# 1 安装环境

操作系统：CentOS Linux release 7.4.1708 (Core)

内存：12G

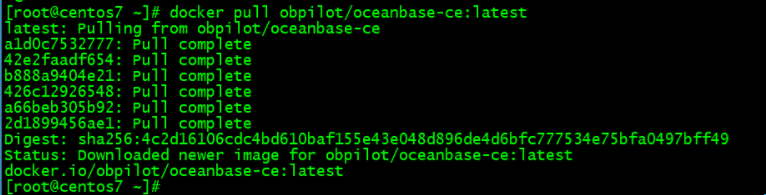
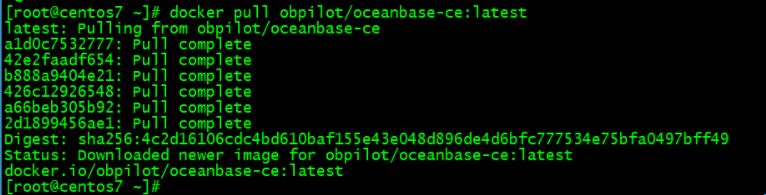
硬盘：40G

DOCKER版本：23.0.1

# 2 DOCKR安装OB数据库

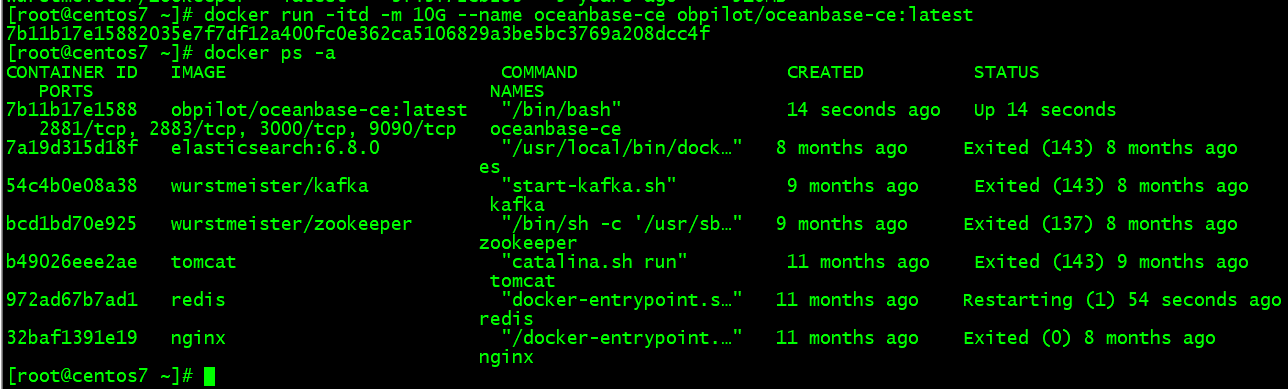
## 2.1 镜像的拉取和启动

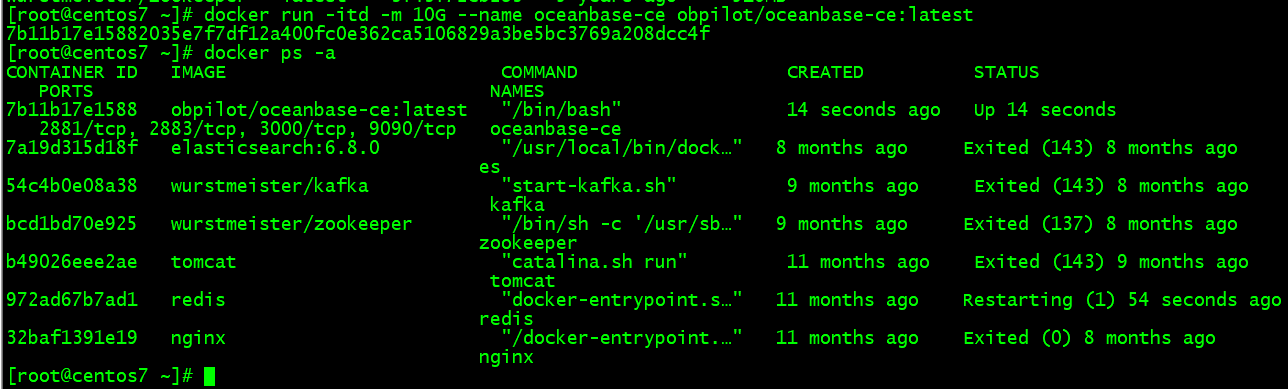
docker pull obpilot/oceanbase-ce:latest



[root@centos7 ~]# docker run -itd -m 10G --name oceanbase-ce obpilot/oceanbase-ce:latest

