



# 3.5 信息系统的数据呈现

广州市执信中学 林剑辉





# 知识回顾

## 信息系统的 数据处理

页面间的数据传递

根据用户输入查询数据

根据用户输入自动修改、删除数据





信息系统在数据处理之后将以各种各样的方式向系统使用者输出信息，例如打印输出与屏幕输出，其中以屏幕显示方式最为常见。

各种数据经信息系统处理之后，哪些结果需要呈现，数据结果以什么样的方式呈现？





# 学习目标

1. 能够根据现实需求确定需要输出的数据。
2. 理解数据输出样式对信息应用的影响。







本节我们将围绕“图书管理信息系统的数据输出设计”项目展开学习，体验数据的分类查询结果输出，理解适当的数据输出样式对增强信息表达的重要作用。





# 项目任务

任务1：制作最受欢迎的图书排行榜

任务2：优化借阅管理系统的输出





**思考：**经过一段时间的运行，图书借阅管理系统存储了很多关于学生阅读情况的数据，学校如何才能了解哪些类型的图书最受学生欢迎呢？

## 最受欢迎图书排行榜





# 任务一：制作最受欢迎图书排行榜

活动1：设计图书排行榜

活动2：编写最受欢迎的图书排行榜程序







# 活动1：设计图书排行榜

按照不同的方式对学生借阅图书的数据进行排序输出，可以从不同的视角了解学生的阅读情况。比如按照性别分类，可以了解男生、女生分别喜欢阅读哪种类型的图书；按照图书类型分类，可以了解全体学生的阅读偏好。





**思考：**你还可以从哪些不同的视角对图书进行分类排序呢？

序号	分类方式	结果说明
1	性别—图书类型	对比男、女生的阅读偏好
2	图书类型	了解全体学生的阅读偏好
3	年级—图书类型	了解不同年级学生的阅读偏好
4	年级—性别—图书类型	了解不同年级不同性别学生的阅读偏好
5	图书被借阅次数	了解图书的受欢迎程度

表3.5.1 分类查询及其含义



## 活动2：编写最受欢迎的图书排行榜程序

图书被借阅的次数直接反映了图书的受欢迎程度，由此可以对图书质量以及阅读行为进行科学分析，分析结果可以作为向学生提供阅读建议或向学校提供图书更新建议的依据。

**思考：**如何用程序来统计图书借阅次数最多的10本书呢？







### 1. 连接数据库，创建游标

#连接数据库，创建游标

```
DBfile = os.getcwd()+ u"""\图书管理.mdb""" #起始执行目录\数据库文件
#建立一个连接对象，该对象用odbc连接管道连接应用程序和指定的数据库
conn = pyodbc.connect(u"Driver={Microsoft Access Driver (*.mdb, *.accdb)};DBQ=" + DBfile)
cursor = conn.cursor()
```

### 2. 从借阅表中查询图书累计借阅情况

#从借阅表中查询图书累计借阅情况

```
sql = u" SELECT TOP 10 ISBN, COUNT(*) AS [借阅数量] FROM [borrow]
GROUP BY ISBN ORDER BY [借阅数量] DESC"
cursor.execute(sql)
```

### 3. 输出查询结果

#输出查询结果

```
list = cursor.fetchall()
```

### 4. 将数据传递给show\_new.html文件显示

#将数据传递给show\_new.html文件显示

```
return render_to_response('show_new.html', {'book_list':list} )
```





第二步功能是从借阅表中查询图书累计借阅情况，查询语句如下：

#从借阅表中查询图书累计借阅情况

```
sql = u" SELECT TOP 10 ISBN, COUNT(*) AS [借阅数量] FROM [borrow]  
GROUP BY ISBN ORDER BY [借阅数量] DESC"  
cursor.execute(sql)
```

查询语句的语法：

SELECT TOP 10 ISBN, COUNT(\*) AS [借阅数量] FROM [borrow] GROUP BY  
ISBN ORDER BY [借阅数量] DESC

TOP子句：返回记录的数目

根据指定的字段对结果集进行排序，如  
希望降序，则添加DESC关键字

COUNT()：聚合查询函数

对聚合查询的结果集进行分组





# 聚合查询

在SQL语句中，**聚合查询是指通过包含一个聚合函数来汇总多个数据项的值**。聚合函数对一组值执行计算并返回单一的值。

常见的聚合函数包括**求和函数SUM()**、**求平均值函数AVG()**、**求最大值函数MAX()**、**求最小值函数MIN()**和**计数函数COUNT()**等。

聚合函数经常**与SELECT语句的GROUP BY子句**一同使用。





# TOP子句

在SQL语句中，TOP子句用于规定要返回的记录数目。

TOP子句以两种方式指定输出的记录数：一种是直接指定输出的数量；另一种是指定记录的百分比。







SELECT \* FROM [学生表] #从学生表中查询所有记录

学号	密码	姓名	性别	年龄	年级	班级
001	001	李晓峰	男	16	高一	01
002	002	张玉龙	男	16	高一	01
003	003	刘玉菲	女	16	高一	02
004	004	何飞	男	16	高一	03
005	005	崔晓云	女	17	高二	01
006	006	罗春	男	17	高二	02
007	007	王盈盈	女	17	高二	02
008	008	赵羽然	女	18	高三	03
009	009	林思尧	女	16	高一	02
010	010	岳珊珊	女	17	高二	03
011	011	梁正亮	男	18	高三	01
012	012	陈欣	女	16	高二	02







