair RMe Series RM750e 750W 80 Plus Gold Modular

132,89€

Recibelo mañana

−

1

+

Servicios disponibles


☐ Protégelo frente a daños con PcCare **por 8.89€** ⓘ

+1 año

+2 años

☐ Protégelo frente a daños 4 años y amplía 1 año la garantía **por 36€** ⓘ

☐ Soporte premium 365 + técnico a domicilio **por 34€** ⓘ



WD Blue 3.5" 2TB 5400RPM 64MB CRM SATA 3

75,99€

Recibelo mañana

−

1

+

Servicios disponibles

☐ Protégelo frente a daños con PcCare **por 5.99€** ⓘ

+1 año

+2 años

☐ Protégelo frente a daños 4 años y amplía 1 año la garantía **por 22.39€** ⓘ

☐ Soporte premium 365 + técnico a domicilio **por 34€** ⓘ



## **Justificación de la elección de componentes para el equipo de una clínica oftalmológica**

### **1. Procesador: AMD Ryzen 7 5700G 4.6GHz**

El Ryzen 7 5700G es un procesador de alto rendimiento con 8 núcleos y 16 hilos, ideal para el uso en una clínica oftalmológica donde se requiere un funcionamiento fluido de software especializado en imágenes médicas, gestión de pacientes y análisis de datos. Además, sus gráficos integrados Radeon Vega 8 permiten el procesamiento de imágenes sin necesidad de una tarjeta gráfica dedicada, reduciendo costos y consumo energético.

### **2. Placa base: Gigabyte B550I Aorus Pro AX**

Esta placa base Mini-ITX es ideal para un entorno clínico gracias a su tamaño compacto y alto rendimiento. Su compatibilidad con PCIe 4.0 permite un acceso rápido a datos en SSD de alta velocidad, mejorando la fluidez del software oftalmológico. Además, incluye conectividad Wi-Fi 6 y Bluetooth 5.1, lo que permite una conexión estable con dispositivos médicos y sistemas de red sin depender exclusivamente de cables.

### **3. Almacenamiento: MSI Spatium M460 SSD 1TB M.2 NVMe PCIe Gen4x4**

En una clínica oftalmológica, el almacenamiento rápido es clave para el manejo eficiente de imágenes de diagnóstico y expedientes electrónicos. Este SSD de alto rendimiento con velocidades superiores a 5000 MB/s garantiza una carga rápida de archivos y software, mejorando la productividad del personal médico.

### **4. Chasis: Nox Hummer ELEMENT Semi Torre ATX Cristal Templado USB-C Blanca y Madera**

Un gabinete con diseño moderno y elegante, adecuado para un entorno clínico. Su estructura en blanco y madera proporciona una apariencia profesional y armoniosa con el ambiente de la clínica. La inclusión de USB-C facilita la conexión de dispositivos médicos modernos y su ventilación eficiente mantiene una temperatura óptima para el equipo.

### **5. SAI: Tecnoware Era Plus 1100 SAI Línea Interactiva 1100VA 770W con 2 Salidas Schuko**

Un sistema de alimentación ininterrumpida es crucial en una clínica para evitar la pérdida de datos en caso de cortes de energía. Con una capacidad de 770W, este SAI permite mantener el equipo funcionando el tiempo suficiente para guardar información crítica y apagar el sistema de manera segura.

### **6. Fuente de alimentación: Corsair RMe Series RM750e 750W 80 Plus Gold Modular**

La eficiencia 80 Plus Gold asegura un consumo energético optimizado, algo esencial en un entorno clínico donde varios equipos electrónicos están en funcionamiento simultáneamente. Su diseño modular permite una mejor gestión de cables, reduciendo

desorden y mejorando la ventilación del gabinete.

7. **Almacenamiento adicional: WD Blue 3.5" 2TB 5400RPM 64MB CRM SATA 3**

Este disco duro mecánico ofrece una capacidad de 2TB, ideal para almacenar grandes volúmenes de datos, como historiales médicos, imágenes oftalmológicas y registros de pacientes. Su baja velocidad de 5400 RPM lo hace silencioso y eficiente en consumo energético.