7b11b17e15882035e7f7df12a400fc0e362ca5106829a3be5bc3769a208dcc4f

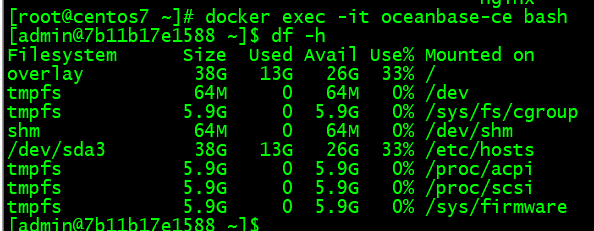




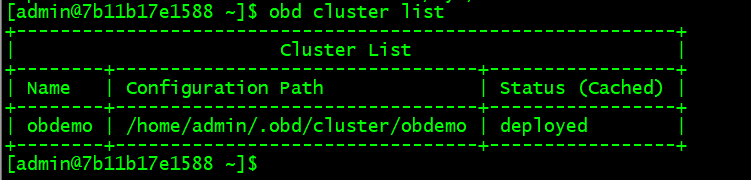
## 2.2 检查集群状态和启动集群

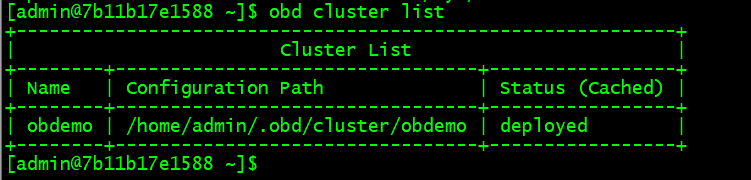
查看集群状态

[root@centos7 ~]# docker exec -it oceanbase-ce bash



[admin@7b11b17e1588 ~]$ obd cluster list