SELECT TOP 2 \* FROM [学生表] #从学生表中选择前两条记录

学号	密码	姓名	性别	年龄	年级	班级
001	001	李晓峰	男	16	高一	01
002	002	张玉龙	男	16	高一	01

SELECT TOP 50 PERCENT \* FROM [学生表] #从学生表中选择前50%的记录

学号	密码	姓名	性别	年龄	年级	班级
001	001	李晓峰	男	16	高一	01
002	002	张玉龙	男	16	高一	01
003	003	刘玉菲	女	16	高一	02
004	004	何飞	男	16	高一	03
005	005	崔晓云	女	17	高二	01
006	006	罗春	男	17	高二	02
*						





# ORDER BY子句

在SQL语句中，**ORDER BY**子句用于根据指定的列对结果集进行排序。**ORDER BY**子句默认按照**升序**对记录进行排序。如果需要按照**降序**对记录进行排序，可以使用**DESC**关键字。





学号	密码	姓名	性别	年龄	年级	班级
001	001	李晓峰	男	16	高一	01
002	002	张玉龙	男	16	高一	01
003	003	刘玉菲	女	16	高一	02
004	004	何飞	男	16	高一	03
005	005	崔晓云	女	17	高二	01
006	006	罗春	男	17	高二	02
007	007	王盈盈	女	17	高二	02
008	008	赵羽然	女	18	高三	03
009	009	林思尧	女	16	高一	02
010	010	岳珊珊	女	17	高二	03
011	011	梁正亮	男	18	高三	01
012	012	陈欣	女	16	高二	02







SELECT \* FROM [学生表] ORDER BY 姓名 #按姓名升序输出所有记录

学号	密码	姓名	性别	年龄	年级	班级
012	012	陈欣	女	16	高二	02
005	005	崔晓云	女	17	高二	01
004	004	何飞	男	16	高一	03
001	001	李晓峰	男	16	高一	01
011	011	梁正亮	男	18	高三	01
009	009	林思尧	女	16	高一	02
003	003	刘玉菲	女	16	高一	02
006	006	罗春	男	17	高二	02
007	007	王盈盈	女	17	高二	02
010	010	岳珊珊	女	17	高二	03
002	002	张玉龙	男	16	高一	01
008	008	赵羽然	女	18	高二	03







SELECT \* FROM [学生表] ORDER BY 姓名 **DESC** #按姓名降序输出所有记录

学号	密码	姓名	性别	年龄	年级	班级
008	008	赵羽然	女	18	高三	03
002	002	张玉龙	男	16	高一	01
010	010	岳珊珊	女	17	高二	03
007	007	王盈盈	女	17	高二	02
006	006	罗春	男	17	高二	02
003	003	刘玉菲	女	16	高一	02
009	009	林思尧	女	16	高一	02
011	011	梁正亮	男	18	高三	01
001	001	李晓峰	男	16	高一	01
004	004	何飞	男	16	高一	03
005	005	崔晓云	女	17	高二	01
012	012	陈欣	女	16	高二	02



**练一练：**请参考课本P102页代码，编写最受欢迎的图书排行榜程序。查看你所制作的最受欢迎图书排行榜网页，观察这一网页反馈了哪些信息？

```
#连接数据库，创建游标
DBfile = os.getcwd()+ u"""\图书管理.mdb""" #起始执行目录\数据库文件
#建立一个连接对象，该对象用odbc连接管道连接应用程序和指定的数据库
conn = pyodbc.connect(u"Driver={Microsoft Access Driver (*.mdb,
*.accdB)};DBQ=" + DBfile)
cursor = conn.cursor()

#从借阅表中查询图书累计借阅情况
sql = u" SELECT TOP 10 ISBN, COUNT(*) AS [借阅数量] FROM [borrow]
GROUP BY ISBN ORDER BY [借阅数量] DESC"
cursor.execute(sql)

#输出查询结果
list = cursor.fetchall()

#将数据传递给show_new.html文件显示
return render_to_response('show_new.html', {'book_list':list})
```



欢迎您, !

