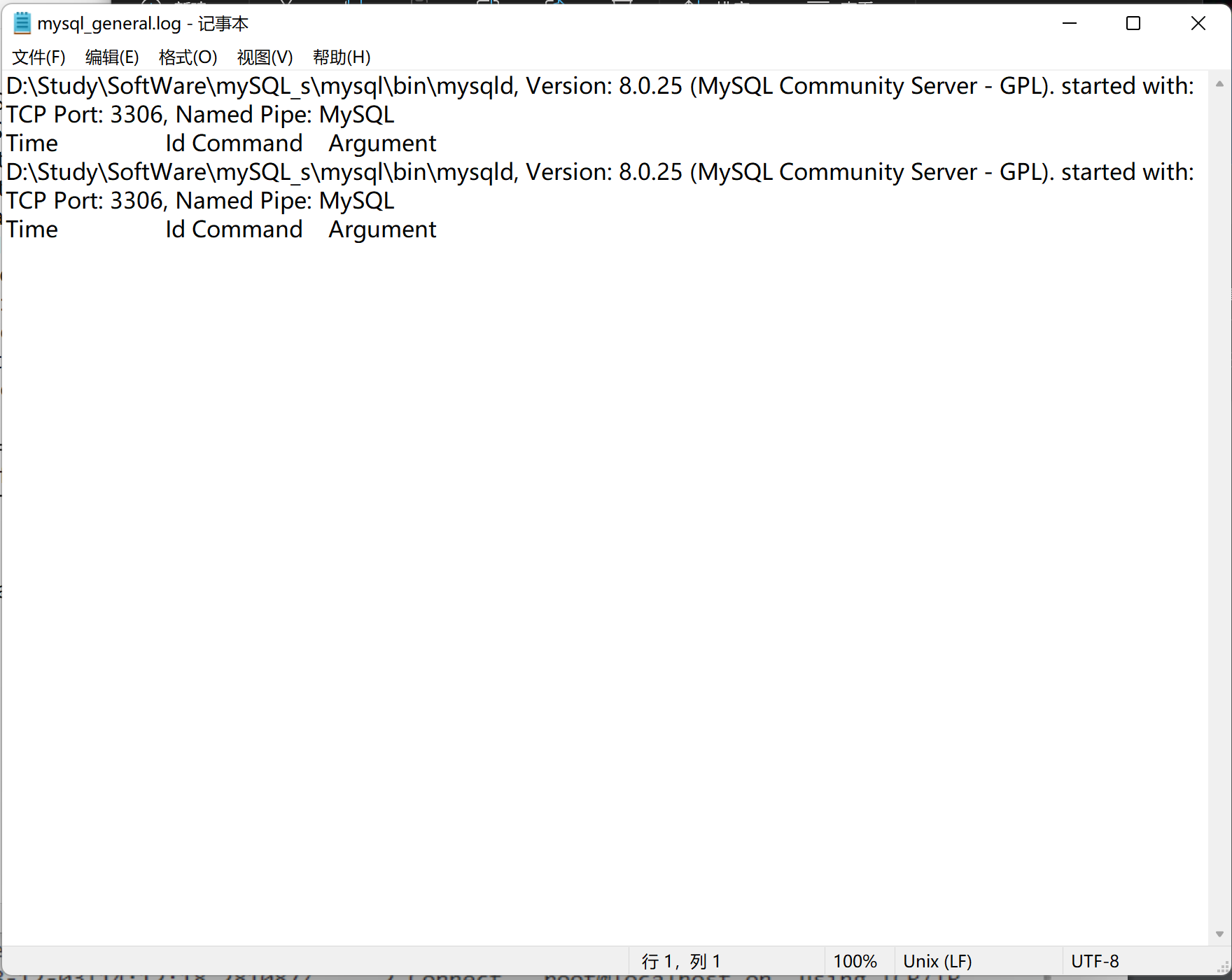
**设计性实验十一**

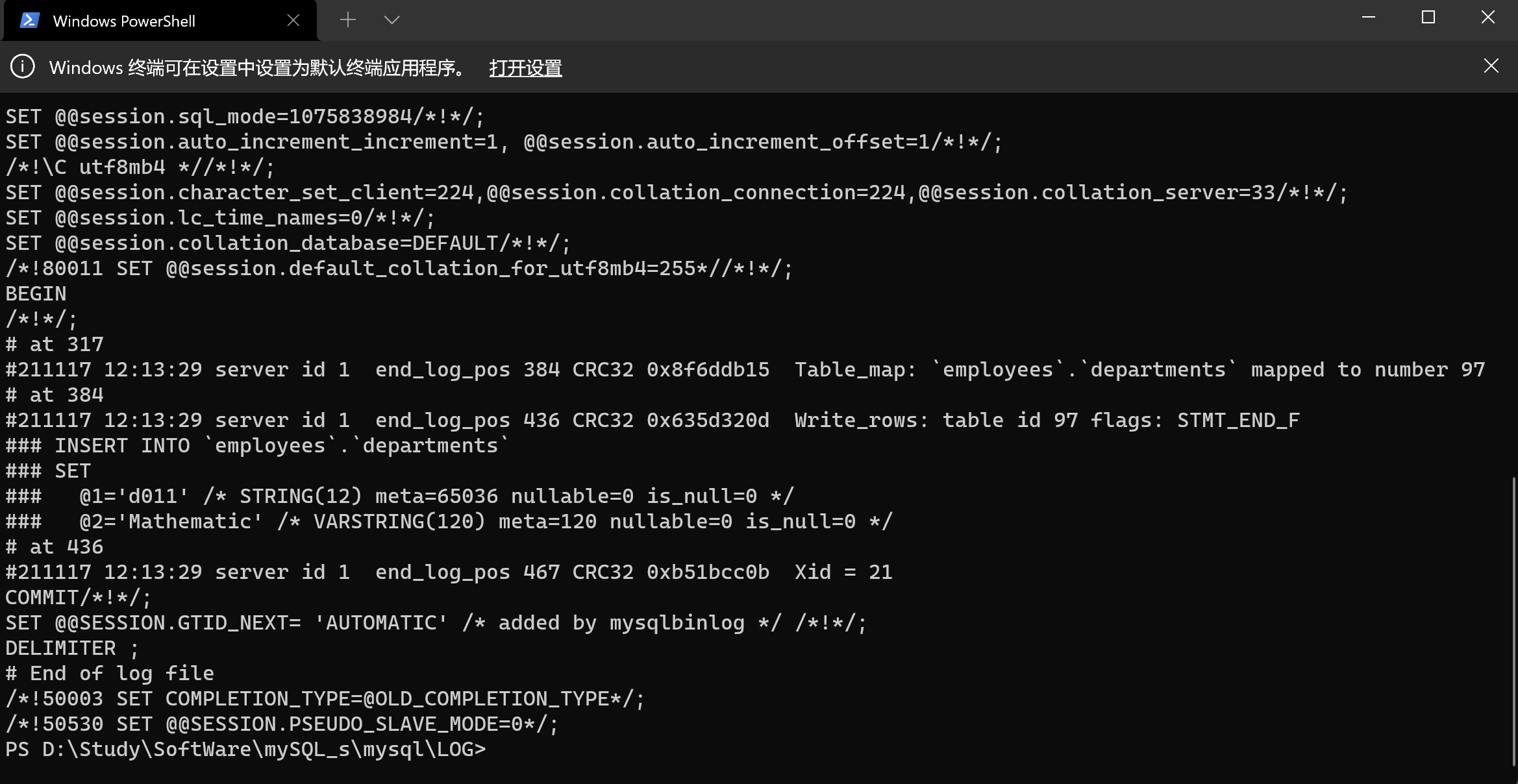