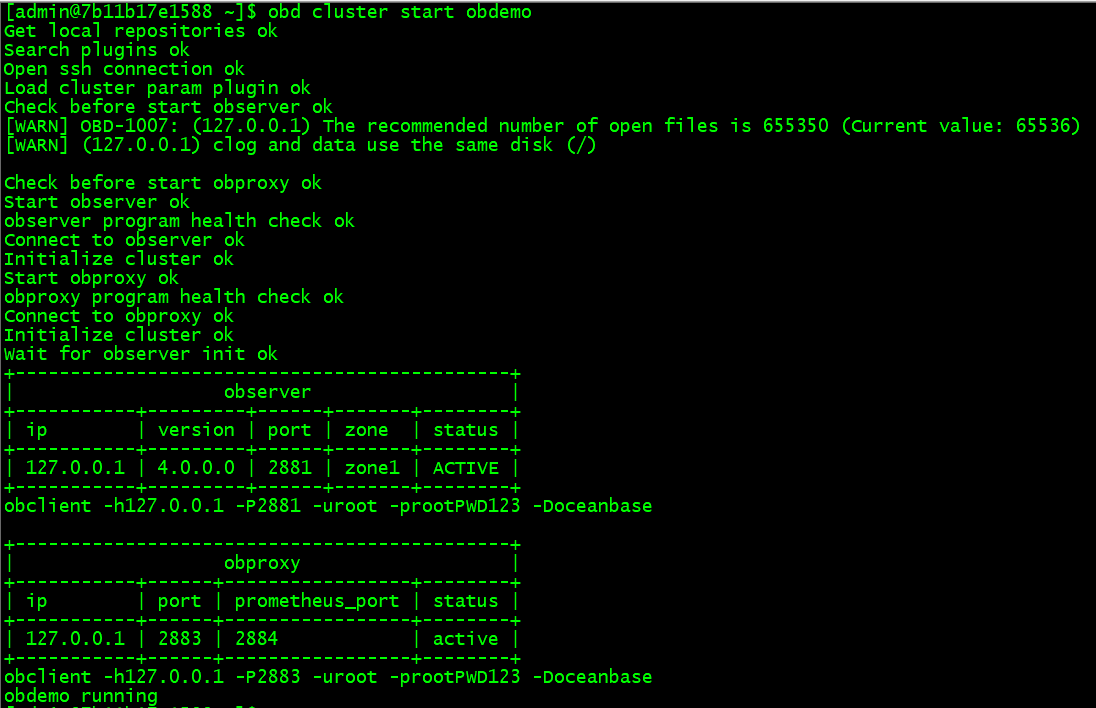


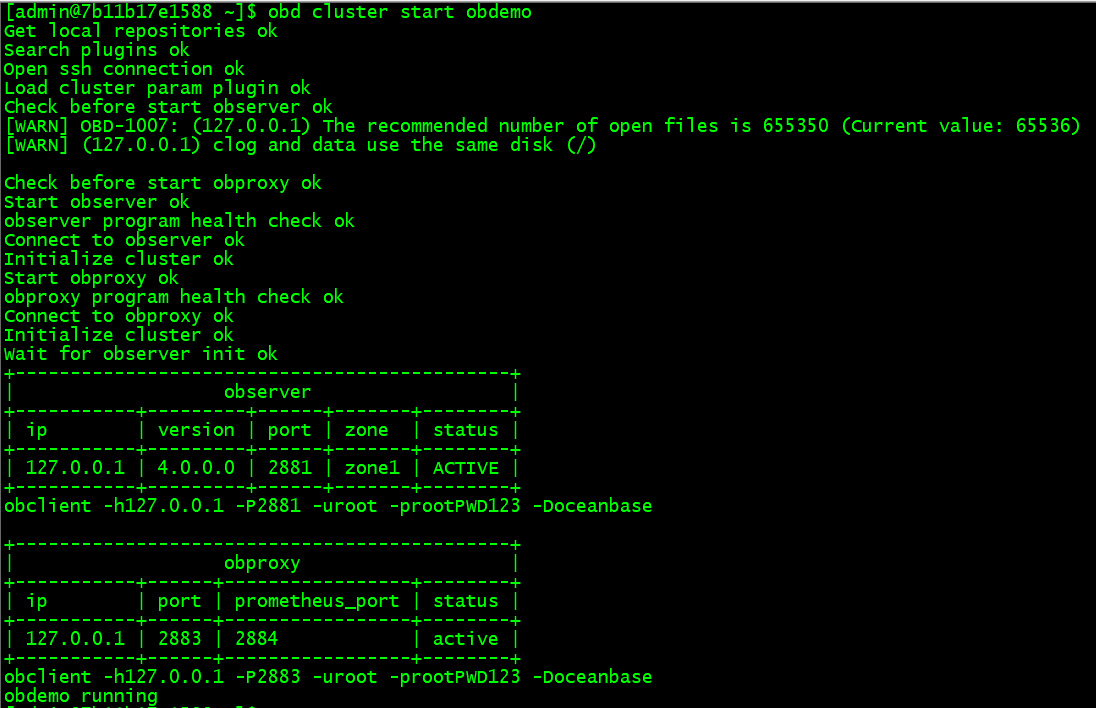


启动数据库

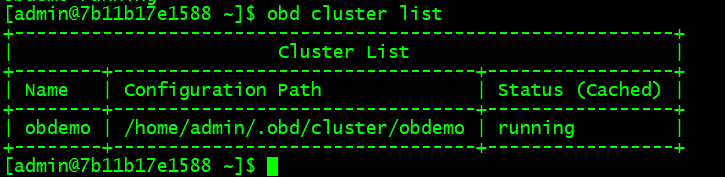
[admin@7b11b17e1588 ~]$ obd cluster start obdemo

admin密码：adminPWD123





再查看状态为running



# 3 创建租户，数据库，表

## 3.1 登录数据库

登录数据库

[admin@7b11b17e1588 ~]$ obclient -h127.0.0.1 -uroot@sys -P2881 -prootPWD123 -c -A oceanbase

