

数据库工程作业

要求:

- 根据第一次作业第一题的设计(可调整题目), 完成一个小型的数据库信息管理系统(或部分功能), 并填写工程作业报告; 程序和报告请在规定时间之内上传。
 - 开发模式(B/S 或 C/S)、开发高级语言任选, 后台数据库使用大型数据库管理系统(SQL Server、Oracle 等), 不要使用桌面数据库。
 - 报告中所列举的四种操作, 每种操作举一个例子即可。
 - 作业成绩按照报告中的标准评分, 程序只实现报告中涉及的部分即可。
-

工程作业报告

1. 项目信息

学号	1110568	姓名	郭艺璇	专业	计算机科学与技术
项目名称	电影院网络管理系统				
必备环境	Windows 7, My SQL, Eclipse , NetBeans IDE 7.3.1				

2. 系统配置 (10 分)

说明		(2分) 请说明系统配置情况(后台数据库, 高级语言); (8分) 请使用连接串连接高级语言和数据库, 并分析字符串的各个部分。			
配 置 步 骤 2 分	DBMS	数据库由 phpMyAdmin 建立			
高 级 语 言	前台界面由 PHP 语言编写, 采用 NetBeans IDE 7.3.1 平台进行开发。				
连接串 分析	序号	名称	功能说明		取值
	1	\$DB_SERVER	本地服务器名称		localhost
	2	\$DB_NAME	数据库名称		theaters
	3	\$DB_USER	数据库授权访问者		gyx
(6)	4	\$DB_PASS	访问者密码		gyx
连接串代 码(截屏) 2(分)	定义在 setup.php 文件中的变量				

	<pre>\$DB_SERVER="localhost"; \$DB_NAME="theaters"; \$DB_USER="gyx"; \$DB_PASS="gyx";</pre>
备注	连接串代码： <pre>require 'setup.php'; \$connect=mysql_connect (\$DB_SERVER,\$DB_USER,\$DB_PASS) or die("LINK FAIL sql") \$db=mysql_select_db(\$DB_NAME,\$connect) or die("SELECTION FAIL sql");</pre>

3. 数据库设计 (20 分)

说明	(10 分) 按照数据表的创建顺序，依次给出所涉及数据表的信息，其中参照字段以“(字段 1，字段 2，……，字段 n)”的形式给出，被参照字段以“表名(字段 1，字段 2，……，字段 n)”的形式给出； (10 分) 一般 DBMS 都可以为数据库生成关系图，请将该图片截屏并粘贴到表格中。				
数据表 (10)	创建 顺序	数据表名称	主键	参照属性	被参照表及属性
	1	theater	thrid	无	无
	2	movies	movid	无	无
	3	price	priceid	无	无
	4	display	playid	movid	movies(movid)
				thrid	theater(thrid)
				priceid	price(priceid)
	5	level	levelid	无	无
	6	customer	cusid	levelid	level(levelid)
	7	orders	orderid	playid	display(playid)
关系图 (10)	(截屏)				

	<pre> erDiagram theater --o movies : "movies" theater --o display : "display" theater --o price : "price" movies --o display : "display" price --o display : "display" display --o orders : "orders" customer --o orders : "orders" level --o customer : "customer" </pre>
备注	level 表并未在作业中涉及，其功能暂时没有进行拓展

1. 含有事务应用的删除操作（15分）

说明	<p>(1 分) 简要说明该操作所要完成的功能；</p> <p>(2 分) 该操作会涉及的表（必须含有两张或两张以上的关系表，同时以“表名”的形式给出）</p> <p>(1 分) 表连接涉及字段描述（描述方式为“表 1. 属性=表 2. 属性”）</p> <p>(1 分) 删除条件涉及的字段描述(以“表名.属性=?”形式给出)</p> <p>(6 分) 实现该操作的关键代码（高级语言、SQL），截图即可；（其中如果删除语句中不包含任何形式的事务应用将扣除 3 分）</p> <p>(4 分) 如何执行该操作，按所述方法能够正常演示程序则给分。</p>
功能描述 (1 分)	通过选择电影的名称进行删除电影的操作。 若将要删除的电影包含放映信息，则列出放映信息，提示用户是否删除。用户决定取消则事物回滚；用户决定继续删除则将该电影信息,放映信息和相关的订单信息都删除。
涉及的表 (2 分)	电影信息表 movies, 电影放映信息表 display, 订单信息表 orders, 作业 4 当中创建的视图 playinfo.
表连接涉及字段 (1 分)	movies.movid=display.movid, display.playid=orders.playid