8. **Memoria RAM: Team Group T-Force Delta RGB DDR5 6000MHz 32GB (2x16GB) CL30**

La memoria DDR5 de 6000MHz garantiza un rendimiento óptimo en aplicaciones médicas avanzadas. Los 32GB permiten ejecutar sin problemas software especializado, multitarea intensiva y procesamiento de imágenes oftalmológicas con alta resolución. Además, su velocidad y baja latencia aseguran una respuesta rápida del sistema.

### **Conclusión**

La selección de estos componentes garantiza un equipo eficiente, estable y rápido, adecuado para una clínica oftalmológica. Su capacidad de procesamiento, almacenamiento veloz y respaldo de energía aseguran un flujo de trabajo óptimo, mejorando la experiencia tanto para los profesionales de la salud como para los pacientes.

1. Lista todos los dispositivos de bloque y sus puntos de montaje.

```
java@DESKTOP-F2TV6UF MINGW64 ~/Desktop/AlpinePractica
$ vagrant ssh
alpine319:~$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda   8:0    0  128G  0 disk
├─sda1 8:1    0   300M  0 part /boot
└─sda2 8:2    0 127.7G  0 part /
sdb   8:16   0   20G   0 disk
sdc   8:32   0   10G   0 disk
alpine319:~$
```

2. Muestra el espacio libre en los dispositivos con un formato de numeros humanos.

```
alpine319:~$ df -h
Filesystem      Size      Used Available Use% Mounted on
devtmpfs        10.0M         0      10.0M   0% /dev
shm             990.9M         0      990.9M   0% /dev/shm
/dev/sda2       125.1G    169.3M    118.6G   0% /
tmpfs           396.4M    156.0K    396.2M   0% /run
df: /sys/kernel/debug/tracing: Permission denied
/dev/sda1       271.1M     23.0M    229.1M   9% /boot
tmpfs           990.9M      4.0K    990.9M   0% /tmp
alpine319:~$
```

3. Muestra el uso de disco de cada directorio en el sistema, ocultando los

```
alpine319:~$ du -hs /* 2>/dev/null
2.5M    /bin
22.9M   /boot
0       /dev
2.8M    /etc
16.0K   /home
41.4M   /lib
16.0K   /media
4.0K    /mnt
4.0K    /opt
0       /proc
156.0K  /run
3.4M    /sbin
4.0K    /srv
0       /sys
4.0K    /tmp
89.7M   /usr
27.2M   /var
alpine319:~$
```

errores.

#### 4. Crea una nueva partición en dev/sdb.

```
alpine319:~$ echo -e "n\np\n1\n\n+5G\nnw" | sudo fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI, OSF or GPT disklabel
Building a new DOS disklabel. Changes will remain in memory only,
2 requierir reinicio and will not be written to disk. Use the write command to write them. After that the previous content
won't be recoverable.

The number of cylinders for this disk is set to 2610.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
 1) software that runs at boot time (e.g., old versions of LILO)
 2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): Partition type
   p   primary partition (1-4)
   e   extended

Partition number (1-4): First sector (63-41943039, default 63): Using default value 63
Last sector or +size{K,M,G,T} (63-41943039, default 41943039):
Command (m for help): The partition table has been altered.
Calling ioctl() to re-read partition table
alpine319:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0  128G  0 disk
├─sda1       8:1    0   300M  0 part /boot
└─sda2       8:2    0 127.7G  0 part /
sdb          8:16   0   20G   0 disk
└─sdb1       8:17   0    5G   0 part
sdc          8:32   0   10G   0 disk
alpine319:~$
```