**一、日志管理**

1. 开启通用日志，将其输出形式改为TABLE,并截图通用日志内容。

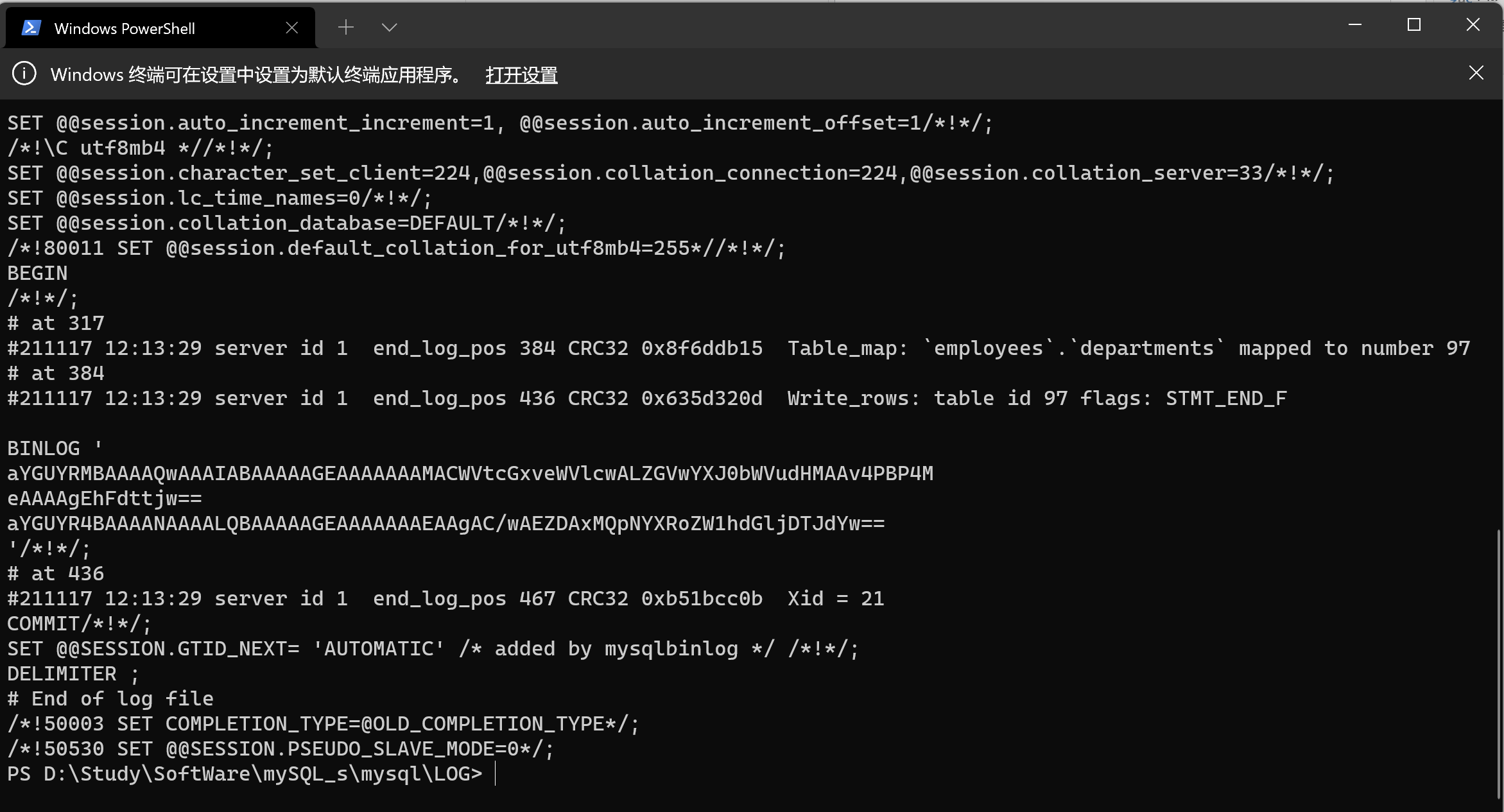