	字段	规则
删除 条件 字段 描述 (1 分)	movies.movid	若 movid 在放映信息表 display 中, 则提示用户存在放映信息, 是否继续删除, 若用户决定放弃则事物回滚, 不能删除, 若用户继续删除则进行删除。
	display.playid	若 playid 所对应的信息当中包含要删除的 movie 信息, 则根据用户决定是否删除, 用户决定删除则在删除电影信息前删除放映信息。
	orders.orderid	若 orderid 所对应的信息当中包含要删除的 playid 信息, 则根据用户决定是否删除, 用户决定删除则在删除电影放映信息前删除预订信息。
代码 (6 分)	<pre>//含有事务应用的删除操作 //删除电影的信息, 如果该电影有对应的订单, 则将订单也删除 function delform()//删除信息 { require "setup.php"; \$sql="select* from \$MVE"; \$connect=mysql_connect(\$DB_SERVER,\$DB_USER,\$DB_PASS)or die("LINK FAIL"); \$db=mysql_select_db(\$DB_NAME,\$connect)or die("SELECTION FAIL"); \$query=mysql_query(\$sql,\$connect)or die("EXECUTION FAIL"); if(\$query) { print "

<select name=selmovie>"; while(\$list=mysql_fetch_array(\$query)) { print "<option value=\$list[movid]>\$list[movname]</option>"; } print "</select><input type=submit name=dodel value=Choose_a_na } }</pre>	

```
function dodel($selname)//删除电影信息
{
    //print $selname;
    require "setup.php";//连接数据库的信息
    $connect=mysql_connect ($DB_SERVER, $DB_USER, $DB_PASS) or die("Connecting failed");
    $db=mysql_select_db ($DB_NAME, $connect) or die("Selecting failed");
    mysql_query(' START TRANSACTION');

    $sql="select * from $DPL where movid='$selname'";
    $query=mysql_query($sql,$connect) or die("Executing failed finding in dis");
    $list=mysql_fetch_array($query);
    //PRINT "$list[playid]";
    if($list[playid])//删除的电影存在与放映信息表display中，判断是否进行删除
    {
        print "<p id=\"thr_table\"><th><center>!!!!There are display information
        //显示订单，提醒用户该电影有放映信息！！
        print "<table id=\"thr_table\">";
        print "<tr class=\"alt\">
            <th>Moviename</th><th>Theater</th><th>Price</th><th>Type</th><th>Time</th><th>Intro</th></tr>";

        while($listlist=mysql_fetch_array($query))
        {
            print " <tr ><td>$listlist[movname]</td>
                  <td>$listlist[thrname]</td>
                  <td>$listlist[thrprice]</td>
                  <td>$listlist[pricetype]</td>
                  <td>$listlist[playtime]</td>
                  <td>$listlist[intro]</td>
            </tr> ";
        }
        print "</table>";
        //显示出该电影，用submit的方式提交，若用户点击则跳转到casadeddel函数进行级
        print "<br><select name=selmovie>";
        print "<option value=$selname>$list[movname]</option>";
        print "</select><input type=submit name=casadedel value=Sure_Deletion>";
        //撤销删除，跳转到事物回滚roll_back函数
        print "<input type=submit name=roll_back value=Canseling_Deletion><br>";
    }
}
```

```
else//放映信息里面没有要删除的电影，可以直接删除
{
    $sql="delete from $MVE where movid='".$selname'";//状态符合要求，可以
    $query1=mysql_query($sql,$connect)or die("Executing failed 2");
    if(!$query1)
    {
        mysql_query('ROLLBACK');
        print"deleting error-->rollback 2";
    }
    else { mysql_query('COMMIT');
        PRINI "<br>Deletion Movie $selname OK!!!</br>";
    }
}
```

roll_back 函数，进行事物回滚：

```
function roll_back()
{
    require 'setup.php';

    $connect=mysql_connect($DB_SERVER,$DB_USER,$DB_PASS)or die("Connecting fail");
    $db=mysql_select_db($DB_NAME,$connect)or die("Selecting failed");
    mysql_query('ROLLBACK',$connect);
    print"<br><p id=\"$thr_table\"><th>!!!!Deletion is given up!!!!</th></p>";
```