参数解析：

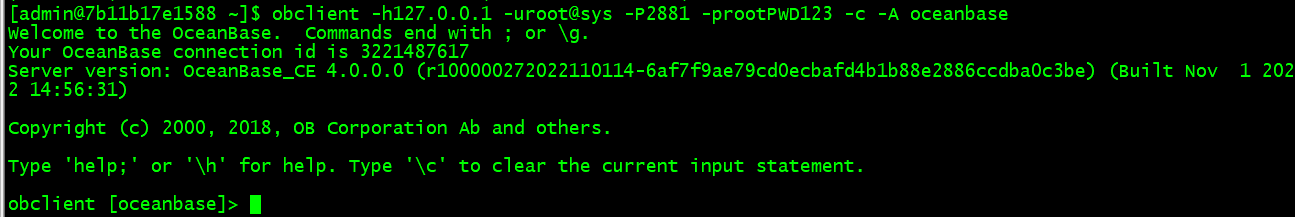
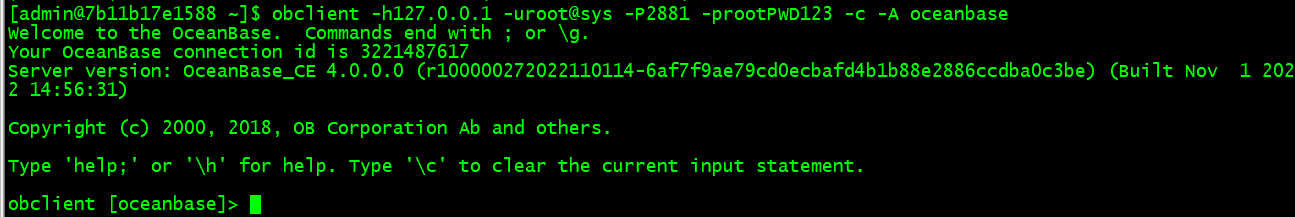
-h：提供 OceanBase 数据库连接的域名

-P：提供 OceanBase 数据库连接端口，Oracle 模式租户默认是 1521。

-u：提供租户的连接账号。

-p：提供账号密码。为了安全可以不提供，改为在后面提示符下输入，密码文本不可见。

-A：表示连接数据库时不获取全部表信息，以快速登录数据库。



创建用户租户需理解下面4张表的逻辑关系：

**资源规格**：对应 DBA\_OB\_Unit\_CONFIGS 表，定义了常见物理资源项的大小，包括 CPU、内存、磁盘空间、IOPS 等。创建资源池时指定其资源规格，从而根据定义创建资源单元

**资源单元Unit**：对应 DBA\_OB\_UNITS 表，是资源调度的基本单位，具有节点、Zone、Region 等位置属性

**资源池**：对应 DBA\_OB\_RESOURCE\_POOLS 表，是资源分配的基本单位，创建资源池时会实际创建 Unit，如对应节点预留资源不够将会创建失败

**租户**：对应 DBA\_OB\_TENANTS 表，创建租户需指定RESOURCE\_POOL\_LIST资源池，从而拥有指定资源池的 Unit

创建租户的依赖关系是：租户 -> 资源池 -> 资源规格，因此创建租户时需要反过来，先创建资源规格。

## 3.2 创建资源规格

USE oceanbase;

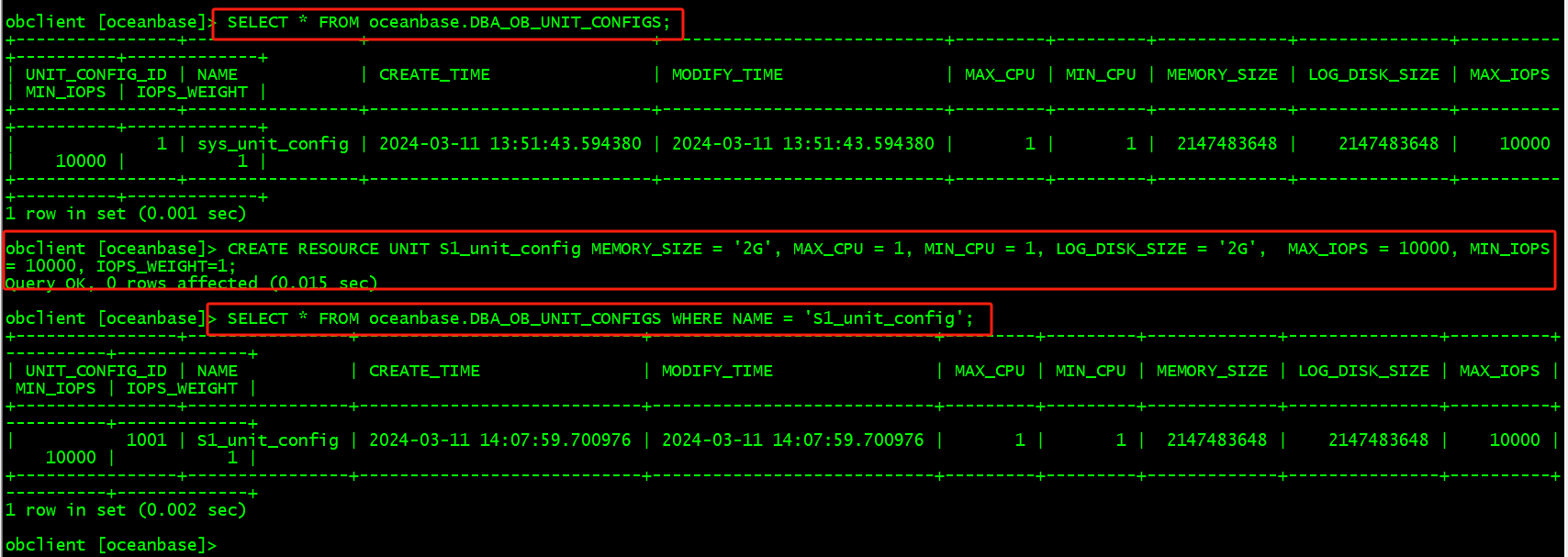
查看资源规格信息：SELECT \* FROM oceanbase.DBA\_OB\_UNIT\_CONFIGS;