5. Formatea la partition /dev/sdb1 con el sistema de archivos XFS

```
alpine319:~$ sudo mkfs.xfs -f /dev/sdb1
meta-data=/dev/sdb1            isize=512    agcount=4, agsize=327680 blks
       =                       sectsz=512    attr=2, projid32bit=1
       =                       crc=1        finobt=1, sparse=1, rmapbt=0
       =                       reflink=1    bigtime=1 inobtcount=1 nnext64=0
data      =                       bsize=4096   blocks=1310720, imaxpct=25
       =                       sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log        =internal log      bsize=4096   blocks=16384, version=2
       =                       sectsz=512    sunit=0 blks, lazy-count=1
realtime  =none               extsz=4096   blocks=0, rtextents=0

alpine319:~$ sudo mkfs.xfs -f /dev/sdb1
meta-data=/dev/sdb1            isize=512    agcount=4, agsize=327680 blks
       =                       sectsz=512    attr=2, projid32bit=1
       =                       crc=1        finobt=1, sparse=1, rmapbt=0
       =                       reflink=1    bigtime=1 inobtcount=1 nnext64=0
data      =                       bsize=4096   blocks=1310720, imaxpct=25
       =                       sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log        =internal log      bsize=4096   blocks=16384, version=2
       =                       sectsz=512    sunit=0 blks, lazy-count=1
realtime  =none               extsz=4096   blocks=0, rtextents=0

alpine319:~$ sudo mkfs.xfs -f /dev/sdb1
meta-data=/dev/sdb1            isize=512    agcount=4, agsize=327680 blks
       =                       sectsz=512    attr=2, projid32bit=1
       =                       crc=1        finobt=1, sparse=1, rmapbt=0
       =                       reflink=1    bigtime=1 inobtcount=1 nnext64=0
data      =                       bsize=4096   blocks=1310720, imaxpct=25
       =                       sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log        =internal log      bsize=4096   blocks=16384, version=2
```

6. Comprueba el sistema de archivos en /dev/sdb1/debe estar desmon

```
lpine319:~$ sudo umount /dev/sdb1
mount: can't unmount /dev/sdb1: Invalid argument
lpine319:~$
```

8. Lista los volúmenes físicos con LVM.

```
alpine319:~$ sudo vgdisplay vg_datos
--- Volume group ---
VG Name                vg_datos
System ID
Format                 lvm2
Metadata Areas         2
Metadata Sequence No   1
VG Access              read/write
VG Status              resizable
MAX LV                 0
Cur LV                0
Open LV               0
Max PV                 0
Cur PV                2
Act PV                2
VG Size                29.99 GiB
PE Size                4.00 MiB
Total PE              7678
Alloc PE / Size        0 / 0
Free PE / Size         7678 / 29.99 GiB
VG UUID                9QY7Sq-8lRL-0XMP-rjr8-d0H2-a8yT-JWcGfA
```

9. Crea grupo de volúmenes llamado vg\_datos con los dispositivos disponibles y extiende el grupo con un nuevo dispositivo.

```
alpine319:~$ sudo vgcreate vg_datos /dev/sdb1
Volume group "vg_datos" successfully created
alpine319:~$
```

```
alpine319:~$ sudo vgextend vg_datos /dev/sdc
Volume group "vg_datos" successfully extended
alpine319:~$
```

10. Crea un volumen lógico de 30Gb en el grupo de volúmenes vg\_datos y extiende el volumen lógico para usar todo el espacio libre disponible

```
alpine319:~$ sudo lvcreate -L 30G -n lv_datos vg_datos
```

```
Logical volume "lv_datos" created.
alpine319:~$
```

```
alpine319:~$ sudo lvextend -l +100%FREE /dev/vg_datos/lv_datos
```



1. Utiliza el comando Docker pull para descargar la imagen oficial de MYSQL desde Docker Hub.

```
java@DESKTOP-F2TV6UF MINGW64 ~
$ docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
893b018337e2: Download complete
23d22e42ea50: Download complete
df1ba1ac457a: Download complete
f56a22f949f9: Download complete
2be0d473cadf: Download complete
cc9646b08259: Download complete
277ab5f6ddde: Download complete
43759093d4f6: Download complete
d255dceb9ed5: Download complete
431b106548a3: Download complete
```

