

Contenedor con imagen MySQL

1. En un solo comando Docker:

- a. Se comprueba, desde la línea de comandos de Docker, que existe una imagen de nombre **mysql**.

```
root@CLIENTE:/home/alumno# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
zarritas/ubuntu	latest	df6b868e4eef	2 weeks ago	77.9MB
mi-hola-mundo	latest	4018e0017d13	4 weeks ago	7.38MB
ubuntu	latest	fd1d8f58e8ae	4 weeks ago	77.9MB
mysql	latest	73246731c4b0	2 months ago	619MB
httpd	latest	6fd77d7e5eb7	4 months ago	167MB
hello-world	latest	d2c94e258dcb	9 months ago	13.3kB

- b. Se descarga esa imagen de DockerHub. La página oficial es https://hub.docker.com/_/mysql. Se recomienda leer en contenedor de esta página a que se dan indicaciones de cómo arrancar la imagen en un contenedor, qué parámetros se pueden usar...
- c. Se ejecutará la imagen en un contenedor, de nombre **contiene_mysql**.
- i. En el propio comando de arranque de la imagen en el contenedor, y para facilitar el uso de MySQL, se creará una variable de entorno con la que se le indica a MySQL cuál es la contraseña de root, que será **admin**. Para conocer qué variables de entorno hay y cómo se pueden usar, se puede consultar la página oficial de la imagen, antes citada.
- ii. El contenedor se ejecutará en segundo plano (*detached*).

```
root@CLIENTE:/home/alumno# docker run --net=red_despliegue --name contiene_mysql -e  
MYSQL_ROOT_PASSWORD=admin -d mysql  
f2a20c9fcce33c26457edab16f9bbf74200ffa4e7b9995ac0c8fe89c57af6  
d3e
```

2. En la máquina anfitriona, instalar el IDE *MySQL Workbench Community Edition* para conectarse al servidor de MySQL y ejecutar los scripts DDL y DML.

```
root@CLIENTE:/home/alumno# mysql -h 172.18.0.2 -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 12

Server version: 8.2.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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.

```
mysql> CREATE DATABASE SCOTT;
```

Query OK, 1 row affected (0,01 sec)

mysql> USE SCOTT;

Database changed

mysql> DROP TABLE dept; CREATE TABLE dept (

ERROR 1051 (42S02): Unknown table 'SCOTT.dept'

-> deptno decimal(2,0), dname varchar(14), loc varchar(13),

-> CONSTRAINT `PK_dept` PRIMARY KEY(deptno)

->);

Query OK, 0 rows affected (0,06 sec)

mysql> INSERT INTO dept VALUES (10,'ACCOUNTING','NEW YORK'); INSERT INTO

dept VALUES (20,'RESEARCH','DALLAS'); INSERT INTO dept VALUES

(30,'SALES','CHICAGO'); INSERT INTO dept VALUES (40,'OPERATIONS','BOSTON');

Query OK, 1 row affected (0,01 sec)

Query OK, 1 row affected (0,01 sec)

Query OK, 1 row affected (0,01 sec)

Query OK, 1 row affected (0,00 sec)

mysql> DROP TABLE emp;

ERROR 1051 (42S02): Unknown table 'SCOTT.emp'

mysql> CREATE TABLE emp (

-> empno decimal(4,0), ename varchar(10), job varchar(9),

-> mgr decimal(4,0), hiredate date, sal decimal(7,2),

-> comm decimal(7,2), deptno decimal(2,0),

-> CONSTRAINT `PK_emp` PRIMARY KEY (empno),

-> CONSTRAINT `FK_emp_dept_deptno` FOREIGN KEY (deptno) REFERENCES dept
(deptno),

-> CONSTRAINT `FK_emp_emp_mgr` FOREIGN KEY (mgr)

-> REFERENCES emp (empno)

->);