创建一个名称为 S1\_unit\_config 的资源规格，其资源配置为 CPU 为 1 核，内存 2G，日志盘空间 2G：

CREATE RESOURCE UNIT S1\_unit\_config MEMORY\_SIZE = '2G', MAX\_CPU = 1, MIN\_CPU = 1, LOG\_DISK\_SIZE = '2G', MAX\_IOPS = 10000, MIN\_IOPS = 10000, IOPS\_WEIGHT=1;

查询 DBA\_OB\_UNIT\_CONFIGS 视图，确认资源规格创建成功

SELECT \* FROM oceanbase.DBA\_OB\_UNIT\_CONFIGS WHERE NAME = 'S1\_unit\_config';



**unit\_name**：资源规格名称。

**MEMORY\_SIZE**：内存规格，其最小值由隐藏配置项 \_\_min\_full\_resource\_pool\_memory 控制，默认为 5G。

**MAX\_CPU**：CPU 规格上限，MIN\_CPU 是 CPU 规格下限，单位是核数。如果用户没有指定 MIN\_CPU，默认等于 MAX\_CPU 值。

**MIN\_CPU**：最小的 CPU 规格，所有租户的 MIN\_CPU 的总和不能超过该节点 CPU 总容量 CPU\_CAPACITY。

**MAX\_IOPS 和 MIN\_IOPS**： 指定当前 Unit 可使用的 IOPS 资源的上限和下限，最小值均为 1024，并且要求 MAX\_IOPS >= MIN\_IOPS。

**LOG\_DISK\_SIZE**：日志盘空间，OceanBase 数据库 V4.0 会按租户管理日志盘空间，系统为各个租户预留日志盘空间，实现按租户隔离。当用户没有指定时，LOG\_DISK\_SIZE 默认值是内存规格的 3 倍大小，最小为 2G。

OceanBase 数据库 V4.0 支持租户间 IOPS 隔离，通过 MAX\_IOPS、MIN\_IOPS、IOPS\_WEIGHT 三个参数决定。IOPS 规格参数默认不需要指定，系统根据 CPU 规格自动计算。