1. 截图比较二进制日志输出格式 ROW和STATEMENT,比较其不同。

ROW输出格式



Statement格式



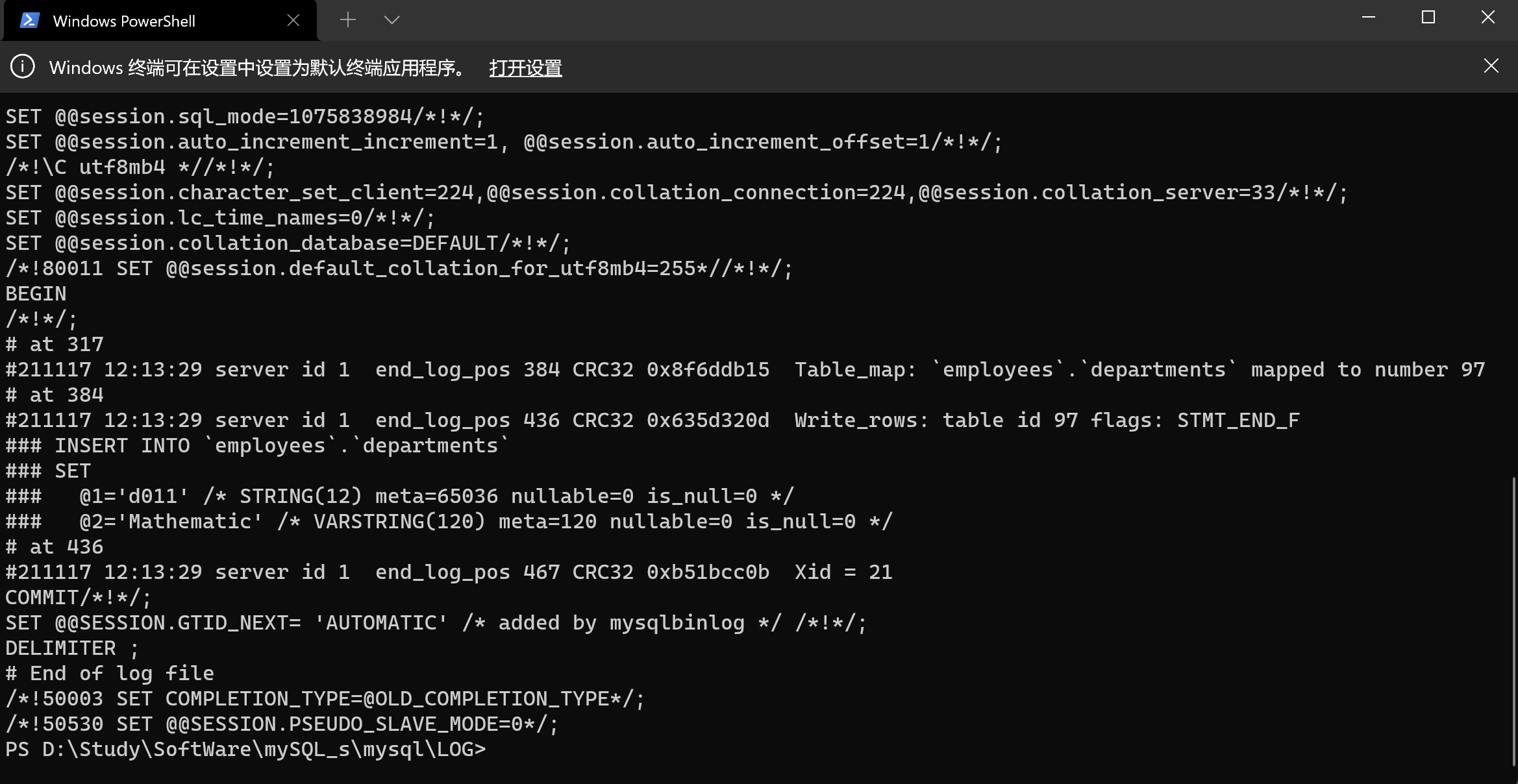
statement:基于SQL语句的模式。

row:基于行的模式，记录的是行的变化，很安全。但是binloq会比其他两种模式大很多，在一些大表中清除大量数据时在binlog中会生成很多条语句，可能导致从库延迟变大。

