

# MySQL 导出数据

MySQL中你可以使用**SELECT...INTO OUTFILE**语句来简单的导出数据到文本文件上。

## 使用 SELECT ... INTO OUTFILE 语句导出数据

以下实例中我们将数据表 runoob\_tbl 数据导出到 /tmp/runoob.txt 文件中:

```
mysql> SELECT * FROM runoob_tbl
-> INTO OUTFILE '/tmp/runoob.txt';
```

你可以通过命令选项来设置数据输出的指定格式，以下实例为导出 CSV 格式：

```
mysql> SELECT * FROM passwd INTO OUTFILE '/tmp/runoob.txt'
-> FIELDS TERMINATED BY ',' ENCLOSED BY '"'
-> LINES TERMINATED BY '\r\n';
```

在下面的例子中，生成一个文件，各值用逗号隔开。这种格式可以被许多程序使用。

```
SELECT a,b,a+b INTO OUTFILE '/tmp/result.text'
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\n'
FROM test_table;
```

### SELECT ... INTO OUTFILE 语句有以下属性:

- LOAD DATA INFILE是SELECT ... INTO OUTFILE的逆操作，SELECT句法。为了将一个数据库的数据写入一个文件，使用SELECT ... INTO OUTFILE，为了将文件读回数据库，使用LOAD DATA INFILE。
- SELECT...INTO OUTFILE 'file\_name'形式的SELECT可以把被选择的行写入一个文件中。该文件被创建到服务器主机上，因此您必须拥有FILE权限，才能使用此语法。
- 输出不能是一个已存在的文件。防止文件数据被篡改。
- 你需要有一个登陆服务器的账号来检索文件。否则 SELECT ... INTO OUTFILE 不会起任何作用。
- 在UNIX中，该文件被创建后是可读的，权限由MySQL服务器所拥有。这意味着，虽然你就可以读取该文件，但可能无法将其删除。

## 导出表作为原始数据

**mysqldump** 是 mysql 用于转存储数据库的实用程序。它主要产生一个 SQL 脚本，其中包含从头重新创建数据库所必需的命令 CREATE TABLE INSERT 等。

使用 **mysqldump** 导出数据需要使用 **--tab** 选项来指定导出文件指定的目录，该目标必须是可写的。

以下实例将数据表 runoob\_tbl 导出到 /tmp 目录中：

```
$ mysqldump -u root -p --no-create-info \  
            --tab=/tmp RUNOOB runoob_tbl \  
password *****
```

## 导出 SQL 格式的数据

导出 SQL 格式的数据到指定文件，如下所示：

```
$ mysqldump -u root -p RUNOOB runoob_tbl > dump.txt \  
password *****
```

以上命令创建的文件内容如下：

```
-- MySQL dump 8.23  
--  
-- Host: localhost    Database: RUNOOB  
-----  
-- Server version    3.23.58  
  
--  
-- Table structure for table `runoob_tbl`  
--  
  
CREATE TABLE runoob_tbl (  
  runoob_id int(11) NOT NULL auto_increment,  
  runoob_title varchar(100) NOT NULL default '',  
  runoob_author varchar(40) NOT NULL default '',  
  submission_date date default NULL,  
  PRIMARY KEY (runoob_id),  
  UNIQUE KEY AUTHOR_INDEX (runoob_author)  
) TYPE=MyISAM;  
  
--  
-- Dumping data for table `runoob_tbl`  
--  
  
INSERT INTO runoob_tbl  
VALUES (1,'Learn PHP','John Poul','2007-05-24');  
INSERT INTO runoob_tbl  
VALUES (2,'Learn MySQL','Abdul S','2007-05-24');  
INSERT INTO runoob_tbl  
VALUES (3,'JAVA Tutorial','Sanjay','2007-05-06');
```

如果你需要导出整个数据库的数据，可以使用以下命令：

```
$ mysqldump -u root -p RUNOOB > database_dump.txt
password *****
```

如果需要备份所有数据库，可以使用以下命令：

```
$ mysqldump -u root -p --all-databases > database_dump.txt
password *****
```

--all-databases 选项在 MySQL 3.23.12 及以后版本加入。

该方法可用于实现数据库的备份策略。

## 将数据表及数据库拷贝至其他主机

如果你需要将数据拷贝至其他的 MySQL 服务器上, 你可以在 mysqldump 命令中指定数据库名及数据表。

在源主机上执行以下命令，将数据备份至 dump.txt 文件中：

```
$ mysqldump -u root -p database_name table_name > dump.txt
password *****
```

如果完整备份数据库，则无需使用特定的表名称。

如果你需要将备份的数据库导入到MySQL服务器中，可以使用以下命令，使用以下命令你需要确认数据库已经创建：

```
$ mysql -u root -p database_name < dump.txt
password *****
```

你也可以使用以下命令将导出的数据直接导入到远程的服务器上，但请确保两台服务器是相通的，是可以相互访问的：

```
$ mysqldump -u root -p database_name \
| mysql -h other-host.com database_name
```

以上命令中使用了管道来将导出的数据导入到指定的远程主机上。

← MySQL 及 SQL 注入

MySQL 导入数据 →



2 篇笔记



写笔记



将指定主机的数据库拷贝到本地

如果你需要将远程服务器的数据拷贝到本地，你也可以在 `mysqldump` 命令中指定远程服务器的IP、端口及数据库名。

在源主机上执行以下命令，将数据备份到 `dump.txt` 文件中：

请确保两台服务器是相通的：

```
mysqldump -h other-host.com -P port -u root -p database_name > dump.txt  
password ****
```

CarolLi 2年前 (2017-08-21)



在写出的时候会出现The MySQL server is running with the `--secure-file-priv` option so it cannot execute this statement的错误解决方法：

出现这个错误是因为没有给数据库指定写出文件的路径或者写出的路径有问题。

首先使用下面的命令 `show variables like '%secure%'`；查看数据库的存储路径。如果查出的 `secure_file_priv` 是 `null` 的时候就证明在 `my.ini` 文件里面没有配置写出路径。

这时候就可以在 `mysql.ini` 文件的 `[mysqld]` 代码下增加 `secure_file_priv=E:/TEST` 再重启 `mysql` 就可以了。然后在导出的地址下面写上刚才配置的这个地址 eg: `select * from tb_test into outfile "E:/TEST/test.txt"`；就可以了。

且听风吟 2周前 (03-06)