2. listado de imagenes: Utiliza el comando docker images para listar las imagenes Docker disponibles en tu maquina local.

```
java@DESKTOP-F2TV6UF MINGW64 ~
$ docker images
```

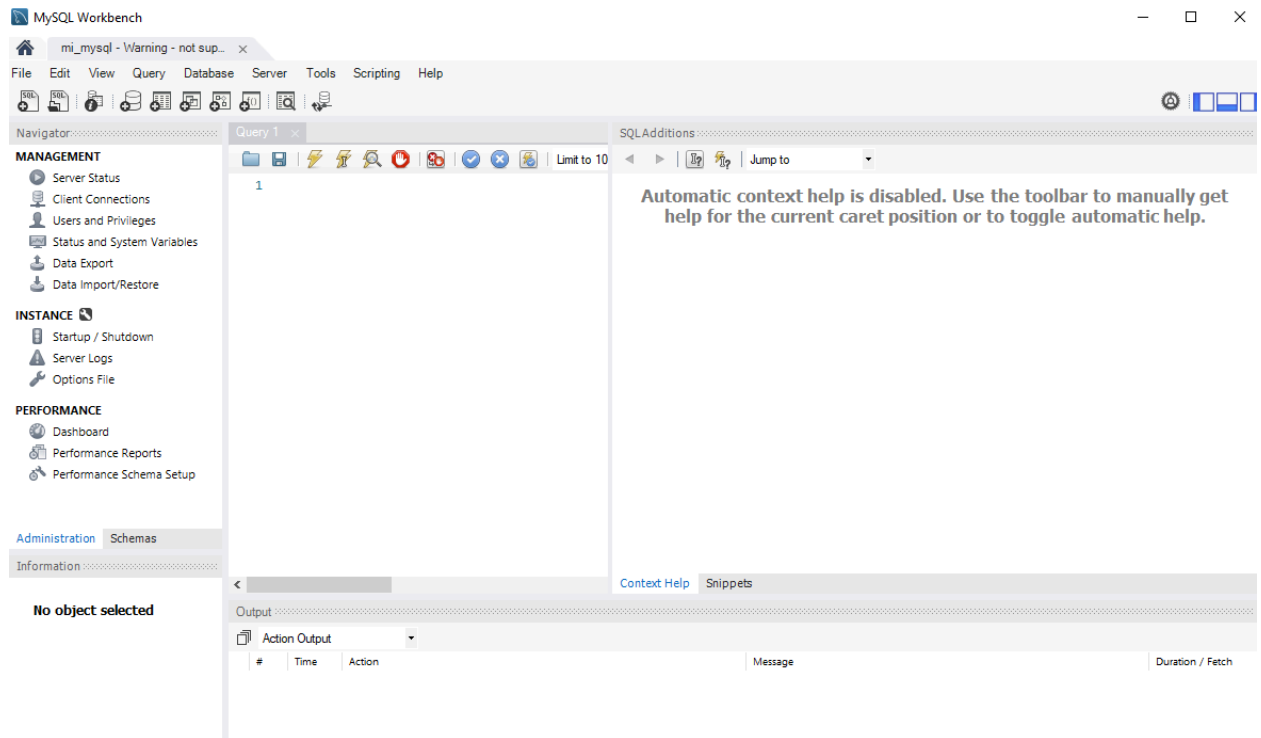
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
getting-started	latest	092a9df24c72	2 days ago	342MB
nginx	latest	9d6b58feebd2	3 weeks ago	279MB
ubuntu	latest	72297848456d	5 weeks ago	117MB
ubuntu	22.04	ed1544e45498	5 weeks ago	117MB
mysql	latest	146682692a3a	5 weeks ago	1.09GB
rockylinux	9-minimal	305de618a568	15 months ago	176MB
docker/welcome-to-docker	latest	eedaff45e3c7	16 months ago	29.5MB
docker/getting-started	latest	d79336f4812b	2 years ago	73.9MB
nginx	1.21.6	2bcabc23b454	2 years ago	213MB

3. Ejecucion de un Contenedor: Utiliza el comando docker run para crear y ejecutar un contenedor basado en la imagen MYSQL.  
Configura variables de entorno para establecer contraseña de root y nombre de la base de datos.  
Mapea el puerto 3305 del contenedor al puerto 3305 de tu maquina local.