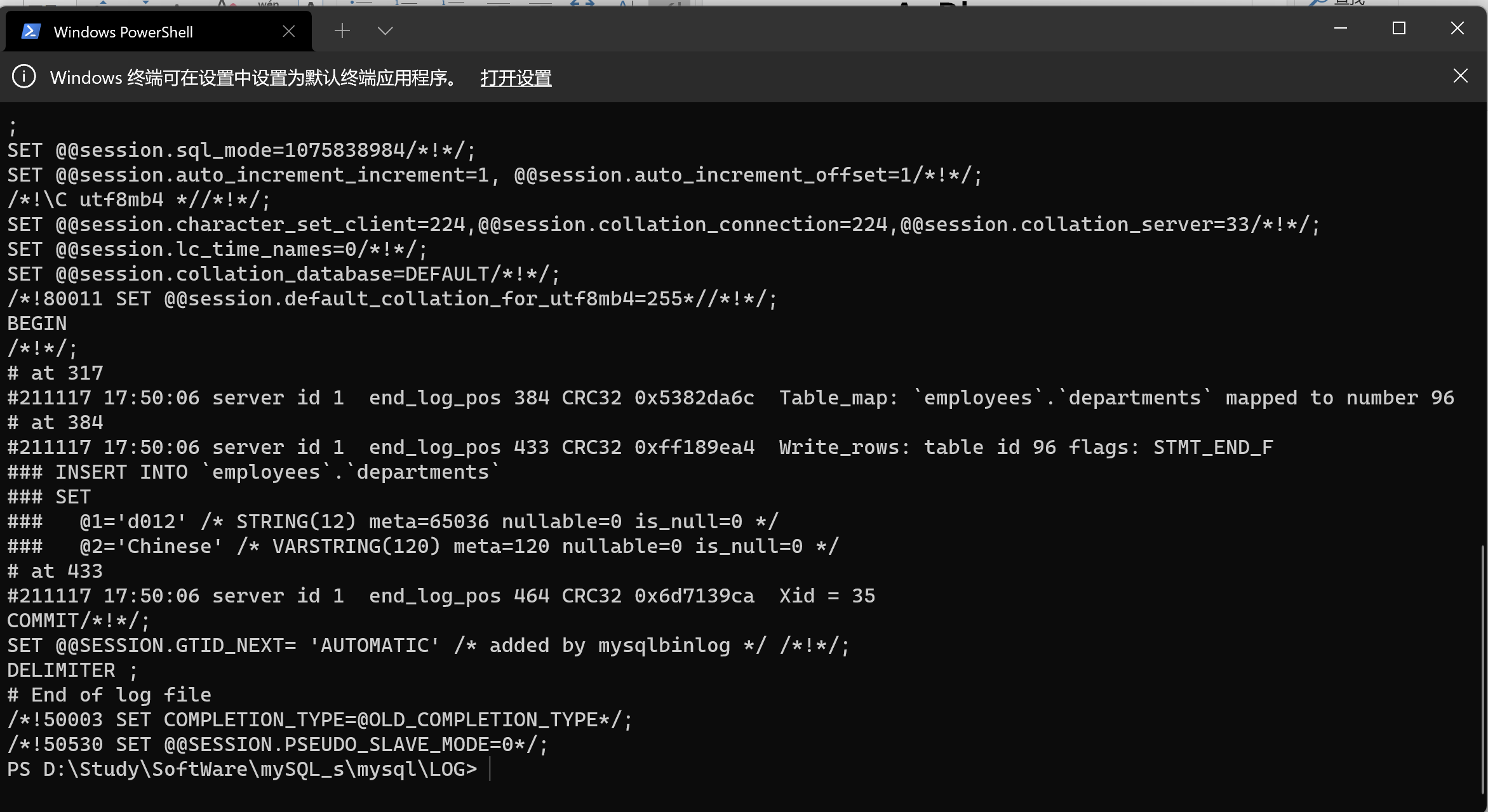
1. 将二进制日志输出格式改为ROW,比较binlog\_row\_image为FULL与MINIMAL的不同