casadedele 函数，进行级联删除：

```

function casadedel($selname)
{ //print "casadele!!!!!! $selname !!!!!!!";
    require"setup.php";//连接数据库的信息
    $connect=mysql_connect("$DB_SERVER,$DB_USER,$DB_PASS")or die("Connecting failed");
    $db=mysql_select_db("$DB_NAME",$connect)or die("Selecting failed");
    //删除与该电影有关的订单消息
    $sql="delete from $ORD where playid=
        (select playid from $DPL where movid='$selname')";
    //print $sql;删除与该电影有关的放映消息
    $queryd=mysql_query($sql,$connect)or die("Executing failed 3");
    $sql2="delete from $DPL where movid='$selname'";
    //删除电影信息
    $queryd=mysql_query($sql2,$connect)or die("Executing failed 4");
    $sql="delete from $MVE where movid='$selname'";//状态符合要求，可以进行
    $queryl=mysql_query($sql,$connect)or die("Executing failed 2");
    if(!$query1||!$queryd){
        mysql_query("ROLLBACK");
        print"deleting error-->rollback 2";
    }
    else {
        mysql_query("COMMIT");
        print "<br><p id=\"$thr_table\"><th>Deletion $selname OK!</th>
    }
}

```

删除操作的过程，触发消息与调用函数：

```

$del=$_POST["del"];
$dodel=$_POST["dodel"];
$casadedel=$_POST["casadedel"];
$roll_back=$_POST["roll_back"];

else if($del){delform();}
else if($dodel||$casadedel)
{
    $selname=$_POST["selmovie"];
    if($dodel){//print "$selname";
        dodel($selname);
    }
    else{//print "$selname";
        casadedel ($selname);
    }
}
else if($roll_back){ roll_back();}


```

程序
演示

点击 **DEL_MOVIE** 按钮，触发删除事务：

(4
分)

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

!!!!There are display informations contains this movie!!!!

Moviename	Theater	Price	Type	Time	Introduction
Lost in Thailand	Happy Theater	120	IMAX	2014-01-01 18:00:00	new movie!!!

点击 Canseling_Deletion 按钮，调用 roll_back 函数，事务回滚，提示回滚事务成功：

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

!!!!Deletion is given up!!!!

若点击 Sure_Deletion 按钮，则调用 casadedel 函数进行级联删除：

THEATER MANAGER

*****Click for desired operation*****

ADD_MOVIE

DEL_MOVIE

MOD_PRICE

LIST_PLAY

ADD_PLAY

Deletion 1006 OK!

备注

2. 触发器控制下的添加操作 (20分)