## 3**.3 创建资源池**

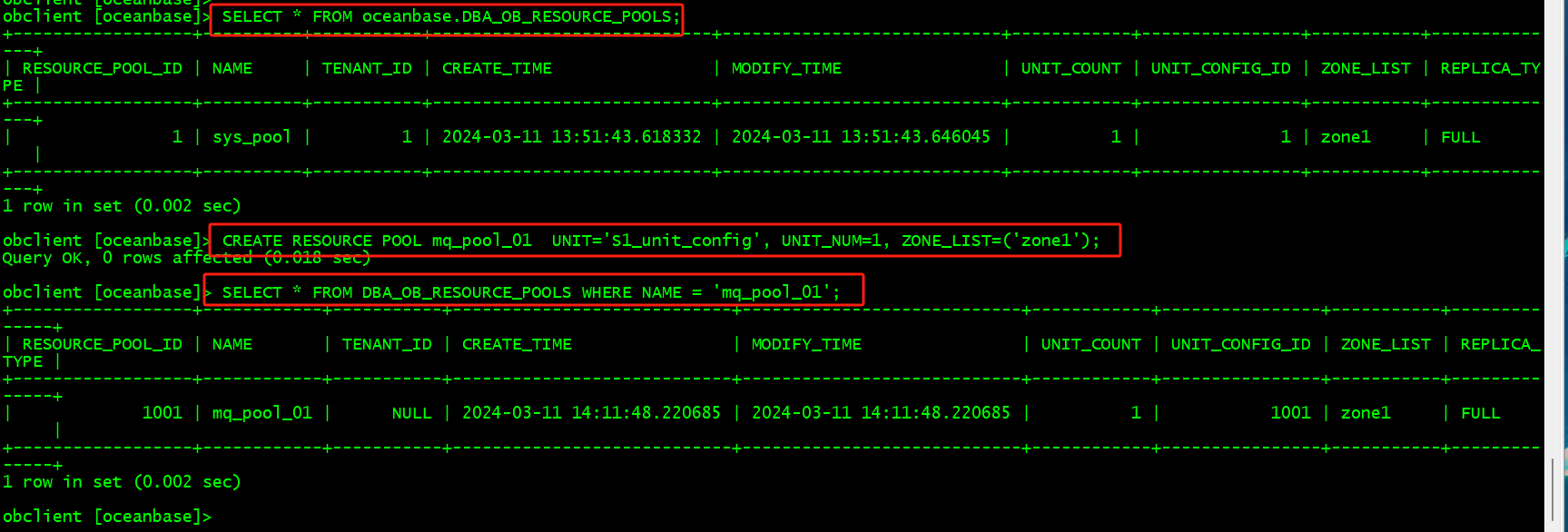
创建资源规格后，可以在创建资源池时指定资源规格，从而使用相应大小的资源单元，并最终分配给相应的租户。

查看已有资源池：SELECT \* FROM oceanbase.DBA\_OB\_RESOURCE\_POOLS;

创建名为mq\_pool\_01 的资源池，在zone1创建1个 Unit，每个Unit的资源规格为S1\_unit\_config

CREATE RESOURCE POOL mq\_pool\_01 UNIT='S1\_unit\_config', UNIT\_NUM=1, ZONE\_LIST=('zone1');

查询资源池是否创建成功：SELECT \* FROM DBA\_OB\_RESOURCE\_POOLS WHERE NAME = 'mq\_pool\_01';