mysqlbinlog --no-defaults -v -v --base64- output=DECODE-ROWS Filename.number

FULL：



MINIMAL

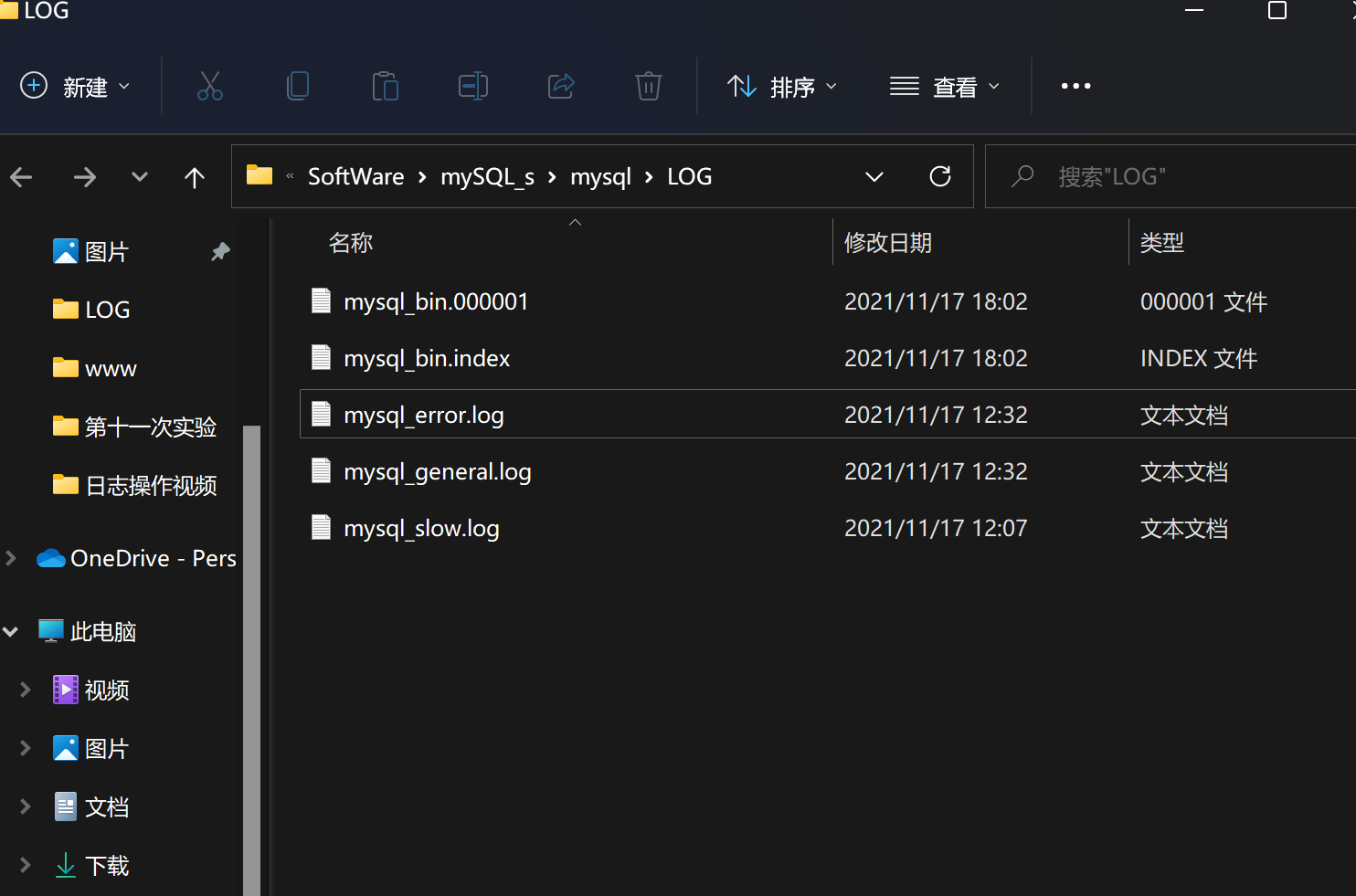


FULL记录每一行的变更，minimal只 记录影响后的行的字段

1. 更新二进制日志，删除二进制日志。

更新二进制文件：flush logs；

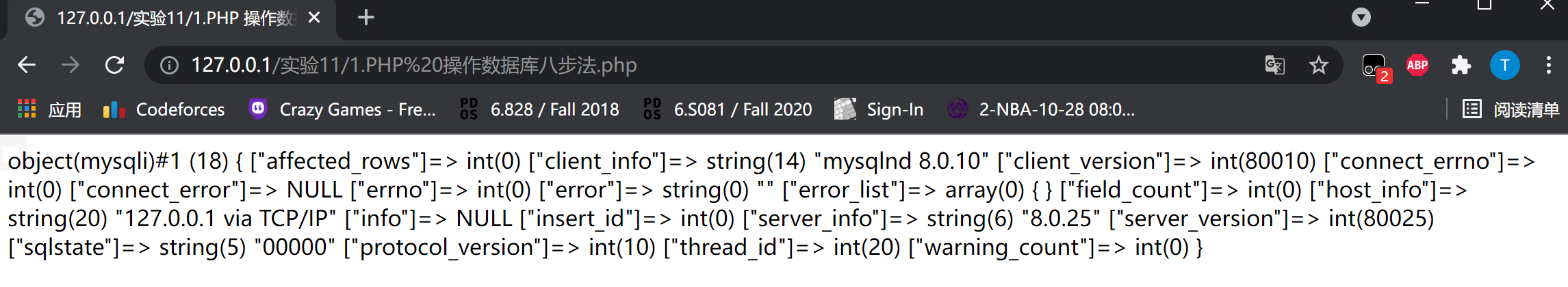