说明 (1分)	(1分) 简要说明该操作所要完成的功能; (2分) 简要说明该触发器所要完成的功能 (1分) 该操作会涉及的表 (以“表名”的形式给出)。 (2分) 该操作输入数据以及输入数据应该满足的条件, 如: 数值范围、是否为空; (6分) 实现该操作的关键代码 (高级语言、SQL), 截图即可; (8分) 如何执行该操作, 按所述方法能够正常演示程序则给分。												
功能描述 (1分)	当添加一条放映电影的信息的时候, 检查添加的输入信息合法性, 并检查电影表中是否有该电影信息, 没有则更新电影表。												
触发器描述 (2分)	当添加一条新的放映信息到 display 表中, 检查电影表 movies 中有没有该电影信息, 若没有通过触发器添加放映信息的电影相关信息到电影表中。												
涉及的表 (1分)	放映信息表 display 电影表 movies 电影院表 theaters												
输入数据 (2分)	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 2px;">字段</th> <th style="text-align: center; padding: 2px;">规则</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">movid</td> <td style="padding: 2px;">两者不能同时为空。 若只填了一项信息, 则判断该电影信息在不在 movies 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。</td> </tr> <tr> <td style="padding: 2px;">movname</td> <td style="padding: 2px;">若两项都填了, 判断该电影信息是否与 movies 表中一项信息对应, 是则进行插入操作, 否则判断该电影信息的 movid 和 movname 是不是都不在 movies 表里, 是则将该电影信息插入 movies 表里头, 允许插入放映信息到 display。</td> </tr> <tr> <td style="padding: 2px;">third</td> <td style="padding: 2px;">判断该电影院信息在不在 theater 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。</td> </tr> <tr> <td style="padding: 2px;">priceid</td> <td style="padding: 2px;">与 third 判断条件类似, 若 price 表中没有改价格信息, 不允许插入。</td> </tr> <tr> <td style="padding: 2px;">playtime</td> <td style="padding: 2px;">不能为空</td> </tr> </tbody> </table>	字段	规则	movid	两者不能同时为空。 若只填了一项信息, 则判断该电影信息在不在 movies 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。	movname	若两项都填了, 判断该电影信息是否与 movies 表中一项信息对应, 是则进行插入操作, 否则判断该电影信息的 movid 和 movname 是不是都不在 movies 表里, 是则将该电影信息插入 movies 表里头, 允许插入放映信息到 display。	third	判断该电影院信息在不在 theater 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。	priceid	与 third 判断条件类似, 若 price 表中没有改价格信息, 不允许插入。	playtime	不能为空
字段	规则												
movid	两者不能同时为空。 若只填了一项信息, 则判断该电影信息在不在 movies 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。												
movname	若两项都填了, 判断该电影信息是否与 movies 表中一项信息对应, 是则进行插入操作, 否则判断该电影信息的 movid 和 movname 是不是都不在 movies 表里, 是则将该电影信息插入 movies 表里头, 允许插入放映信息到 display。												
third	判断该电影院信息在不在 theater 表中, 在则进行插入放映信息操作, 否则报错, 不能插入。												
priceid	与 third 判断条件类似, 若 price 表中没有改价格信息, 不允许插入。												
playtime	不能为空												
插入操作源码 (3分)	(截屏) 首先判断电影院和价格的 id 是否填写正确:												

```

$sql="select * from $DPL where $DPL.playid>=all(select playid from $DPL)";
$connect=mysql_connect($DB_SERVER,$DB_USER,$DB_PASS)or die("con fail 1");
$db=mysql_select_db($DB_NAME,$connect)or die("sel fail 1");
//print $sql;
$query=mysql_query($sql,$connect)or die("sql fail???");
$list=mysql_fetch_array($query);
$id= $list[playid]+1;
$sqltest="select * from $IHR where thrid=$thr_id";//判断电影院存在否?
$querytest= mysql_query($sqltest,$connect) or die("sqltest fail");
$listtest= mysql_fetch_array($querytest);
$thridtest=$listtest[thrname];
$sqltest2="select * from $PRI where priceid=$price_id";//判断价格存在否?
$querytest2= mysql_query($sqltest2,$connect) or die("sqltest fail2");
$listtest2= mysql_fetch_array($querytest2);
$thridtest2=$listtest2[priceid];
$flag=0;//判断电影的信息写得对不对, 初始为不对

```

电影院和价格填写正确, 进一步判断电影 id 和电影名 name:

```

$flag=0;//判断电影的信息写得对不对, 初始为不对
if($thridtest&&$thridtest2)
{
    if($mov_id)//写了电影id
    {
        if($mov_name=="")//没写电影名称
        {
            $sql="select * from $MVE where movid='$mov_id'";
            $query=mysql_query($sql,$connect)or die("sql fail1");
            $movid= mysql_fetch_array($query);
            if($movid[movid])//所填电影id在movies表中, 符合条件
            {
                $flag=1;
                $mov_name=$movid[movname];
            }
        }
        else //写了电影名称
        {
            $sql="select * from $MVE where movid=$mov_id";
            $query=mysql_query($sql,$connect)or die("sql fail2");
            $movid= mysql_fetch_array($query);
            if(!$movid[id]) //电影id不在movies表中
            {
                $sql="select * from $MVE where movname='$mov_name'";
                $query=mysql_query($sql,$connect)or die("sql fail3");
                $movname= mysql_fetch_array($query);
                if(!$movname)//电影名称和电影id都不在movies表中, 说明是新的电影, 分
                {
                    $flag=1;
                }
                else break;//若电影名称不在, 而电影id存在, 说明输入有误
            }
        }
    }
}

```

```
if($movie[movname]==$mov_name)//电影id和电影名称相符合
{
    $flag=1;
}
}
if($mov_id<1000||$mov_id>9999) $flag=0;//电影id不符合要求，须在1000到9999之间
}
else//没输入电影id
{
    $sql="select * from $MVE where movname=$mov_name";
    $movid= mysql_fetch_array($sql);
    if($movid[movid])//所填电影id在movies表中，符合条件
    {
        $flag=1;
        $mov_id=$movid[movid];
    }
}

else//没输入电影id
{
    $sql='select * from $MVE where movname='.$mov_name;
    $movid= mysql_fetch_array($sql);
    if($movid[movid])//所填电影id在movies表中，符合条件
    {
        $flag=1;
        $mov_id=$movid[movid];
    }
}
```

```

if($flag)
{
    $sqlinsert="insert into $DPL values('$mov_id','$thr_id','$price_id','$play_
//print "$sqlinsert";
$query=mysql_query($sqlinsert,$connect)or die("Can't save information");
if($query)
    {print "<p id=\"$thr_table\"><th>***SAVE DISPLAY INFO OK***</th></p>"}
else
    {print "!!!save error!!!";}
}
else//电影信息不符合要求，报错
{
    print "<p id=\"$thr_table\"><th>ERROR, check for the movie information!</th>
echo "<br><center>
    <input type=button class=button green value=\"Back to previous\"      on
echo "</center><br>";
}

