**任务一 php 操作mdb 数据库：写一个文件 paixu.php 后面可带参数 如 paixu.php?id=xxx&mubiao=zhibo**

如果mubiao=zhibo将 字段 msignal的内容按照 配置文件排序配置（1 、2或3）的要求重新排序

如果mubiao=video将 字段 signal的内容按照 配置文件排序配置（3 、4或5）的要求重新排序

1. **特别说明：** mdb数据库采用的编码为 gb2312 ，在写入数据库时大多数数据的实际编码为 gbk, 不影响实际使用

**2 数据结构** 每条数据之间用**{y}**连接 ，每条数据内部又分为三部分 名称 、 站点、 地址 ，它们之间用 **|** 连接

如： 名称**|**站点**|**地址**{y}**第1节**|**migu**|**222**{y}**全场录像**|**pptv**|**333**{y}**全场录像**|**weibo**|**444**{y}**全场录像**|**migu**|**555

**3 配置文件包含内容：**

**A** 数据库配置： $config\_db = 'qqq.mdb';

**B** 排序配置：

配置1： 'zhibo\_name' => array('卫星1','卫星2','官方高清','中文高清','原声高清','原声标清'),

配置2： 'zhibo\_site \_1' => array(' m3u8',' jrs2',' jrs',' yuyan ',' renren ',' wanqiu ',' shanmao ',' qiutt ',' xinying',' pptv',' qq',' migu'),

配置3： 'zhibo\_ site \_2' => array(' jrs2',' m3u8',' jrs',' renren ',' yuyan ',' wanqiu ',' shanmao ',' migu',' qiutt ',' xinying',' pptv',' qq'),

配置4： 'video\_name' => array('精华集锦',’第2节’,'[中文]第\*节','第\*节' ,'上半场','下半场','全场录像’,'第\*局','全场集锦'),

配置5： 'video\_ site \_1 ' => array('cntv2','pptv','migu','xinying','QQ','mycolude','aiqiyi'),

配置6： 'video\_ site \_2' => array('weibo','pptv','migu','cntv2','xinying','QQ','mycolude','aiqiyi'),

**4 排序原理 难点，光从文字比较难说清和理解，这里最好语音沟通**

可以理解为 各站点（如配置2、3、5、6）为排好序的大箱子，每个大箱子里面装了排好序的小盒子，这些小盒子就是信号名称（如配置1、4）。将每个信号按照 信号的二部分站点（相等）和第一部分信号名称（包含），分别放进对应的小盒子里面， 同一个小盒子里的信号再根据通配符的数字顺序排列。 最后得到每个信号的位置，根据这些位置重新排列组合。 \* 对于同一个盒子里的信号如果无法通过上述原则决定位置，则按原先的先后确定）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **weibo** | **pptv** | **migu** | **cntv2** | **xinying** | **QQ** | **mycolude** | **aiqiyi** | **其他** |
| 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 | 精华集锦 |
| 第2节 | 第2节 | 第2节 | 第22节 | 第2节 | 第2节 | 第2节 | 第22节 | 第2节 |
| [中文]第**\***节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 | [中文]第\*节 |
| 第\*节 | 第\*节 | 第\*节 | 第\*节 | 第\*节 | 第\*节 | 第\*节 | 第\*节 | 第\*节 |
| 上半场 | 上半场 | 上半场 | 上半场 | 上半场 | 上半场 | 上半场 | 上半场 | 上半场 |
| 下半场 | 下半场 | 下半场 | 下半场 | 下半场 | 下半场 | 下半场 | 下半场 | 下半场 |
| 全场录像 | 全场录像 | 全场录像 | 全场录像 | 全场录像 | 全场录像 | 全场录像 | 全场录像 | 全场录像 |
| 第\*局 | 第\*局 | 第\*局 | 第\*局 | 第\*局 | 第\*局 | 第\*局 | 第\*局 | 第\*局 |
| 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 | 全场集锦 |
| 其他 | 其他 | 其他 | 其他 | 其他 | 其他 | 其他 | 其他 | 其他 |

**5排序过程**

1 先得到第2部分的站点：如果第2部分包含 “site=”， 站点为为 sitie= 的内容，如果不包含则整个第二部分为站点。

比如 第2节**|**playbox.html?site=jrs2&matchid=**|**111 站点为 jrs2 比如 第2节**|**migu**|**111 站点为 migu

2 根据 站点排序的配置 ，将这些信号进行第一次排序。 （注意，站点判断为相等，不区分大小写）

3 同一个站点内的信号之间，再根据第一部分的信号名称在配置中的位置进行排列 （注意，信号名称判断为包含）

4同一个名称之间 再按照**通配符\***的数字顺序排， 数字有 中文和阿拉伯，最大数字不超过10 **这个最难，参考练习**

**6 练习** 名称中的通配符练习： 请注意配置中红色字体为变化的内容，

例题1： 第2节录像**|**migu**|**111**{y}**第1节录像**|**migu**|**222**{y}**全场录像**|**pptv**|**333**{y}**全场录像**|**weibo**|**444**{y}**全场录像**|**migu**|**555

A 如果采用配置 'video\_ site \_1 ' （该配置没有 weibo，因此归为其他）

处理结果： 全场录像|pptv|3333 {y} 全场录像||migu|555 {y} 第1节录像|migu|222 {y} 第2节录像|migu|111 {y} 全场录像|weibo|444

B如果采用 'video\_play\_1' ，（该配置没有 migu，因此归为最后一档 ）

处理结果 全场录像||weibo|444{y}全场录像|pptv|333{y}全场录像|migu|555{{y}第2节录像|migu|111y}第1节录像|migu|222