删除二进制文件：reset master；



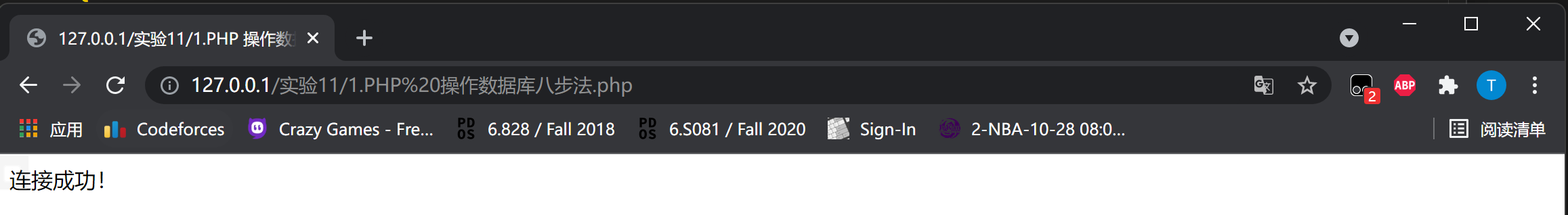
**二 嵌入式SQL**

1. 实践PHP连接数据库，进行查询更新操作。
   1. 连接数据库

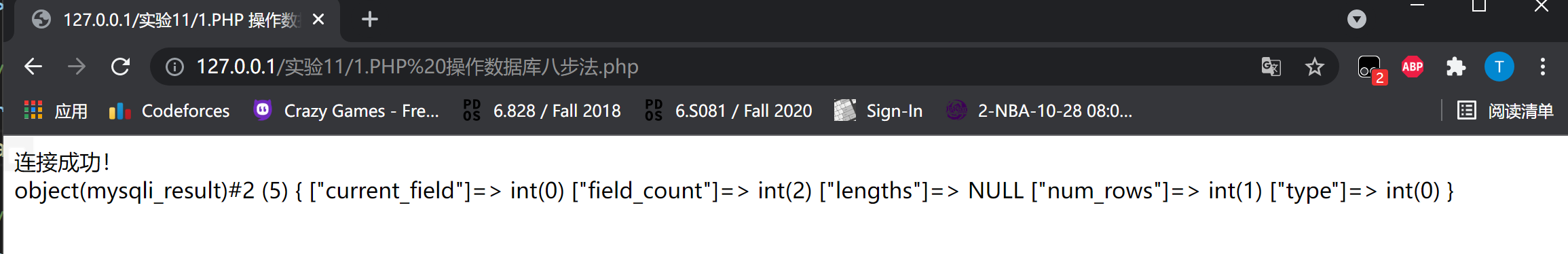
建立连接



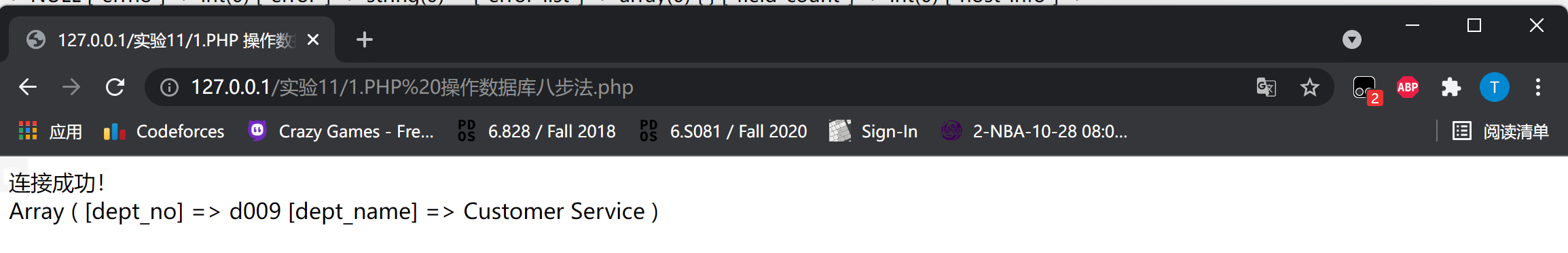
连接成功



选择数据库，执行sql指令