```

电影院不存在或者价格不存在，阻止插入：

```

else//电影院不存在或价格不存在！！！
{
    print "<p id=\"$thr_table\"><th>ERROR, check for the theater OR price is
echo "<center><br><form>
    <input type=button value=\"Back to previous\"      onclick=\"$history.
echo "<br></center>";
}

```

调用更新操作函数过程：

```
$add=$_POST["add"];
```

```
$doadd=$_POST["doadd"];
```

通过按钮的 submit 消息，调用函数

```

else if ($add) { addplay(); }
else if ($doadd)
{
    $mov_id=$_POST["mov_id"];
    $mov_name=$_POST["mov_name"];
    $thr_id=$_POST["thr_id"];
    $price_id=$_POST["price_id"];
    $play_time=$_POST["play_time"];

    addplayfin($mov_id,$mov_name,$thr_id,$price_id,$play_time);
}

```

<p>触发器源 码 (3分)</p>	<p>(截屏) 在数据库中建立触发器:</p> <pre>导出触发器 `play_check` 1 CREATE TRIGGER `play_check` BEFORE INSERT ON `display` 2 FOR EACH ROW begin 3 if new.movid not in(select movid from movies) 4 then insert into movies(movid,movname) values(new.movid,new.movname); 5 end if; 6 end</pre>												
<p>程序演示 (4分)</p>	<p>说明: 不违背触发器能够执行插入操作。 点击 ADD_PLAY 按钮, 进入添加界面, 输入合法的信息:</p>  <p>*****Click for desired operation*****</p> <p>ADD_MOVIE DEL_MOVIE MOD_PRICE LIST_PLAY ADD_PLAY</p> <table border="1"> <tr> <td>Movie id</td> <td>1009</td> </tr> <tr> <td>Movie name</td> <td>Finishing SQL</td> </tr> <tr> <td>Theater id</td> <td>1</td> </tr> <tr> <td>Price id</td> <td>1</td> </tr> <tr> <td>playtime</td> <td>2014-12-29 15:00</td> </tr> <tr> <td colspan="2">Save_MoviePlay_Info</td> </tr> </table> <p>提交表单, 提示成功:</p>  <p>*****Click for desired operation*****</p> <p>ADD_MOVIE DEL_MOVIE MOD_PRICE LIST_PLAY ADD_PLAY</p> <p>***SAVE DISPLAY INFO OK***</p>	Movie id	1009	Movie name	Finishing SQL	Theater id	1	Price id	1	playtime	2014-12-29 15:00	Save_MoviePlay_Info	
Movie id	1009												
Movie name	Finishing SQL												
Theater id	1												
Price id	1												
playtime	2014-12-29 15:00												
Save_MoviePlay_Info													

说明：违背触发器要求，不能够执行插入操作，系统报错。

输入不合法的 movieid，提交表单：

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

Movie id	10009
Movie name	Finishing SQL
Theater id	3
Price id	1
playtime	2014-12-29 15:00
Save_MoviePlay_Info	

程序
演示
(4
分)

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

ERROR, check for the movie information!

[Back to previous](#)

由于 phpMyAdmin 的默认转义符为 ‘;’，而编写的触发器的 sql 语句中有 ‘;’，因此无法直接从 php 里将 sql 语句传到数据库中进行生成触发器操作,会发生错误:

备注

[语句定界符] 在此再次显示此查询 保留查询框

错误

SQL 查询: [?](#)

```
CREATE TRIGGER `play_check` BEFORE INSERT ON `display`
FOR EACH
ROW
BEGIN
IF new.movid NOT
IN (
    SELECT movid
    FROM movies
)
THEN INSERT INTO movies( movid, movname )
VALUES (
    new.movid, new.movname
);
```

MySQL 返回: [?](#)

```
#1064 - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'SELECT movid
FROM movies
)' at line 5
```

若要用 sql 语句执行添加触发器操作，需要在执行前手动修改转义符：

[语句定界符] 在此再次显示此查询 保留查询框

修改后再执行添加触发器操作，执行成功：

 您的 SQL 语句已成功运行 (查询花费 0.0090 秒)

```
CREATE TRIGGER `play_check` BEFORE INSERT ON `display`
FOR EACH
ROW
BEGIN
IF new.movid NOT
IN (
    SELECT movid
);
```

