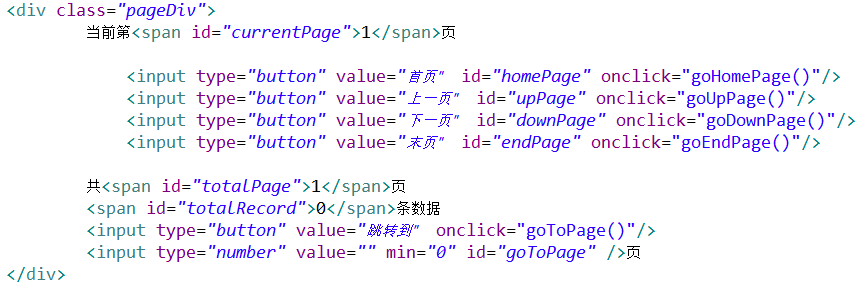
# 项目总结

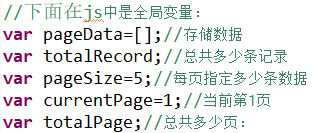
## 一、分页

### 1.js实现前端列表分页





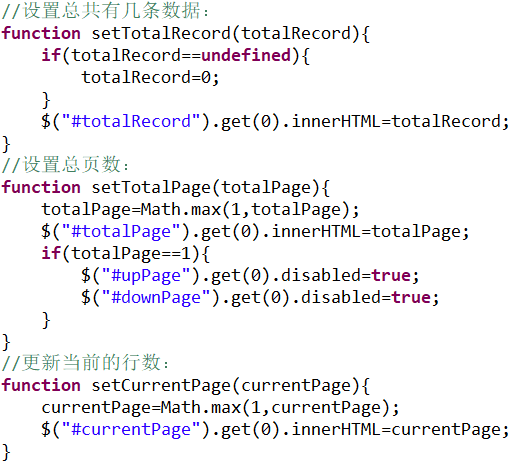
下面是js实现分页的代码：

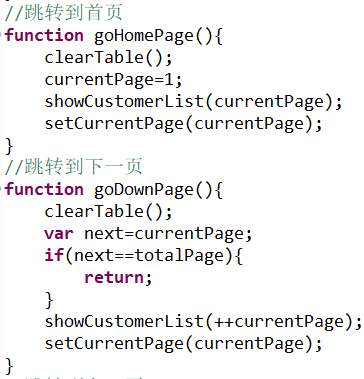


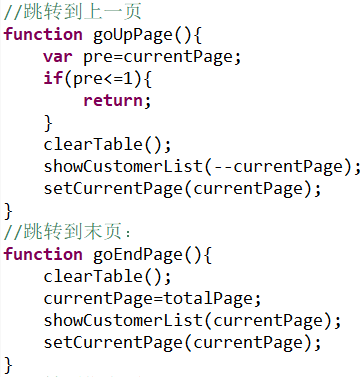


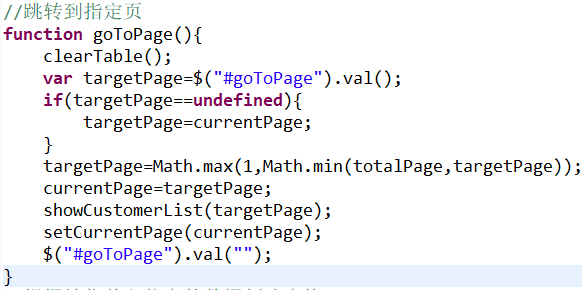


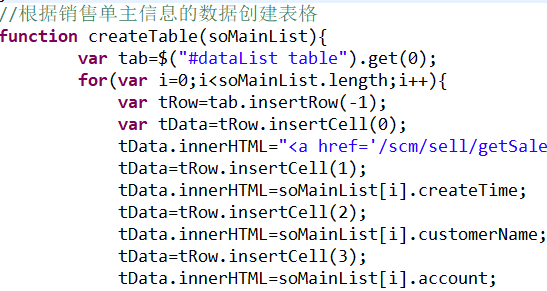


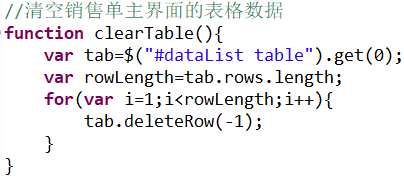








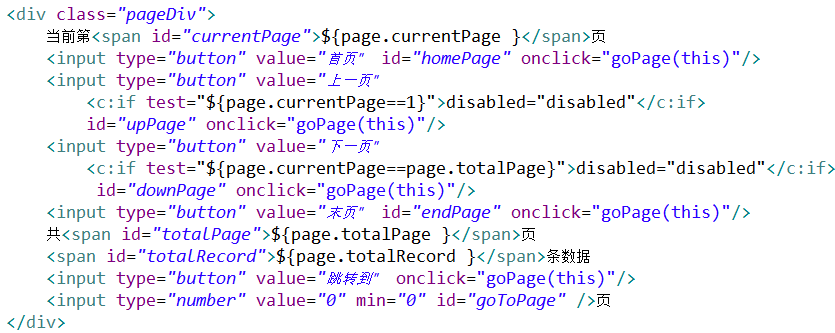


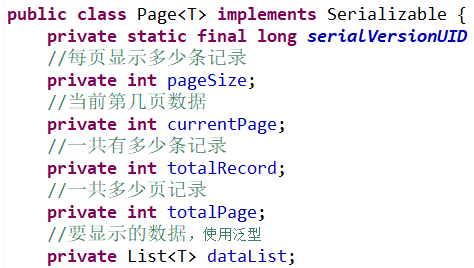


### 2.在后端将数据分好页再传入前端

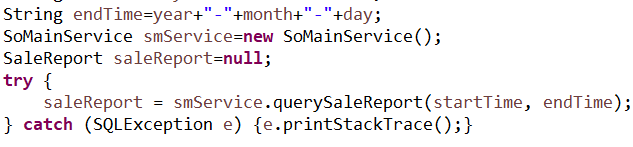
页面点击查询时，将year、month提交到Servlet中。Servlet中进行分页查询数据，并将查询到得数据显示到页面中。



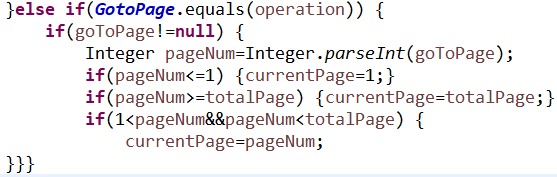






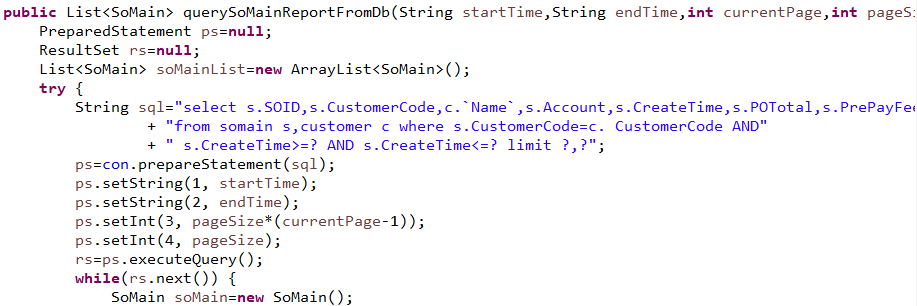








dao层中从数据库中查询指定页指定条数据采用limit m,n。



## 二、其它

### 1.路径后面拼接数据

数据拼接在路径后面时注意单引号和双引号的格式：





自己在路径后面进行数据拼接时，老是会忘记加上个等于号，比如下面这个就忘记了。这会导致作用域中获取不到这个值。**错误写法**：



**正确写法**：



### 2.鼠标点击事件中可以填跳转的路径



## 三、BigDecimal

mysql中得decimal在java中用BigDecimal表示。Java在java.math包中提供的API类BigDecimal，用来对超过16位有效位的数进行精确的运算。float和double只能用来做科学计算或者是工程计算，在商业计算中要用java.math.BigDecimal。**BigDecimal所创建的是对象**，我们不能使用传统的+、-、\*、/等算术运算符直接对其对象进行数学运算，而必须调用其相对应的方法。