Query OK, 0 rows affected (0,09 sec)

mysql> INSERT INTO emp VALUES

-> ('7839','KING','PRESIDENT',NULL,'1981-11-17','5000.00',NULL,'10'),

-> ('7566','JONES','MANAGER','7839','1981-04-02','2975.00',NULL,'20'),

-> ('7698','BLAKE','MANAGER','7839','1981-05-01','2850.00',NULL,'30'),

-> ('7782','CLARK','MANAGER','7839','1981-06-09','2450.00',NULL,'10'),

-> ('7788','SCOTT','ANALYST','7566','1982-12-09','3000.00',NULL,'20'),

-> ('7902','FORD','ANALYST','7566','1981-12-03','3000.00',NULL,'20'),

-> ('7499','ALLEN','SALESMAN','7698','1981-02-20','1600.00','300.00','30'),

-> ('7521','WARD','SALESMAN','7698','1981-02-22','1250.00','500.00','30'),

-> ('7654','MARTIN','SALESMAN','7698','1981-09-28','1250.00','1400.00','30'),

-> ('7844','TURNER','SALESMAN','7698','1981-09-08','1500.00','0.00','30'),

-> ('7900','JAMES','CLERK','7698','1981-12-03','950.00',NULL,'30'),

-> ('7934','MILLER','CLERK','7782','1982-01-23','1300.00',NULL,'10'),

```

-> ('7876','ADAMS','CLERK','7788','1983-01-12','1100.00',NULL,'20'),
-> ('7369','SMITH','CLERK','7902','1980-12-17','800.00',NULL,'20');
Query OK, 14 rows affected (0,00 sec)
Records: 14  Duplicates: 0  Warnings: 0
mysql> DROP TABLE bonus;
ERROR 1051 (42S02): Unknown table 'SCOTT.bonus'
mysql> CREATE TABLE bonus (
->  ename varchar(10), job varchar(9), sal decimal(7,2), comm decimal(7,2)
-> );
Query OK, 0 rows affected (0,05 sec)
mysql> DROP TABLE salgrade;
ERROR 1051 (42S02): Unknown table 'SCOTT.salgrade'
mysql> CREATE TABLE salgrade (
->  grade decimal(2,0), losal decimal(7,2), hisal decimal(7,2),
->  CONSTRAINT `PK_salgrade` PRIMARY KEY (grade)
-> );
Query OK, 0 rows affected (0,05 sec)
mysql> INSERT INTO salgrade VALUES (1,700,1200); INSERT INTO salgrade VALUES
(2,1201,1400); INSERT INTO salgrade VALUES (3,1401,2000); INSERT INTO salgrade
VALUES (4,2001,3000); INSERT INTO salgrade VALUES (5,3001,9999);
Query OK, 1 row affected (0,01 sec)
Query OK, 1 row affected (0,00 sec)
Query OK, 1 row affected (0,01 sec)
Query OK, 1 row affected (0,00 sec)
Query OK, 1 row affected (0,01 sec)
mysql> COMMIT;
Query OK, 0 rows affected (0,00 sec)

```

3. Arrancar una shell **bash** en el contenedor y:

- a. Comprobar la versión de Linux que está corriendo en el contenedor.
- b. Instalar todos los paquetes necesarios.
- c. Instalar un cliente de MySQL de línea de comandos (mysql-client) y comprobar que las operaciones DDL y DML realizadas con Workbench se repercutieron correctamente en la base de datos.

```
mysql> select * from emp;
```