学到一招~

3. 存储过程控制下的更新操作 (20 分)

说明 (1分)	(1分) 简要说明该操作所要完成的功能; (1分) 简要说明该存储过程所要完成的功能; (2分) 说明该操作涉及操作的表(必须包含两张或两张以上的关系表,以“表名形式”描述) (1分) 表连接涉及字段描述(描述方式为“表1.属性=表2.属性”) (2分) 该操作会修改字段(以“表名.字段名”的形式给出),以及修改规则,如新数值的计算方法、在何种条件下予以修改等; (6分) 实现该操作的关键代码(高级语言、SQL),截图即可; (7分) 如何执行该操作,按所述方法能够正常演示程序则给分。						
功能描述 (1分)	当修改电影类型的价格 price 时,会影响订单的总价格 totalprice,因此在修改价格时,使用存储过程更新订单表。						
存储过程功能描述 (1分)	该存储过程涉及两个参数,新的价格 price 和新价格的 priceid。 在修改价格之后调用存储过程进行更新订单表 orders 的信息,修改对应 priceid 价格类型的 totalprice=new.price*amount。						
涉及的关系表 (2分)	价格表 price 订单表 orders 放映信息表 display						
表连接涉及字段 (1)	price.priceid=display.priceid orders.playid=display.playid						
更改字段 (2分)	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">字段</th> <th style="text-align: center; padding: 5px;">规则</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">price.thrprice</td> <td style="padding: 5px;">如果 price 在[0, 200]之间,认为是符合条件,可以进行修改; 否则不能进行修改。</td> </tr> <tr> <td style="padding: 5px;">order.totalprice</td> <td style="padding: 5px;">如果修改的 price 的 priceid 与订单表当中的订单有联系,则修改该订单的总价格为 new.price*amount,否则不改变。</td> </tr> </tbody> </table>	字段	规则	price.thrprice	如果 price 在[0, 200]之间,认为是符合条件,可以进行修改; 否则不能进行修改。	order.totalprice	如果修改的 price 的 priceid 与订单表当中的订单有联系,则修改该订单的总价格为 new.price*amount,否则不改变。
字段	规则						
price.thrprice	如果 price 在[0, 200]之间,认为是符合条件,可以进行修改; 否则不能进行修改。						
order.totalprice	如果修改的 price 的 priceid 与订单表当中的订单有联系,则修改该订单的总价格为 new.price*amount,否则不改变。						
更新代码 (3分)	(截屏)						

```
//存储过程的更新操作
//修改电影票的价格，然后存储过程将会把预定表orders中的与该电影票相关的订单的价格全部更新
function modform()
{
    require "setup.php";
    $sql="select* from $PRI;";//mysql_query("set names gb2312");
    $connect=mysql_connect($DB_SERVER,$DB_USER,$DB_PASS)or die("连接失败");
    $db=mysql_select_db($DB_NAME,$connect)or die("选择失败");
    $query=mysql_query($sql,$connect)or die("执行失败");
    if($query)
    {
        print "<br><center><select id=\"thr_but\" name=selname>";
        while($list=mysql_fetch_array($query))
        {
            print "<option value=$list[priceid]>$list[pricetype]";//t1
        }
        print "</select><input type=submit name=showmod value=Submit_modif
    }
}

function showmod($selname)//执行修改内容
{
    require "setup.php";
    $sql="select* from $PRI where priceid='$selname'";//mysql_query("set name
    $connect=mysql_connect($DB_SERVER,$DB_USER,$DB_PASS)or die("连接失败");
    $db=mysql_select_db($DB_NAME,$connect)or die("选择失败");
    //print $sql;
    $query=mysql_query($sql,$connect)or die("执行失败");
    if($query)
    {
        while($list=mysql_fetch_array($query)) {
            $id=$list[priceid];
            $price=$list[thrprice];
            $type=$list[pricetype];
            print "<tr><tr><table id=\"thr_table\">
                <br><br><tr ><center><th colspan=2>Modify the price</center></tr>
                <tr><th >priceid
                <td><input type=hidden name=price_id value=$id></td>
                <tr><th >price
                <td><input type=text name=price_ value=$price></td>
                <tr><th >type
                <td><input type=text name=price_type value=$type></td>
                <tr><td colspan=2><center><input type=submit name=domod value=Comm
            ";
        }
    }
}
```

```

function domod($priceid,$price,$type)//完成修改价格和存储过程更新操作
{
    require "setup.php";
    $connect=mysql_connect ($DB_SERVER,$DB_USER,$DB_PASS) or die("连接失败");
    $db=mysql_select_db ($DB_NAME,$connect) or die("选择失败");
}

更新操作具体:
//存储过程
if($price<0||$price>200)//价格设置不合理
{
    print "<br><p id=\"thr_table\"><center><th><br>!!!Check for your price!!!<br></th></tr><tr><td><br><center><br><form>
        <input type=button class=button green value=\"Back to previous\" onclick="history.back();"></input>
    </td></tr></table>";
}
else
{
    $sql = "call mod_price($priceid,$price)";
    $query=mysql_query($sql,$connect) or die("EXE FAILxxxx");
    if(!$query){
        PRINT "procedure fail!";
    }
    else print "<br><p id=\"thr_table\"><<th>rocedure calling OK!</th></p>";
    $sql="update $PRI set thrprice='$price',pricetype='$type' where priceid='$priceid'";
    $connect=mysql_connect ($DB_SERVER,$DB_USER,$DB_PASS) or die("连接失败");
    $db=mysql_select_db ($DB_NAME,$connect) or die("SEL FAIL2");
    $query=mysql_query($sql,$connect) or die("EXE FAIL2");
    if(!$query){print "mod error";}
    else{
        print "<p id=\"thr_table\"><th><br>***MODIFY PRICE OK!***<br></th></p>";
    }
}
}