4. Verificación de contenedor: Utiliza el comando `docker ps` para verificar que el contenedor se está ejecutando.

Conéctate al contenedor MySQL utilizando un cliente de base de datos (por ejemplo MySQL Workbench para comprobar que la base de datos está accesible).

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker ps  
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES  
8542ac61a166   mysql    "docker-entrypoint.s..." About a minute Up About a minute 33060/tcp, 0.0.0.0:3305->3306/tcp   mi-mysql  
java@DESKTOP-F2TV6UF MINGW64 ~
```



5. Visualización de logs: Utiliza el comando docker logs para ver los logs del contenedor Mysql

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker logs mi-mysql  
2025-03-05 11:45:30+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.  
2025-03-05 11:45:31+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'  
2025-03-05 11:45:31+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.  
2025-03-05 11:45:31+00:00 [Note] [Entrypoint]: Initializing database files  
2025-03-05T11:45:31.763608Z 0 [System] [MY-015017] [Server] MySQL Server Initialization - start.  
2025-03-05T11:45:31.771528Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 9.2.0) initializing of server in progress as process 81  
2025-03-05T11:45:31.799074Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.  
2025-03-05T11:45:33.443185Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.  
2025-03-05T11:45:36.578422Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.  
2025-03-05T11:45:40.711629Z 0 [System] [MY-015018] [Server] MySQL Server Initialization - end.  
2025-03-05 11:45:40+00:00 [Note] [Entrypoint]: Database files initialized  
2025-03-05 11:45:40+00:00 [Note] [Entrypoint]: Starting temporary server  
2025-03-05T11:45:40.836258Z 0 [System] [MY-015015] [Server] MySQL Server - start.  
2025-03-05T11:45:41.081628Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 9.2.0) starting as process 122  
2025-03-05T11:45:41.102582Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.  
2025-03-05T11:45:41.771066Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.  
2025-03-05T11:45:43.318280Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.  
2025-03-05T11:45:43.318391Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.  
2025-03-05T11:45:43.332592Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a different directory.  
2025-03-05T11:45:43.382657Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock  
2025-03-05T11:45:43.383045Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '9.2.0' socket: '/var/run/mysqld/mysql.sock' port: 0 MySQL Community Server - GPL.  
2025-03-05 11:45:43+00:00 [Note] [Entrypoint]: Temporary server started.  
'/var/lib/mysql/mysql.sock' -> '/var/run/mysqld/mysqld.sock'  
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.  
Warning: Unable to load '/usr/share/zoneinfo/leap-seconds.list' as time zone. Skipping it.
```

6. Ejecución de un comando dentro de un contenedor  
Utiliza el comando docker exec para ejecutar un comando simple dentro del contenedor Mysql (por ejemplo my sql -u root -p para acceder a la consola MYSQL)

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker exec -it mi-mysql mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 11  
Server version: 9.2.0 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2025, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

7. Detencion y reinicio del Contenedor: Utiliza comando docker stop para detener el contenedor.

Utiliza comando docker start para volver a iniciar el contenedor.