empno	ename	job	mgr	hiredate	sal	comm	deptno
7369	SMITH	CLERK	7902	1980-12-17	800.00	NULL	20
7499	ALLEN	SALESMAN	7698	1981-02-20	1600.00	300.00	30
7521	WARD	SALESMAN	7698	1981-02-22	1250.00	500.00	30
7566	JONES	MANAGER	7839	1981-04-02	2975.00	NULL	20
7654	MARTIN	SALESMAN	7698	1981-09-28	1250.00	1400.00	30
7698	BLAKE	MANAGER	7839	1981-05-01	2850.00	NULL	30
7782	CLARK	MANAGER	7839	1981-06-09	2450.00	NULL	10
7788	SCOTT	ANALYST	7566	1982-12-09	3000.00	NULL	20
7839	KING	PRESIDENT	NULL	1981-11-17	5000.00	NULL	10
7844	TURNER	SALESMAN	7698	1981-09-08	1500.00	0.00	30
7876	ADAMS	CLERK	7788	1983-01-12	1100.00	NULL	20
7900	JAMES	CLERK	7698	1981-12-03	950.00	NULL	30
7902	FORD	ANALYST	7566	1981-12-03	3000.00	NULL	20
7934	MILLER	CLERK	7782	1982-01-23	1300.00	NULL	10

```
14 rows in set (0,00 sec)
```

Contenedor con imagen Apache

Se va a construir una imagen que contenga un Apache con la capacidad de ejecutar scripts PHP. Para ello se partirá de una imagen de Ubuntu sobre la que se instalará todos los paquetes necesarios. En el siguiente enlace se detalla cómo realizar esta tarea:

Los pasos a dar son:

1. Se descargará la imagen de Ubuntu, de nombre **ubuntu**.
2. Se ejecuta la imagen en un nuevo contenedor de nombre **contiene_apache**.
 - a. Se “mapeará” (vinculará) el puerto 8888 de la máquina anfitriona con el puerto 80 del contenedor.

```
root@CLIENTE:/home/alumno# nano Dockerfile
GNU nano 6.2 Dockerfile
FROM ubuntu
RUN apt-get update && DEBIAN_FRONTEND=noninteractive apt-get install -y apache2
php php_mysqli && apt-get clean

CMD ["apache2ctl","-D","FOREGROUND"]
```

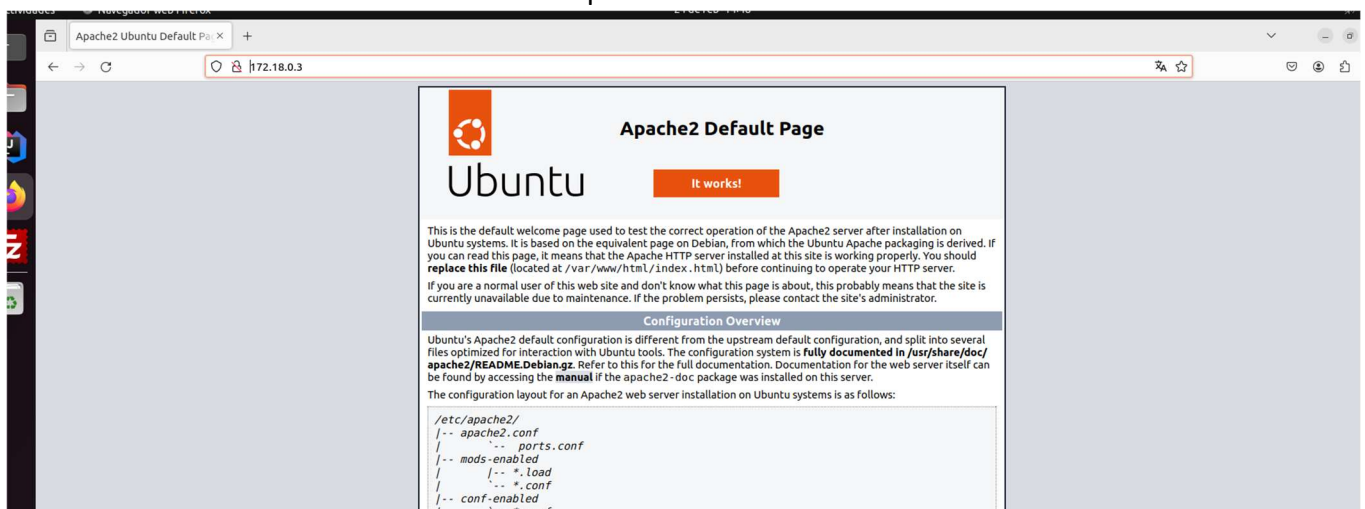
```
root@CLIENTE:/home/alumno/Escritorio# docker build -t mi_imagen_apache.
[+] Building 22.7s (6/6) FINISHED docker:default
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 180B 0.0s
=> [internal] load metadata for docker.io/library/ubuntu:latest 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> CACHED [1/2] FROM docker.io/library/ubuntu:latest 0.0s
=> [2/2] RUN apt-get update && apt-get install -y apache2 21.9s
=> exporting to image 0.7s
=> => exporting layers 0.7s
=> => writing image sha256:3f9aa3144cc9b2ae71bddd467e39bbbfa437e03366c1d75fa9d7636d1e0ab23 0.0s
=> => naming to docker.io/library/mi_imagen_apache
root@CLIENTE:/home/alumno# docker run -i -p 8888:80 --net=red_despliegue --name contiene_apache
mi_imagen_apache
```

```
ls
bin
boot
dev
etc
home
lib
```

```
lib32
lib64
libx32
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var

ls etc/apache2
apache2.conf
conf-available
conf-enabled
envvars
magic
mods-available
mods-enabled
ports.conf
sites-available
sites-enabled
```

