

实验四：MINIOB 实验一

一、实验目的

1. 对 OceanBase 有一定了解
2. 在本地成功运行 minioB
3. 编写基础的 SQL 命令：建表、插入数据、更新、删除
4. 熟悉了解 minioB 环境搭建过程，为下次实验打好基础

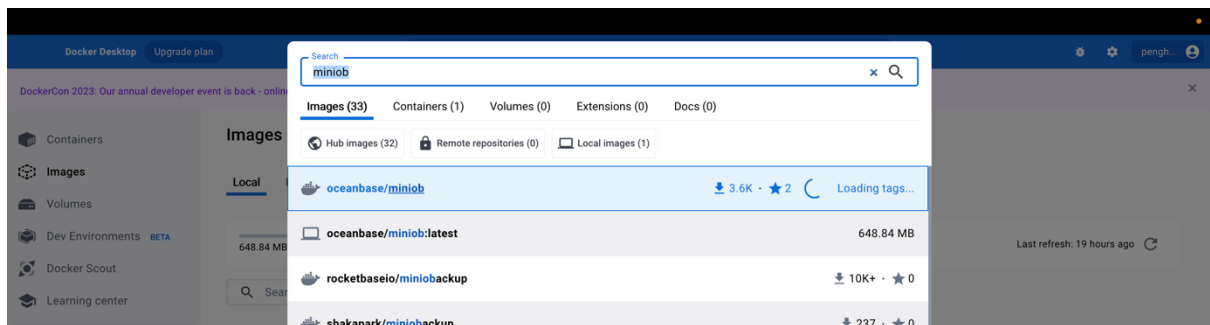
二、实验要求说明

1. 在个人电脑搭建好实验环境
2. 提供 Docker 运行截图
3. 按要求完成第四节中的实验，每步操作后要求截图，实验截图要求包含完整的 SQL 执行语句和输出的实验结果

三、minioB 安装说明

1. 在 Docker 中直接下载官方镜像

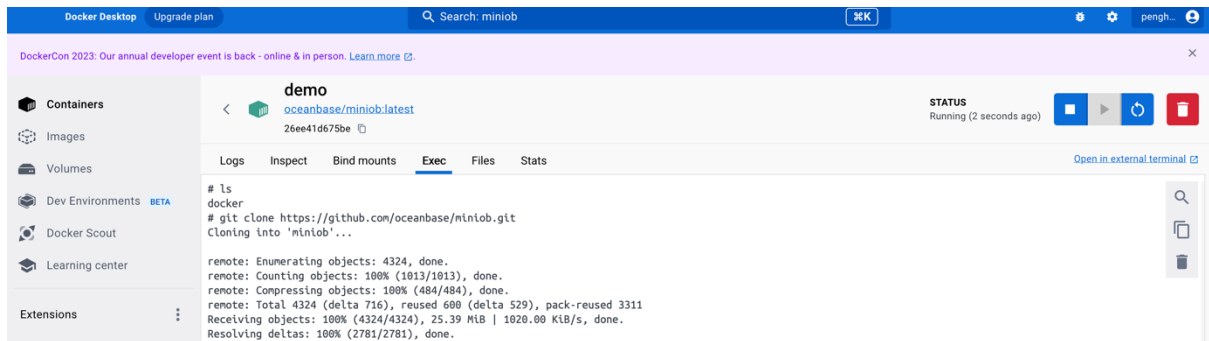
在 Docker 顶部搜索框搜索 minioB，oceanbase/minioB 即为官方镜像。



点击打开后，选择 pull 并 run 即可运行

2. 进入容器，下载并编译 minioB

在 Exec 下运行 `git clone https://github.com/oceanbase/minioB.git` 从 gthub 上下载 minioB。



进入 miniob 目录，使用 `bash build.sh --make -j4` 进行编译

```
# cd miniob
# ls
benchmark  cmake          CODE_OF_CONDUCT.md  deps  docs  etc  NOTICE  src  tools
build.sh   CMakeLists.txt CONTRIBUTING.md     docker Doxyfile License README.md test  unittest
# bash build.sh --make -j4
build.sh --make -j4
create soft link for build_debug, linked by directory named build
cmake -DCMAKE_EXPORT_COMPILE_COMMANDS=1 --log-level=STATUS /root/miniob -DCMAKE_BUILD_TYPE=Debug -DDEBUG=ON
-- The C compiler identification is GNU 11.4.0
-- The CXX compiler identification is GNU 11.4.0
-- Detecting C compiler ABI info
[100%] Linking CXX executable ../bin/pidfile_test
[100%] Built target pidfile_test
[100%] Linking CXX executable ../bin/ring_buffer_test
[100%] Linking CXX executable ../bin/record_manager_test
[100%] Built target persist_test
[100%] Built target ring_buffer_test
[100%] Built target record_manager_test
#
```

3. 运行 miniOB

编译完成后，进入 build 目录，使用 `./bin/observer -s miniob.sock -f ../etc/observer.ini &` 在后台启动服务端。

```
# cd build
# ./bin/observer -s miniob.sock -f ../etc/observer.ini &
# Successfully load ../etc/observer.ini
```

启动服务端后，使用 `./bin/obclient -s miniob.sock` 启动 miniOB 客户端。

```
# ./bin/obclient -s miniob.sock
miniob > help
Commands
show tables;
desc `table name`;
create table `table name` (`column name` `column type`, ...);
create index `index name` on `table` (`column`);
insert into `table` values(`value1`,`value2`);
update `table` set column=value [where `column`=`value`];
delete from `table` [where `column`=`value`];
select [ * | `columns` ] from `table`;
miniob >
```

四、实验内容

1. 在 Docker 中建立 miniob 环境（参考第 0 章的教程 PPT 和第三节的内容）
2. 创建一张表，包括学号，姓名，成绩

表名: Scores		
字段	说明	类型
id	学号	int
name	姓名	char(10)
score	成绩	float

3. 向这张表里面插入几行数据

id	name	score
2251435	李明浩	81.2
2210465	赵毅斌	91.3
2332133	刘孔阳	56.3
2231435	王亚伟	73.2
1950723	孙鹏翼	89.2

4. 使用 select 语句展示学号，姓名
5. 尝试修改指定行的成绩如下表所示，能否成功？为什么？

id	name	score
2251435	李明浩	91.3
2210465	赵毅斌	91.3
2332133	刘孔阳	56.3
2231435	王亚伟	87.2
1950723	孙鹏翼	89.2

6. 删除赵毅斌和孙鹏翼的记录
7. 对 miniob 源码进行阅读，主要选取一个功能（如 create table、insert、delete 等）进行分析理解，做简要报告（不超过两页）