实验四：分布式数据库数据函数和索引

学号：202103151422　　姓名：温家伟　　班级：大数据分析 2101

1、 实验环境安装情况（用☑替换☐即可）

a) ☑自安装环境　☐高校邦系统（如果选择后者，以下三个不用回答）

b) ☐单机　☑多机

c) ☐单数据库实例　☑两个数据库实例　☐多个数据库实例

d) ☑同类型数据库实例　☐两个类型数据库实例　☐多类型数据库实例

e) ☐Mycat　☑ShardingSphere

f) ☑MySQL　☐OpenGauss　☐其他_____

2、 实验内容

a) 针对实验运行环境（用☑替换☐即可）

☑命令行下进行实验　☐Java 应用程序进行实验　☐Python 应用程序进行实验

☐其他_____

b) 针对前三个实验的学生表 Stu、课程表 Cou 和学生选课表 SC，继续插入一些数据，尽量贴合实际，以备后续查询使用，插入数据之后完成如下查询：

利用 Count（）和 Avg（）函数以及 Group by 关键字，查询整个库中各课程的课程名、选课人数和平均分。

能否实现：☑能　☐不能（如果可以完成下表）

整个库（Mycat 或 ShardingSphere）
SQL 语句：select Cname as 课程名,count(wenjw_SC.Sno) as 选课人数, avg(Grade) as 均分 from wenjw_Cou, wenjw_SC where wenjw_Cou.Cno = wenjw_SC.Cno group by Cname; 结果（截图）： mysql> select Cname as 课程名,count(wenjw_SC.Sno) as 选课人数, avg(Grade) as 均分 from wenjw_Cou, wenjw_SC where wenjw_Cou.Cno = wenjw_SC.Cno group by Cname; +----------+----------+---------+ | 课程名 | 选课人数 | 均分 | +----------+----------+---------+ | 人工智能 | 3 | 76.6667 | | 分布式数据库 | 2 | 82.5000 | | 大学物理 | 3 | 64.0000 | | 数字图像处理 | 5 | 66.6000 | | 数学建模 | 4 | 89.0000 | | 数据结构 | 4 | 72.7500 | | 程序设计基础C | 5 | 79.6000 | | 运筹学 | 4 | 70.2500 | +----------+----------+---------+ 8 rows in set (0.03 sec)

利用 Count（）和 Avg（）函数以及 Group by 关键字，查询每个库中各课程的课程名、选课人数和平均分。

数据库 1	数据库 2
SQL 语句： select Cname as 课程	SQL 语句： select Cname as 课程

<table>
<tr>
<td>名,count(wenjw_SC0.Sno) as 选课人数, avg(Grade) as 均分 from wenjw_Cou, wenjw_SC0 where wenjw_Cou.Cno = wenjw_SC0.Cno group by Cname;
结果（截图）：

```
mysql> select Cname as 课程名,count(wenjw_SC0.Sno) as
选课人数, avg(Grade) as 均分 from wenjw_Cou, wenjw_SC0
 where wenjw_Cou.Cno = wenjw_SC0.Cno group by Cname;
+--------------------+--------------+----------+
| 课程名             | 选课人数     | 均分     |
+--------------------+--------------+----------+
| 分布式数据库       |            1 | 65.00000 |
| 大学物理           |            2 | 63.00000 |
| 数字图像处理       |            2 | 71.50000 |
| 数学建模           |            2 | 87.50000 |
| 数据结构           |            3 | 74.33333 |
| 程序设计基础C      |            2 | 92.50000 |
| 运筹学             |            3 | 75.33333 |
+--------------------+--------------+----------+
7 rows in set (0.00 sec)
```

</td>
<td>名,count(wenjw_SC1.Sno) as 选课人数, avg(Grade) as 均分 from wenjw_Cou, wenjw_SC1 where wenjw_Cou.Cno = wenjw_SC1.Cno group by Cname;
结果（截图）：

```
MySQL [wenjw_db1]> select Cname as 课程名,count(wenjw_SC1
.Sno) as 选课人数, avg(Grade) as 均分 from wenjw_Cou, wen
jw_SC1 where wenjw_Cou.Cno = wenjw_SC1.Cno group by Cname
;
+--------------------+--------------+----------+
| 课程名             | 选课人数     | 均分     |
+--------------------+--------------+----------+
| 人工智能           |            3 |  76.66667 |
| 分布式数据库       |            1 | 100.00000 |
| 大学物理           |            1 |  66.00000 |
| 数字图像处理       |            3 |  63.33333 |
| 数学建模           |            2 |  90.50000 |
| 数据结构           |            1 |  68.00000 |
| 程序设计基础C      |            3 |  71.00000 |
| 运筹学             |            1 |  55.00000 |
+--------------------+--------------+----------+
8 rows in set (0.00 sec)
```