3. Se comprueba desde la máquina anfitrión, usando un navegador, que se accede correctamente al Apache existente en el contenedor.



4. Se ejecuta un **bash** sobre ese contenedor.
- Se instalan todos los paquetes necesarios.
 - Si no está ya operativo, se activa el funcionamiento de PHP en Apache.

- c. Se creará, en el directorio raíz de documentos, un script PHP de prueba de nombre **phpinfo.php** para comprobar que funciona el intérprete de PHP en Apache. El código del script es:

```
<?php
phpinfo();
?>
```

- d. Se comprueba desde la máquina anfitrión, invocándolo con un navegador, que se ejecuta correctamente al script PHP recién creado.

5. Se copia el script PHP que accede a MySQL desde la máquina anfitriona al contendor.

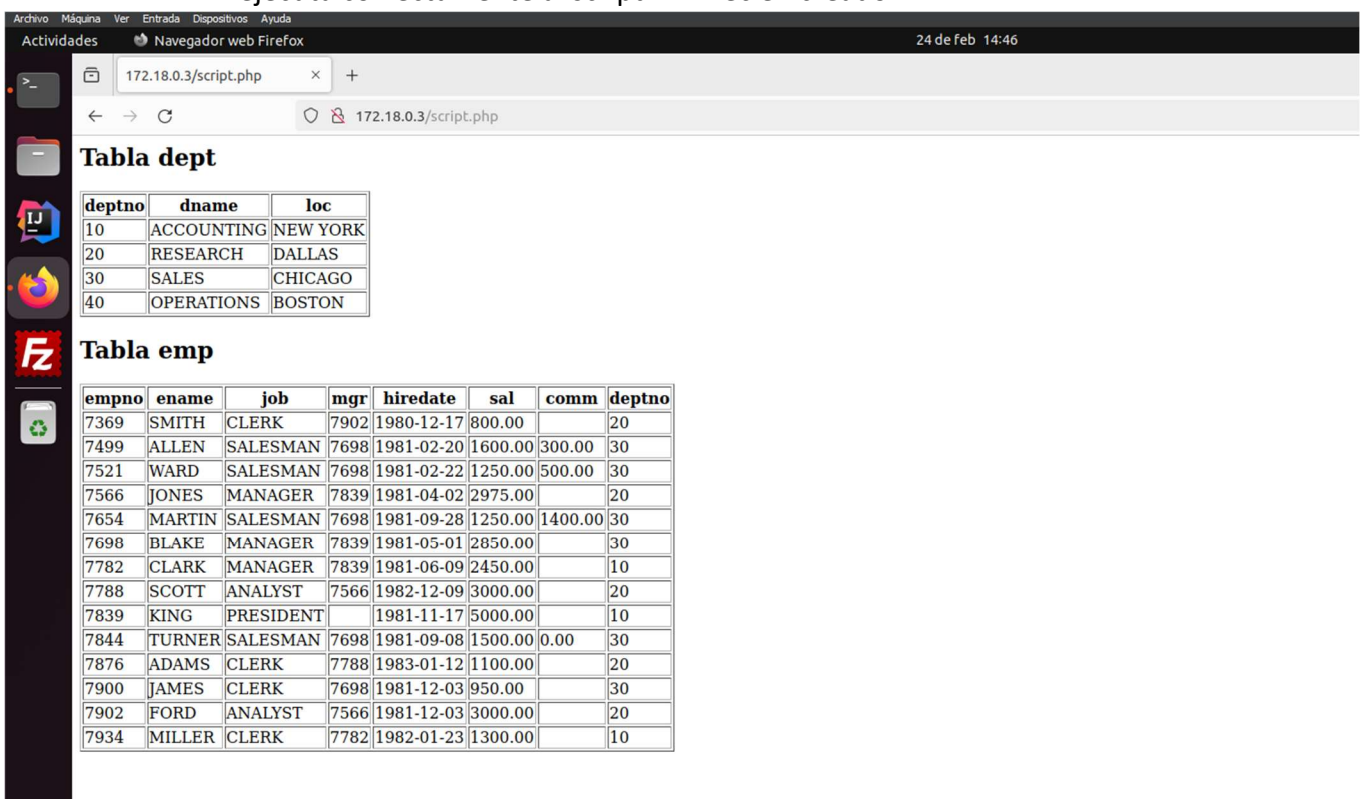
```
<?php
$servername = "172.18.0.2";
$username = "root";
$password = "admin";
$dbname = "SCOTT";
$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Conexión fallida: " . $conn->connect_error);
}
$sql_dept = "SELECT * FROM dept";
$result_dept = $conn->query($sql_dept);
$sql_emp = "SELECT * FROM emp";