```

创建
存储
过程
源码
(3
分)

(截屏)

在数据库中建立存储过程:

```

CREATE DEFINER='gyx'@'localhost'
PROCEDURE `mod_price` (IN id_in INT(11), IN pri_in FLOAT)
BEGIN
    UPDATE orders SET totalprice=pri_in*amount
    WHERE playid in
    (
        SELECT playid FROM display dpl
        WHERE dpl.priceid=id_in
    );
END

```

存储
过程
执行
源码

(截屏)

提交更新消息，给变量赋值作为调用函数的依据:

(1分)	<pre>\$mod=\$_POST["mod"]; \$showmod=\$_POST["showmod"]; \$domod=\$_POST["domod"]; 调用相应函数: else if(\$mod) {modform();} else if(\$showmod) {\$selname=\$_POST["selname"]; showmod(\$selname);} else if(\$domod) {//\$id=\$_POST["id"]; \$price=\$_POST["price_"]; \$priceid=\$_POST["price_id"]; \$type=\$_POST["price_type"]; domod(\$priceid,\$price,\$type); }</pre>
程序演示(3分)	<p>说明：不违背存储过程，能够执行更新操作 点击更新按钮出现更新界面，点击按钮 Submit_modify:</p>  <p>The screenshot shows a yellow-themed user interface for a theater manager. At the top, it says "THEATER MANAGER". Below that is a text area with the instruction "Click for desired operation". Underneath are five buttons labeled "ADD_MOVIE", "DEL_MOVIE", "MOD_PRICE", "LIST_PLAY", and "ADD_PLAY". At the bottom left is a dropdown menu set to "2D", and at the bottom right is a button labeled "Submit_modify".</p> <p>将价格修改为 80，点击 Commit_modify:</p>

THEATER MANAGER

*****Click for desired operation*****

ADD_MOVIE

DEL_MOVIE

MOD_PRICE

LIST_PLAY

ADD_PLAY

Modify the price

priceid	
price	85
type	2D
Commit_modify	

提示存储成功

THEATER MANAGER

*****Click for desired operation*****

ADD_MOVIE

DEL_MOVIE

MOD_PRICE

LIST_PLAY

ADD_PLAY

PROCEDURE OK

procedure calling OK!

MODIFY PRICE OK!

程序
演示
(3
分)

说明：违背存储过程，系统报错；
若修改的价格不在区间[0, 200]内，报错：

THEATER MANAGER

*****Click for desired operation*****

ADD_MOVIE

DEL_MOVIE

MOD_PRICE

LIST_PLAY

ADD_PLAY

PROCEDURE OK

!!!Check for your price!!!