**UNIT**：该资源池的资源规格，输入资源规格名称。

**UNIT\_NUM**：该资源池在目标 Zone 的 Unit 个数，它的值要小于目标 Zone 的节点个数，且每个节点上最多只能分布同一个租户的一个 Unit。

**ZONE\_LIST**：资源池的 Zone 分布。OceanBase 数据库会在 ZONE\_LIST 描述的每个 Zone 里面创建 UNIT\_NUM 个 Unit，每个 Unit 的资源规格为 UNIT 参数设置的值。

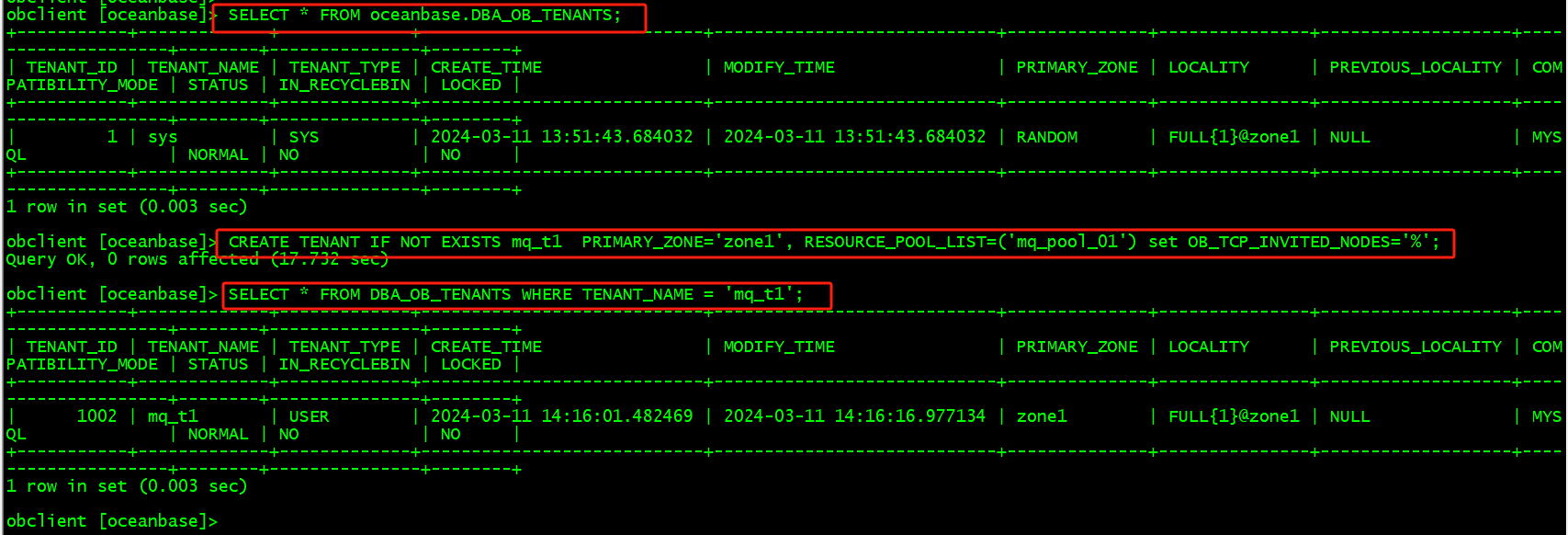
## 3.4 创建租户

查看所有的租户信息：SELECT \* FROM oceanbase.DBA\_OB\_TENANTS;

创建一个名为 mq\_t1 的租户(默认为 MySQL 模式租户)，副本数为1，资源池指定为 mq\_pool\_01，Primary Zone 为 zone1允许所有 IP 连接数据库。

CREATE TENANT IF NOT EXISTS mq\_t1 PRIMARY\_ZONE='zone1', RESOURCE\_POOL\_LIST=('mq\_pool\_01') set OB\_TCP\_INVITED\_NODES='%';

查询确认租户创建成功：SELECT \* FROM DBA\_OB\_TENANTS WHERE TENANT\_NAME = 'mq\_t1';