$result_emp = $conn->query($sql_emp);
echo "<h2>Tabla dept</h2>";
if ($result_dept->num_rows > 0) {
    echo "<table border='1'><tr><th>deptno</th><th>dname</th><th>loc</th></tr>";
    while($row = $result_dept->fetch_assoc()) {
        echo
        "<tr><td>".$row["deptno"]."</td><td>".$row["dname"]."</td><td>".$row["loc"]."</td>
        </tr>";
    }
    echo "</table>";
} else {
    echo "0 resultados";
}
echo "<h2>Tabla emp</h2>";
if ($result_emp->num_rows > 0) {
    echo "<table
    border='1'><tr><th>empno</th><th>ename</th><th>job</th><th>mgr</th><th>hiredate</th><th>sal</th><th>comm</th><th>deptno</th></tr>";
    while($row = $result_emp->fetch_assoc()) {
        echo
```

```

" <tr><td>".$row["empno"]."</td><td>".$row["ename"]."</td><td>".$row["job"]."</td>
<td>".$row["mgr"]."</td><td>".$row["hiredate"]."</td><td>".$row["sal"]."</td><td>".$
row["comm"]."</td><td>".$row["deptno"]."</td></tr>";
}
echo "</table>";
} else {
echo "0 resultados";
}
$conn->close();
?>

```

6. Se comprueba desde la máquina anfitrión, usando un navegador, que se ejecuta correctamente al script PHP recién creado.