C 此时如果把 录像名称配置 的 第2节 删掉，则排序结果变为

全场录像||weibo|444{y}全场录像|pptv|333{y}全场录像|migu|555{y}第1节|migu|222{y}第2节|migu|111

例题2： 名称中的通配符理解：

全场录像 全场集锦 第3节录像 第2节录像 第4节录像 [中文]第5节录像

**如果配置为：** 'video\_name' => array('精华集锦', '第\*节录像' ,'上半场录像','下半场录像','全场录像’,'第\*局录像','全场集锦'),

排序结果为： 第2节录像 第3节录像 第4节录像 [中文]第5节录像 全场录像 全场集锦

**如果配置为：** 'video\_name' => array('精华集锦', '[中文]第\*节录像','第\*节录像' ,'上半场录像','下半场录像','全场录像’,'第\*局录像','全场集锦'),

排序结果为： [中文]第5节录像 第2节录像 第3节录像 第4节录像 全场录像 全场集锦

**如果配置为：** 'video\_name' => array('精华集锦',’第4节’,'[中文]第\*节录像','第\*节录像' ,'上半场录像','下半场录像','全场录像’,'第\*局录像','全场集锦'), 排序结果为：第4节录像 [中文]第5节录像 第2节录像 第3节录像 全场录像 全场集锦

**7**  **操作方式**

**写一个php文件，比如 paixu.php 当该文件后面带id参数时，进行单个数据处理，如果id为空，进行批量数据处理，具体说明如下：**

**操作方式1：** 单个数据处理： Paixu.php?id=xxx&mubiao=zhibo 其中：id为数据的id号,表示要对数据表中该id数据进行处理。

当 mubiao= zhibo 时 操作的字段为Msignal 站点排序配置为：配置3 **名称排序配置为** 配置1

当 mubiao= video 时 操作的字段为Signal 站点排序配置为：配置6**' 名称排序配置为** 配置4

处理过程：1读取参数 id 和 mubiao 根据目标确定对应的字段和 排序配置

2 读取该id的 字段的内容，按照配置要求重新排序，

3 将重新排序后的内容写入 该字段（替代原先的内容）。 其他数据无须做任何改动

**操作方式2**： 批量数据处理： Paixu.php?mb=1 不带id参数时的操作界面

**输出结果** 是 否 // 默认选中是

**排序目标**  **录像** 直播 // 默认选中根据 mb的值，当mb=1是默认选中 录像 否则默认选中 直播

**时间范围**  起始时间  **+** 2020-02-08 06:24 **-** 结束时间 **+** 2020-02-08 12:24 **-**

**开始处理**

说明：1排序目标，可以切换

如果选择直播，则操作字段为Msignal 站点排序配置为：配置2 **名称排序配置为** 配置1

如果选择录像 则操作字段为Signal 站点排序配置为：配置5 **名称排序配置为** 配置4

2 默认起始时间和结束时间 .只保留到分钟，秒去掉不显示，输入框中的数字修改

选择录像 起始时间默认值为 当前时间的前六小时 结束时间为当前时间

选择直播 起始时间默认值为 当前时间，**结束时间为** 当前时间的后六小时

按 加减少 每按一次增加或者减少一小时，也可以之间在输入框内修改

3 输出结果默认选择 是

点击 开始处理 调出数据库满足时间条件的数据开始处理，处理完毕 根据选择的输出结果 做相应提示

如果提示为否，则只需提示 处理完毕，一共处理 xx 条数据

如果提示为是，还需要把处理的每一条处理的对应数据输出来//主要在测试阶段用来观察对比

第一行 比赛时间：stime 比赛信息： title game1 - game2

第二行 处理前信号 signal (或msignal)

第三行 处理后信号 signal (或msignal)

处理过程：1根据页面选择确定各参数 起始时间 结束时间 排序目标

2 从数据库中调取满足 起始时间 < stime < 结束时间 singal(或msignal ，由选中的排序目标确定) 不为空的数据

3 处理每一条数据，得到重新排列后的信号

4 将重新排序后的内容写入 该字段（替代原先的内容）。 其他数据无须做任何改动

5 输出结果

**任务二 操作数据库** act\_mdb.php

红色部分为选择或者填写的内容

1数据表 yet\_play ▼

2查询字段 signal ▼ 1该字段条件

3字段条件 口包含 .php 口等于 口数据为空 口无条件

4范围 口id范围 至 口stime范围 2007-2-5 12:00:00 至 2017-2-5 15:00:00

5操作方式 口删除该字符 口替换新字符 口清空该字段全部内容 口删除该条数据 口重设其他字段msgnal ▼ 重设内容

开始操作

操作过程：根据1选择的数据表和2选择的查询字段，找出符合3和4 条件的数据。然后按照5的要求处理数据库对应的内容

说明：1数据表自动列出该数据库的全部数据表供选择。提供的数据库只有一个数据表，如有需要测试再另外提供

2查询字段：当选择了其中一个表，则自动列出该表的全部字段供选择

3字段条件 为四选一勾选，其中选择了 包含或者等于，还需要填写对应的内容。

4 范围： 为二选一勾选并填写 起始id和终点id, 或stime起点和终点，其中 stime范围只针对 YET\_TYPE 这个表有效（对应字段为 stime）

5操作方式为五选一 ， 其中第二项和第五项如果选择了还需要填写内容

选择删除该字符 表示把条件4 包含或者等于 填写的字符删除。

选择替换新字符 表示用这个新字符替换 条件4 包含或者等于 填写的字符

选择清空该字段全部内容 表示把这个字段的内容设为空

选择删除该条数据 表示把这个id删除

选择重设其他字段 则表示把选中的字段的内容统一设为填写的内容（内容可以为空）

操作之前 先备份数据库，并先检查必填和必选的内容，如果缺少则提示，操作完毕提示一共处理多少数据，