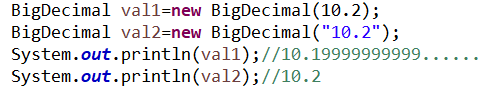
### 1.构造器

**public** **class** BigDecimal **extends** Number **implements** Comparable<BigDecimal> {...}

**public** BigDecimal(String val){...}

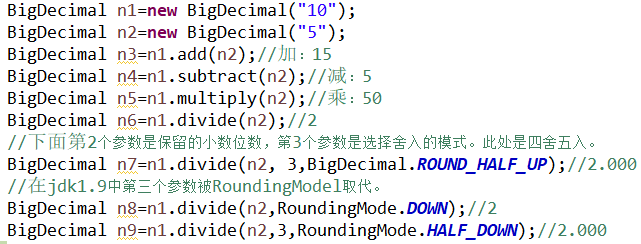
**public** BigDecimal(**double** val){...}

采用构造器创建BigDecimal对象。注意在创建该对象的过程中尽量使用String类型的数据作为参数，因为数据的准确性有保证。而使用double类型的数据作为参数时，不能保证数据的准确性。如下：

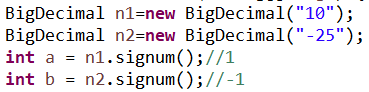


### 2.四则运算

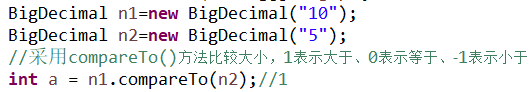
BigDecimal类型的数据进行加、减、乘、除运算分别调用**add()**、**subtract()**、**multiply()**、**divide()**方法等。



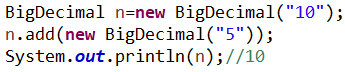
BigDecimal的**signum()**方法返回BigDecimal的正负号。**1表示正值，-1表示负值，0表示0**。



BigDecimal类型的数据比较大小不能直接用>、<、==等比较。因为BigDecimal实现了Comparable接口，所以可以调用**compareTo()**方法进行比较，返回值1表示大于，0表示等于，-1表示小于。



BigDecimal和String 一样具有对象不可变行，一旦赋值就不会再变。



BigDecimal类赋值为0有两种方法：



mysql中的decimal类型的数据在java中对应的数据为BigDecimal类型。将获得的字符串转换成BigDecimal采用如下方法：