The screenshot shows a web browser window with the address bar displaying '172.18.0.3/script.php'. The page content includes two tables:

Tabla dept

deptno	dname	loc
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

Tabla emp

empno	ename	job	mgr	hiredate	sal	comm	deptno
7369	SMITH	CLERK	7902	1980-12-17	800.00		20
7499	ALLEN	SALESMAN	7698	1981-02-20	1600.00	300.00	30
7521	WARD	SALESMAN	7698	1981-02-22	1250.00	500.00	30
7566	JONES	MANAGER	7839	1981-04-02	2975.00		20
7654	MARTIN	SALESMAN	7698	1981-09-28	1250.00	1400.00	30
7698	BLAKE	MANAGER	7839	1981-05-01	2850.00		30
7782	CLARK	MANAGER	7839	1981-06-09	2450.00		10
7788	SCOTT	ANALYST	7566	1982-12-09	3000.00		20
7839	KING	PRESIDENT		1981-11-17	5000.00		10
7844	TURNER	SALESMAN	7698	1981-09-08	1500.00	0.00	30
7876	ADAMS	CLERK	7788	1983-01-12	1100.00		20
7900	JAMES	CLERK	7698	1981-12-03	950.00		30
7902	FORD	ANALYST	7566	1981-12-03	3000.00		20
7934	MILLER	CLERK	7782	1982-01-23	1300.00		10

7. Una vez que la imagen esté funcionando correctamente dentro del contenedor, se exportará y se subirá a la cuenta personal de DockerHub.

```

root@CLIENTE:/home/alumno# docker commit contiene_apache
zarritas/imagen_apache_php:mysql
sha256:8efc884a8a2359992aa7ec8ee79669e8f28293bdfd9df0d78293621667
063474
root@CLIENTE:/home/alumno# docker push zarritas/imagen_apache_php:mysql
The push refers to repository
[docker.io/zarritas/imagen_apache_php]
c1ce626febcb: Pushed
e238773396cf: Pushed
1a102d1cac2b: Mounted from zarritas/ubuntu

```


mysql: digest:

sha256:55bc0f649070616e85f18e82a6c79721659a101377d10b7a3adfd290ad
2cdc48 size: 949

Comprobaciones finales

1. Lista de contenedores con su información asociada.
2. Detener los dos contenedores, volver a arrancarlos y comprobar que se sigue ejecutando correctamente la mini-aplicación.

```
root@CLIENTE:/home/alumno# docker ps
```

CONTAINER ID	IMAGE	COMMAND NAMES	CREATED	STATUS
4d97dc9f99f9 second	mi_imagen_apache 0.0.0.0:8888->80/tcp, :::8888->80/tcp	"apache2ctl -D FOREG..." contiene_apache	16 minutes ago	Up 1
f2a20c9fcce3 hours	mysql 3306/tcp, 33060/tcp	"docker-entrypoint.s..." contiene_mysql	3 hours ago	Up 3

```
root@CLIENTE:/home/alumno# docker start contiene_apache
```

```
contiene_apache
```

```
root@CLIENTE:/home/alumno# docker start contiene_mysql
```

```
contiene_mysql
```

```
root@CLIENTE:/home/alumno# docker ps
```

CONTAINER ID	IMAGE	COMMAND NAMES	CREATED	STATUS
--------------	-------	---------------	---------	--------

ArchivoMáquinaVerEntradaDispositivosAyuda

ActividadesNavegador web Firefox24 de feb 14:46

172.18.0.3/script.php

172.18.0.3/script.php

Tabla dept

deptno	dname	loc
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

Tabla emp

empno	ename	job	mgr	hiredate	sal	comm	deptno
7369	SMITH	CLERK	7902	1980-12-17	800.00		20
7499	ALLEN	SALESMAN	7698	1981-02-20	1600.00	300.00	30
7521	WARD	SALESMAN	7698	1981-02-22	1250.00	500.00	30
7566	JONES	MANAGER	7839	1981-04-02	2975.00		20
7654	MARTIN	SALESMAN	7698	1981-09-28	1250.00	1400.00	30
7698	BLAKE	MANAGER	7839	1981-05-01	2850.00		30
7782	CLARK	MANAGER	7839	1981-06-09	2450.00		10
7788	SCOTT	ANALYST	7566	1982-12-09	3000.00		20
7839	KING	PRESIDENT		1981-11-17	5000.00		10
7844	TURNER	SALESMAN	7698	1981-09-08	1500.00	0.00	30
7876	ADAMS	CLERK	7788	1983-01-12	1100.00		20
7900	JAMES	CLERK	7698	1981-12-03	950.00		30
7902	FORD	ANALYST	7566	1981-12-03	3000.00		20
7934	MILLER	CLERK	7782	1982-01-23	1300.00		10