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker stop mi-mysql  
mi-mysql  
  
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker start mi-mysql  
mi-mysql
```

8. Informacion sobre una imagen:

Utiliza el comando Docker inspect para obtener informacion detallada sobre la imagen de MySQL (puedes usar el nombre o el ID).

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker inspect mysql  
[  
  {  
    "Id": "sha256:146682692a3aa409eae7b7dc6a30f637c6cb49b6ca901c2cd160becc81127d3b",  
    "RepoTags": [  
      "mysql:latest"  
    ],  
    "RepoDigests": [  
      "mysql@sha256:146682692a3aa409eae7b7dc6a30f637c6cb49b6ca901c2cd160becc81127d3b"  
    ],  
    "Parent": "",  
    "Comment": "buildkit.dockerfile.v0",  
    "Created": "2025-01-22T17:15:22Z",  
    "DockerVersion": "27.5.1",  
    "Author": "",  
    "Config": {  
      "Hostname": "",  
      "Domainname": "",  
      "User": "",  
      "AttachStdin": false,  
      "AttachStdout": false,  
      "AttachStderr": false,  
      "ExposedPorts": {  
        "3306/tcp": {},  
        "33060/tcp": {}  
      },  
      "Tty": false,  
      "OpenStdin": false,  
      "StdinOnce": false,  
      "Env": [  
        "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",  
        "GOSU_VERSION=1.17",  
        "MYSQL_MAJOR=innovation",  
        "MYSQL_VERSION=9.2.0-1.el9",  
      ]  
    },  
    "Architecture": "amd64",  
    "Os": "linux"  
  }  
]
```

```

        "MYSQL_VERSION=9.2.0-1.el9",
        "MYSQL_SHELL_VERSION=9.2.0-1.el9"
    ],
    "Cmd": [
        "mysqld"
    ],
    "Image": "",
    "Volumes": {
        "/var/lib/mysql": {}
    },
    "WorkingDir": "/",
    "Entrypoint": [
        "docker-entrypoint.sh"
    ],
    "OnBuild": null,
    "Labels": null
},
"Architecture": "amd64",
"Os": "linux",
"Size": 241141026,
"GraphDriver": {
    "Data": null,
    "Name": "overlayfs"
},
"RootFS": {
    "Type": "layers",
    "Layers": [
        "sha256:d7b2257a2277cf62be373f5e32a77274c67cdcdb474af84f2f01b95d8f6ebd0d",
        "sha256:9986054dfacc4debd8fc996e555c5e2180d49fc21ebfd005c0b1e78136ce1104",
        "sha256:8c039733b996247dfe9588782ce1385a887e4995037a6707abba9fed1b7a67ba",
        "sha256:433b6ea8deb0f770e884e6fbb5fe88e18c0a5e6625906c6c5dd05cc753b0c60f",
        "sha256:5494a032973c2c9f03c2d978222d83555d6e86b6d1602f2f69335202d1c6125f",
        "sha256:bfb4799a5e3c7bee4d194639adf9f4393c9b02fe45289ddd5d499ddeb9bd277",
        "sha256:501c1708175892aebd454706f7e09ff3bbea4c899e26af302ccd87eba0580f1d",
        "sha256:21cd7e76ec3c2ddf6445fd06ef7d21d46e9ff4e4c04e732b5e1aef54d3d969e8",
        "sha256:cfb04b43e3382dfe2082e7f8d800454ded906bc71a284363a7a820583bac93ed",

```

```

        "sha256:bccedbf428d268739e947541440ec43f4ce59524d4ae1034634256ae5fb4e7a1"
    ]
},
"Metadata": {
    "LastTagTime": "2025-03-05T11:08:54.235003026Z"
}
}
]

```

Eliminacion del contenedor: Utiliza comando docker rm para eliminar contenedor.

```

java@DESKTOP-F2TV6UF MINGW64 ~
$ docker stop mi-mysql
mi-mysql

java@DESKTOP-F2TV6UF MINGW64 ~
$ docker rm mi-mysql
mi-mysql

```

Eliminacion de una imagen: Utiliza el comando docker rmi para eliminar la imagen de MySQL(si ya no la necesitas)

```
java@DESKTOP-F2TV6UF MINGW64 ~  
$ docker rmi mysql  
Untagged: mysql:latest  
Deleted: sha256:146682692a3aa409eae7b7dc6a30f637c6cb49b6ca901c2cd160becc81127d3b
```