**原理：**

1. Master主服务器将对数据的操作记录到二进制日志中（Binary log），MySQL将事务串行（交叉执行）的写入二进制日志；
2. Slave从服务器将二进制日志（Binary log）copy到中继日志当中（Relay log）。首先，slave开始一个工作线程——I/O线程。

* slave开始一个工作线程——I/O线程。I/O线程在master上打开一个普通的连接；
* 开始binlog dump process；
* 如果binlog dump process已经跟上master，它会睡眠并等待master产生新的事件。
* I/O线程将这些事件写入中继日志；
* SQL slave thread（SQL从线程）从中继日志读取事件，并重放其中的事件而更新slave的数据，使其与master中的数据一致

**实现步骤：**

1. 主数据库配置文件（my.cnf）修改

|  |
| --- |
| # 服务的唯一编号  server-id = 1  # 开启mysql binlog功能  log-bin = mysql-bin  # binlog记录内容的方式，记录被操作的每一行  binlog\_format = ROW  # 减少记录日志的内容，只记录受影响的列  binlog\_row\_image = minimal  # 指定需要复制的数据库名为jgyw  binlog-do-db = jgyw |

1. 重启主数据库服务

|  |
| --- |
| service mysqld restart |

1. 创建从库同步数据的账号

|  |
| --- |
| create user jgyw;  grant replication slave on \*.\* to 'jgyw'@'192.168.197.136' identified by 'jgyw@123';  flush privileges;  注：grant replication slave on \*.\* to '**账号**'@'**ip**' identified by '**jgyw@123**';（这里ip为从库地址） |

1. 查看主库的状态：

命令：

|  |
| --- |
| show master status\G; |

返回信息：

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  File: mysql-bin.000002  Position: 2380  Binlog\_Do\_DB: jgyw  Binlog\_Ignore\_DB:  Executed\_Gtid\_Set:  1 row in set (0.00 sec) |

1. 从数据库配置文件（my.cnf）修改

|  |
| --- |
| # 服务的唯一编号  server-id = 2  # 开启mysql binlog功能  log-bin = mysql-bin  # binlog记录内容的方式，记录被操作的每一行  binlog\_format = ROW  # 减少记录日志的内容，只记录受影响的列  binlog\_row\_image = minimal  # 指定需要复制的数据库名为jgyw  replicate-do-db = jgyw |

1. 从数据库重启

|  |
| --- |
| service mysqld restart |

1. 执行同步命令

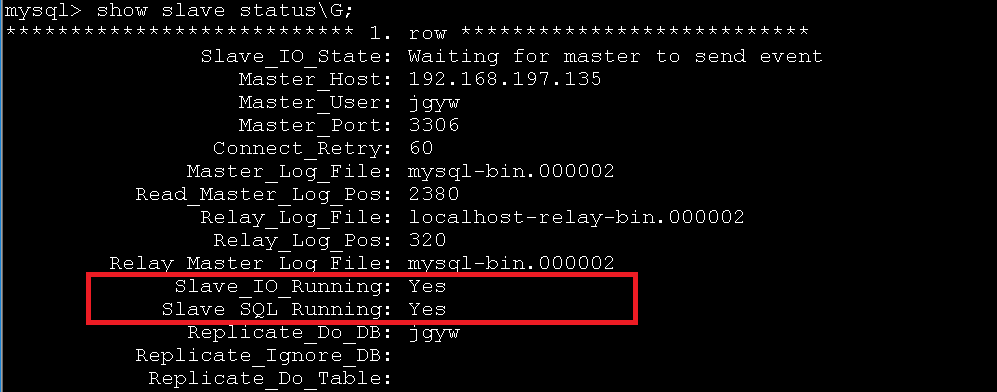
|  |
| --- |
| # 设置主服务器ip，同步账号密码，同步位置  change master to master\_host='192.168.197.135',master\_user='jgyw',master\_password='jgyw@123',master\_log\_file='mysql-bin.000002',master\_log\_pos=2380;  # 开启同步功能  start slave; |

1. 查看从库的状态

命令：

|  |
| --- |
| show slave status\G; |

返回信息



**注：Slave\_IO\_Running和Slave\_SQL\_Running的状态都为Yes时，说明从库配置成功。**