ANEXO I: Scripts de creación de tablas (DDL) e inserción de datos (DML).

```
mysql> CREATE DATABASE SCOTT;
Query OK, 1 row affected (0,03 sec)

mysql> USE SCOTT;
Database changed

DROP TABLE dept;
CREATE TABLE dept (
    deptno decimal(2,0),
    dname varchar(14),
    loc varchar(13),
    CONSTRAINT `PK_dept` PRIMARY KEY(deptno)
);

INSERT INTO dept VALUES (10,'ACCOUNTING','NEW YORK');
INSERT INTO dept VALUES (20,'RESEARCH','DALLAS');
INSERT INTO dept VALUES (30,'SALES','CHICAGO');
INSERT INTO dept VALUES (40,'OPERATIONS','BOSTON');

DROP TABLE emp;
CREATE TABLE emp (
    empno decimal(4,0),
    ename varchar(10),
    job varchar(9),
    mgr decimal(4,0),
    hiredate date,
    sal decimal(7,2),
    comm decimal(7,2),
    deptno decimal(2,0),
    CONSTRAINT `PK_emp` PRIMARY KEY (empno),
    CONSTRAINT `FK_emp_dept_deptno` FOREIGN KEY (deptno)
        REFERENCES dept (deptno),
    CONSTRAINT `FK_emp_emp_mgr` FOREIGN KEY (mgr)
        REFERENCES emp (empno)
);

INSERT INTO emp VALUES
('7839','KING','PRESIDENT',NULL,'1981-11-17','5000.00',NULL,'10'),
('7566','JONES','MANAGER','7839','1981-04-02','2975.00',NULL,'20'),
('7698','BLAKE','MANAGER','7839','1981-05-01','2850.00',NULL,'30'),
('7782','CLARK','MANAGER','7839','1981-06-09','2450.00',NULL,'10'),
('7788','SCOTT','ANALYST','7566','1982-12-09','3000.00',NULL,'20'),
('7902','FORD','ANALYST','7566','1981-12-03','3000.00',NULL,'20'),
('7499','ALLEN','SALESMAN','7698','1981-02-20','1600.00','300.00','30'),
('7521','WARD','SALESMAN','7698','1981-02-22','1250.00','500.00','30'),
('7654','MARTIN','SALESMAN','7698','1981-09-28','1250.00','1400.00','30'),
('7844','TURNER','SALESMAN','7698','1981-09-08','1500.00','0.00','30'),
('7900','JAMES','CLERK','7698','1981-12-03','950.00',NULL,'30'),
('7934','MILLER','CLERK','7782','1982-01-23','1300.00',NULL,'10'),
('7876','ADAMS','CLERK','7788','1983-01-12','1100.00',NULL,'20'),
('7369','SMITH','CLERK','7902','1980-12-17','800.00',NULL,'20');

DROP TABLE bonus;
```

```
CREATE TABLE bonus (  
    ename varchar(10),  
    job varchar(9),  
    sal decimal(7,2),  
    comm decimal(7,2)  
);  
  
DROP TABLE salgrade;  
CREATE TABLE salgrade (  
    grade decimal(2,0),  
    losal decimal(7,2),  
    hisal decimal(7,2),  
    CONSTRAINT `PK_salgrade` PRIMARY KEY (grade)  
);  
INSERT INTO salgrade VALUES (1,700,1200);  
INSERT INTO salgrade VALUES (2,1201,1400);  
INSERT INTO salgrade VALUES (3,1401,2000);  
INSERT INTO salgrade VALUES (4,2001,3000);  
INSERT INTO salgrade VALUES (5,3001,9999);  
  
COMMIT;
```