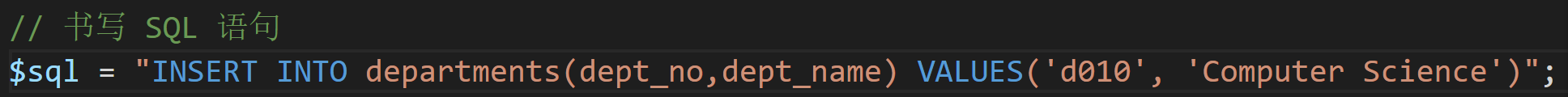


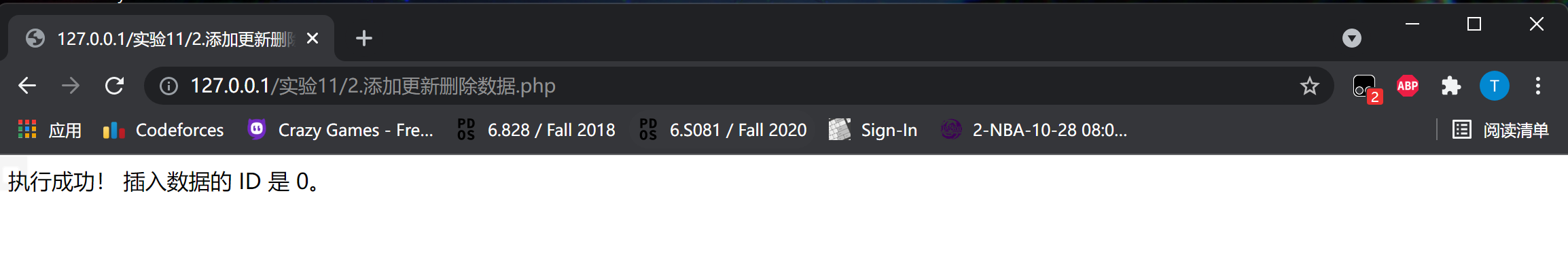
输出



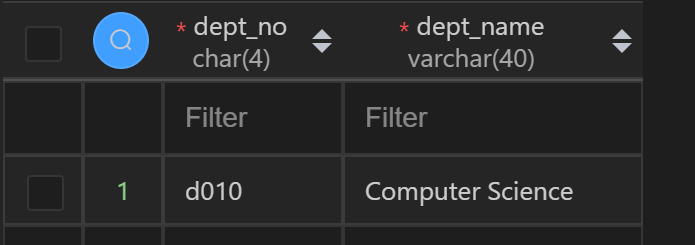
* 1. 添加更新删除操作

插入一行数据VALUES('d010', 'Computer Science')





插入成功



* 1. 查询和数据解析

$sql = "SELECT \* FROM departments";

查询成功