</td>
</tr>
</table>

c) 针对前三个实验的学生表 Stu、课程表 Cou 和学生选课表 SC，保证数据与上一致，完成如下查询：

利用 Count（）和 Avg（）函数以及 Group by 和 Having 关键字，查询整个库中各课程的课程名、及格学生人数和及格学生平均分。

能否实现：☑能　　☐不能（如果可以完成下表）

<table>
<tr><td>整个库（Mycat 或 ShardingSphere）</td></tr>
<tr><td>SQL 语句：select Cname as 课程名,count(wenjw_SC.Sno) as 及格人数, avg(Grade) as 及格均分 from wenjw_Cou, wenjw_SC where wenjw_Cou.Cno = wenjw_SC.Cno and Grade >= 60 group by Cname;
结果（截图）：

```
mysql> select Cname as 课程名,count(wenjw_SC.Sno) as
及格人数, avg(Grade) as 及格均
分 from wenjw_Cou, wenjw_SC where wenjw_Cou.Cno = wen
jw_SC.Cno and Grade >= 60 group by Cname;
+--------------------+--------------+--------------+
| 课程名             | 及格人数     | 及格均分     |
+--------------------+--------------+--------------+
| 人工智能           |            3 |      76.6667 |
| 分布式数据库       |            2 |      82.5000 |
| 大学物理           |            3 |      64.0000 |
| 数字图像处理       |            3 |      87.6667 |
| 数学建模           |            4 |      89.0000 |
| 数据结构           |            3 |      77.6667 |
| 程序设计基础C      |            3 |      95.0000 |
| 运筹学             |            2 |      83.5000 |
+--------------------+--------------+--------------+
8 rows in set (0.03 sec)
```

</td></tr>
</table>

利用 Count（）和 Avg（）函数以及 Group by 和 Having 关键字，查询每个库中各课程的课程名、及格学生人数和及格学生平均分。

<table>
<tr><td>数据库 1</td><td>数据库 2</td></tr>
<tr><td>SQL 语句：
select Cname as 课程名,count(wenjw_SC1.Sno) as 及格人数, avg(Grade) as 及格均分 from wenjw_Cou, wenjw_SC1 where</td><td>SQL 语句：
select Cname as 课程名,count(wenjw_SC0.Sno) as 及格人数, avg(Grade) as 及格均分 from wenjw_Cou, wenjw_SC0 where</td></tr>
</table>

wenjw_Cou.Cno = wenjw_SC1.Cno and Grade >= 60 group by Cname;

结果（截图）：

```
MySQL [wenjw_db1]> select Cname as 课程名,count(wenjw_SC1
.Sno) as 及格人数, avg(Grade) as 及格均分 from wenjw_Cou,
 wenjw_SC1 where wenjw_Cou.Cno = wenjw_SC1.Cno and Grade
>= 60 group by Cname;
+--------------------+-------------+-------------+
| 课程名             | 及格人数    | 及格均分    |
+--------------------+-------------+-------------+
| 人工智能           |           3 |    76.66667 |
| 分布式数据库       |           1 |   100.00000 |
| 大学物理           |           1 |    66.00000 |
| 数字图像处理       |           2 |    87.50000 |
| 数学建模           |           2 |    90.50000 |
| 数据结构           |           1 |    68.00000 |
| 程序设计基础C      |           1 |   100.00000 |
+--------------------+-------------+-------------+
7 rows in set (0.00 sec)
```

wenjw_Cou.Cno = wenjw_SC0.Cno and Grade >=60 group by Cname;