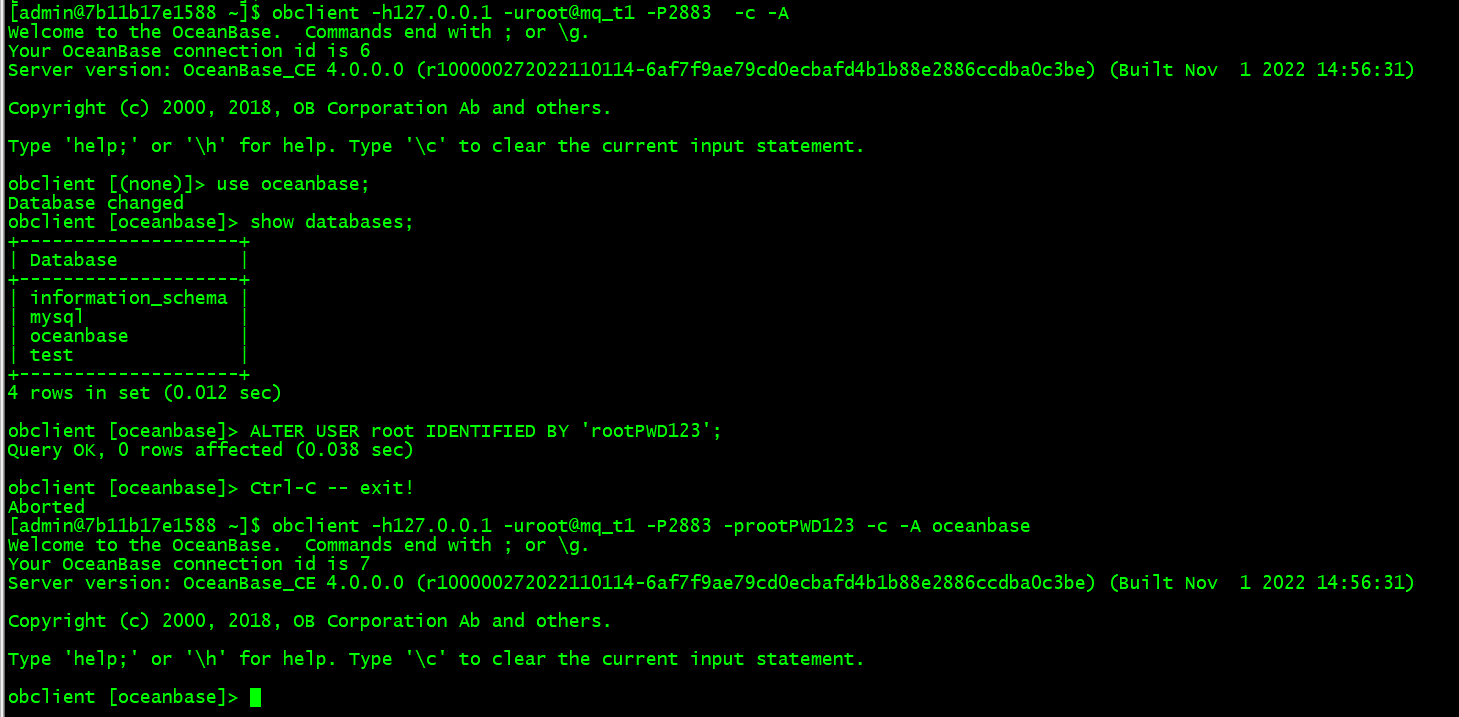


租户创建成功后，可以尝试登录 mq\_t1 租户,默认管理员用户（MySQL 模式为 root，Oracle 模式为 sys）的密码为空,需要及时修改管理员用户的密码

登录：obclient -h127.0.0.1 -uroot@mq\_t1 -P2883 -c -A

修改密码：use oceanbase; ALTER USER root IDENTIFIED BY 'rootPWD123';

修改密码成功后，重新登录租户：obclient -h127.0.0.1 -uroot@mq\_t1 -P2883 -prootPWD123 -c -A oceanbase



**IF NOT EXISTS**：可选参数，如果要创建的租户名已存在，并且没有指定 IF NOT EXISTS，则会出现错误。

t**enant\_name**：指定租户名称，最长 128 个字符，字符只能是大小写英文字母、数字和下划线，而且必须以字母或下划线开头。

**PRIMARY\_ZONE**：指定租户的 Primary Zone，指定了租户提供读写服务的 Zone 的优先级。实际上是一个 Zone 的列表，列表中包含多个 Zone。当列表包含多个 Zone 时，使用分号（;）分隔的 Zone 具有从高到低的优先级，使用逗号（,）分隔的 Zone 具有相同优先级，表示流量打散在多个 Zone 上，这几个 Zone 同时提供服务。

例如，primary\_zone ='zone1;zone2,zone3' 表示该租户优先由 zone1 提供读写服务，zone1 比 zone2、zone3 的优先级高，zone2 和 zone3 是同一优先级。在指定 PRIMARY\_ZONE 时，其值可以设置为 RANDOM（必须大写），表示随机选择最高优先级内的任意一个 Zone 作为 Primary Zone。

**RESOURCE\_POOL\_LIST**：指定分配给租户的资源池列表，必填。如果有多个资源池，要求多个资源池的 UNIT\_NUM 个数一致。

租户的副本分布的 Zone List 继承于 RESOURCE\_POOL\_LIST 的 ZONE\_LIST 属性，租户的副本个数继承于 RESOURCE\_POOL\_LIST 的 ZONE\_LIST 属性的个数，租户在每个 Zone 的 Unit 个数继承于 RESOURCE\_POOL\_LIST 的 UNIT\_NUM 属性、租户的 Unit 的资源规格继承于 RESOURCE\_POOL\_LIST 的 Unit 属性。

**ENABLE\_ARBITRATION\_SERVICE**：指定租户是否开启仲裁服务，如果不显示指定，默认创建的租户其仲裁服务为关闭状态。如果创建租户的时候未开启仲裁服务，可以在创建租户后再开启仲裁服务，详细操作请参见 为租户开启仲裁服务。该内容仅适用于 OceanBase 数据库企业版。OceanBase 数据库社区版暂不支持仲裁服务功能。

**system\_var\_name**：指定系统变量或配置项。

**OB\_TCP\_INVITED\_NODES**：用于指定租户连接的白名单，即允许哪些客户端 IP 连接该租户。示例中 % 表示所有客户端都可登录，如果不指定 OB\_TCP\_INVITED\_NODES 的值，则默认租户的连接方式为只允许本机的 IP 登录该租户。白名单配置的详细介绍，参见 查看和设置白名单。

**OB\_COMPATIBILITY\_MODE**：用于指定租户的兼容模式，可选择 MySQL 或 Oracle 兼容模式，并且只能在创建时指定。如果不指定 OB\_COMPATIBILITY\_MODE，则默认兼容模式为 MySQL 模式。