[Back to previous](#)

备注

4. 含有视图的查询操作 (15分)

说明 (1分)	(1分) 简要说明该操作所要完成的功能; (1分) 简要说明建立的该视图的功能; (2分) 简要说明该操作涉及的关系数据表 (以“表名”的形式给出) (1分) 简要说明表连接涉及的字段 (以“表 1.属性=表 2.属性”) (6分) 实现该操作的关键代码 (高级语言、SQL), 截图即可; (4分) 如何执行该操作, 按所述方法能够正常演示程序则给分。
操作功能描述 (1分)	选择电影名, 查询放映信息
视图功能描述 (1分)	显示查询结果, 显示内容包括电影名称, 电影院, 电影价格, 电影效果, 放映时间和电影介绍。
涉及的关系表 (2分)	电影表 movies 电影院表 theater 价格表 price 放映信息表 display
表连接字段 (1分)	movies.movid=display.movid, theater.thrid=display.thrid, price.priceid=display.priceid,
创建视图代码 (3分)	(截屏) 在数据库中建立视图: <hr/> <pre> CREATE VIEW playinfo AS select mov.movname, mov.movid, thrname, thrprice, pricetype, playtime, intro from movies mov, display dpl, price pri, theater thr where mov.movid=dpl.movid and thr.thrid=dpl.thrid and pri.priceid=dpl.priceid </pre>
查询代码 (3分)	(截屏) 点击 L I S T _ P L A Y 按钮, 触发函数 listform()

```
//创建视图
//通过选择电影名称，创建视图，筛选出电影放映信息并显示出来
function listform()
{
    require "setup.php";
    $sql="select* from $MVE";
    $connect=mysql_connect ($DB_SERVER,$DB_USER,$DB_PASS)or die("Link fail");
    $db=mysql_select_db($DB_NAME,$connect)or die("sel fail");
    $query=mysql_query($sql,$connect)or die("exe fail");
    if($query)
    {
        print "<br><table id=\"thr_table\"><tr>
Movies_Available</tr>";
        print "<br><br><select name=selmovie>";
        while($list=mysql_fetch_array($query))
        {
            print "<option value=$list[movid]>$list[movname]</option>";
        }
        print "</select><input type=submit name=showmovie value=Choose_a_m
print "</table>";
    }
}
```

建立视图，之后显示出来

```
function playview($selmovie)
{
    require "setup.php";

    $connect=mysql_connect ($DB_SERVER,$DB_USER,$DB_PASS)or die("Link fail");
    $db=mysql_select_db($DB_NAME,$connect)or die("sel fail");
```

```

//显示视图
$sqlv="select * from playinfo where movid=$selmovie";
$queryv=mysql_query($sqlv,$connect)or die("exe failv");
print "<br><center> ***The movie display information are listed***</center>";
print "<br><table id=\"thr_table\">";
print "<tr class=\"alt\">";
    <th>Moviename</th><th>Theater</th><th>Price</th><th>Type</th><th>Time</th>
while($list=mysql_fetch_array($queryv))
{
    print "<tr ><td>$list[movname]</td>
            <td>$list[thrname]</td>
            <td>$list[thrprice]</td>
            <td>$list[pricetype]</td>
            <td>$list[playtime]</td>
            <td>$list[intro]</td></tr> ";
}
print"</table>";
echo "<center><br><form>
        <input type=button class=button green value=\"Back to previous\" or
echo "<br></center>";

}

声明变量， submit 提交表单时触发消息获得表单的值：

$list=$_POST["list"];
$showmovie=$_POST["showmovie"];

调用函数，开始视图功能：
else if($list){listform();}
else if($showmovie){$selmovie=$_POST["selmovie"];playview($selmovie);}

```

点击 ADD_PLAY 按钮，出现选择电影的界面：

THEATER MANAGER

程序演示
(4分)

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

Movies_Available

[Finding Mr.Right ▾](#) [Choose_a_movie_you_want](#)

点击 Choose_a_movie_you_want 按钮，列出 Finding Mr.Right 电影的全部信息

THEATER MANAGER

*****Click for desired operation*****

[ADD_MOVIE](#) [DEL_MOVIE](#) [MOD_PRICE](#) [LIST_PLAY](#) [ADD_PLAY](#)

The movie display information are listed

Moviename	Theater	Price	Type	Time	Introduction
Finding Mr.Right	Sunny Theater	80	2D	2013-12-30 18:00:00	Very Moving and surprising!
Finding Mr.Right	Sunny Theater	120	IMAX	2013-12-30 21:00:00	Very Moving and surprising!
Finding Mr.Right	Sunny Theater	150	3D	2013-12-30 10:00:00	Very Moving and surprising!
Finding Mr.Right	Happy Theater	80	2D	2013-12-30 18:00:00	Very Moving and surprising!

[Back to previous](#)

备注