结果（截图）：

```
mysql> select Cname as 课程名,count(wenjw_SC0.Sno) as
及格人数, avg(Grade) as 及格均分 from wenjw_Cou, wenjw
_SC0 where wenjw_Cou.Cno = wenjw_SC0.Cno and Grade >=
60 group by Cname;
+--------------------+-------------+-------------+
| 课程名             | 及格人数    | 及格均分    |
+--------------------+-------------+-------------+
| 分布式数据库       |           1 |    65.00000 |
| 大学物理           |           2 |    63.00000 |
| 数字图像处理       |           1 |    88.00000 |
| 数学建模           |           2 |    87.50000 |
| 数据结构           |           2 |    82.50000 |
| 程序设计基础C      |           2 |    92.50000 |
| 运筹学             |           2 |    83.50000 |
+--------------------+-------------+-------------+
7 rows in set (0.00 sec)
```

d) 针对实验一的学生表 Stu 基于学生姓名 Sname 属性创建名为 Sname_index 的索引：利用 Create index 命令在整个库的 Stu 表的 Sname 属性上创建索引。

能否实现：☑能　　☐不能（如果可以完成下表）

SQL 语句：create index idx_Sname on wenjw_Stu(Sname);

创建索引结果截图：

```
mysql> create index idx_Sname on wenjw_Stu(Sname);
Query OK, 0 rows affected (0.04 sec)

mysql> show index from wenjw_Stu;
+-----------+------------+-----------+--------------+----------
-+---------------+
| Table     | Non_unique | Key_name  | Seq_in_index | Column_nar
 | Index_comment |
+-----------+------------+-----------+--------------+----------
-+---------------+
| wenjw_Stu |          0 | PRIMARY   |            1 | Sno
 |               |
| wenjw_Stu |          1 | idx_Sname |            1 | Sname
 |               |
+-----------+------------+-----------+--------------+----------
-+---------------+
2 rows in set (0.00 sec)
```

利用 Create index 命令在每个库的 Stu 表的 Sname 属性上创建索引

数据库 1	数据库 2
SQL 语句： 创建索引结果截图：	SQL 语句： 创建索引结果截图：

查日志可以看到：再逻辑表上执行不含分片键的 sql 语句时，会进行全路由，即在每个数据库实例中执行一遍对应的 sql 语句。

```
Logic SQL: create index idx_Sname on wenjw_Stu(Sname)
SQLStatement: MySQLCreateIndexStatement(super=CreateIndexStatement(super=AbstractSQLS
Actual SQL: wenjw_db1 ::: create index idx_Sname_wenjw_Stu1 on wenjw_Stu1(Sname)
Actual SQL: wenjw_db0 ::: create index idx_Sname_wenjw_Stu0 on wenjw_Stu0(Sname)
```

e) 拓展实验（可选做），练习和掌握 Explain 命令——可以很好的查看查询执行计划，包括查询用没用到索引等，可以试着查看如下查询 SQL 语句的查询执行计划：利用 Select 关键字查询整个库中姓名为“张三”的学生的所有信息。

能否实现：☑能　　☐不能（如果可以完成下表）

整个库（Mycat 或 ShardingSphere）
SQL 语句：

select * from wenjw_Stu where Sname = '张三';

explain select * from wenjw_Stu where Sname = '张三'\G

Explain 命令结果截图：

```
mysql> select * from wenjw_Stu where Sname = '张三';
+------+--------+--------+-------------+------+------+---------
| Sno  | Sname  | Sdept  | Stel        | Sex  | Sage | Saddr
+------+--------+--------+-------------+------+------+---------
| 2112 | 张三   | 计院   | 18914160052 | 男   |   19 | 上海市
+------+--------+--------+-------------+------+------+---------
1 row in set (0.03 sec)

mysql> explain select * from wenjw_Stu where Sname = '张三'\G
*************************** 1. row ***************************
           id: 1
  select_type: SIMPLE
        table: wenjw_Stu0
   partitions: NULL
         type: ref
possible_keys: idx_Sname_wenjw_Stu0
          key: idx_Sname_wenjw_Stu0
      key_len: 303
          ref: const
         rows: 1
     filtered: 100.0
        Extra: NULL
1 row in set (0.00 sec)
```

利用 Select 关键字查询每个库中姓名为“张三”的学生的所有信息。

能否实现：☑能　　☐不能（如果可以完成下表）

数据库 1	数据库 2
SQL 语句： select * from wenjw_Stu0 where Sname = '张三'; explain select * from wenjw_Stu0 where Sname = '张三'\G Explain 命令结果截图： [illegible]	SQL 语句： select * from wenjw_Stu1 where Sname = '张三'; explain select * from wenjw_Stu1 where Sname = '张三'\G Explain 命令结果截图： [illegible]