今日:

[图书信息录入](#)[学生信息录入](#)[借阅信息查询](#)[图书查询与借阅](#)[借阅排行榜查询](#)

## 最受欢迎的图书排行榜

ISBN	书名	作者	图书类型	借阅数量
9787115309723	宇宙之书 — 从托勒密、爱因斯坦 坦到多重宇宙	约翰·D.巴罗	科普	7
9787100119696	西游记	吴承恩	小说	7
9787302404330	山海经	孙见坤	文学	5
9787562492368	天文之书	吉姆·贝尔	科普	3
9787519300203	中国通史“慢读”系列	吕思勉	历史	3
9787507520798	孙中山传	李菁	人物传记	3
9787553323350	资治通鉴	司马光	历史	2
9787537963268	中国的地形·壮美画卷	贾文毓	地理	2
9787020008728	三国演义	罗贯中	小说	2
9787514345896	力学原来这么有趣!	大井喜久夫	科普	1







**思考：**数据输出的内容和形式能够体现出信息系统是否优良，如何才能让数据输出更优化呢？





## 任务二：优化借阅管理系统的输出

活动1：设计优化输出的项目

活动2：优化输出的程序实现





# 活动1：设计优化输出的项目

无论是用户登录、预约图书，还是图书录入、图书查询，信息系统的各个功能都需要通过数据输出向使用者描述工作状态或执行结果。







**思考：** 请根据表3.5.4, 分析在已经完成的图书借阅系统中，有哪些地方需要向用户反馈数据，如何反馈能更加有效。

序号	页面	需要反馈的数据	反馈形式
1	录入图书	录入成功与否	弹出对话框
2	用户登录	登录成功与否	弹出对话框
3	图书查询	欢迎XXX使用图书借阅系统！	在页面中显示
4	图书预约	预约成功与否	弹出对话框
5	图书排行榜	图书借阅数量排行情况	在页面中显示
6	图书或用户信息录入	输入文本框为空	提示“请填写此字段”





## 活动2：优化输出的程序实现

根据设计，用户登录成功之后将自动进入图书查询页面。如果能够在图书查询页面显示当前日期、“欢迎XXX使用图书借阅系统”等信息将会提升用户的使用体验。

 图书管理系统

欢迎您，！今日：

[图书信息录入](#) | [学生信息录入](#) | [借阅信息查询](#) | [图书查询与借阅](#) | [借阅排行榜查询](#)

书 名  出版时间

库存图书列表

选择	ISBN	书名	图书类型	出版时间	库存数量
----	------	----	------	------	------

Copyright © 2019 MyLibrary All Rights Reserved. 学校图书馆 版权所有





1. 在view.py中获取用户登录的session信息

2. 获取当前日期

3. 将当前日期时间和欢迎信息提供给HTML页面

4. 在show.html中显示当前日期和欢迎信息

#在show.html中显示当前日期和欢迎信息

`<table style="width: 400px;">` `<!-- 定义一个宽为400像素的表格 -->`

`<tr>` `<!-- 表格的第一行 -->`

`<td>今天是: </td>` `<!-- 第一行的第一个单元格 -->`

`<td> {{today}} </td>` `<!-- 第一行的第二个单元格 -->`

`</tr>` `<!-- 第一行结束 -->`

`<tr>`

`<td>欢迎</td>` CSS中的style属性

`<!-- 以红色加粗的方式显示登录用户 -->`

`<td> <font style='color:red; font-weight: bold;'> {{userID}}</font>`  
`使用图书借阅系统! </td>`

模板变量

`</tr>`

`</table>` HTML中的表格标签





# HTML的表格

在HTML中，用`<table></table>`来定义一个表格。表格包含若干行`<tr></tr>`，每一行包括若干单元格`<td></td>`。

思考：运行图片中代码，网页中将显示什么？

```
<table>
<tr>
<td>第一行第一格</td>
<td>第一行第二格</td>
<td>第一行第三格</td>
</tr>
<tr>
<td>第二行第一格</td>
<td>第二行第二格</td>
<td>第二行第三格</td>
</tr>
</table>
```

两行三列的表格





# CSS

CSS (Cascading Style Sheets, 层叠样式表) 是一种用来表现HTML对象显示属性的语言。HTML标记中的style属性即为该标记的CSS代码。

style属性的取值与其他属性的取值有所不同。其他属性的值为具体的数据。style属性的值为一段代码，能够为该标记的一个或多个属性赋值，更为精细地描述对象的显示状态。





# MVC框架

模型-视图-控制器 (MVC) 把代码的定义和数据访问的方法 (模型, model) 与请求逻辑 (控制器, control) 以及用户接口 (视图, view) 分开来。这种设计模式关键的优势在于各种组件都是松散结合的。

这样, 页面设计师可以改变HTML页面的样式而不用接触Python代码; 数据库管理员只需要关心数据库和数据表的结构设计而不用考虑数据如何访问。由此, 从事系统开发的人员能够更好地进行分工合作, 提高开发效率。

图书管理信息系统的开发



HTML+Python+Acess



用于制作  
信息系统的  
界面



用于制作  
信息处理的  
程序



用于数  
据管理





**练一练：**请参考课本P105页代码，编写程序，实现输出的优化。

```
#在view.py中获取用户登录的session信息
userID=request.session.get(u'userID','没有用户')
#获取当前日期
today=time.strftime('%Y-%m-%d',time.localtime(time.time()))
#将当前日期时间和欢迎信息提供给HTML页面
return render_to_response('show.html', {'userID': userID,
'today': today } )

#在show.html中显示当前日期和欢迎信息
<table style="width: 400px;"> <!-- 定义一个宽为400像素的表格 -->
<tr> <!-- 表格的第一行 -->
<td>今天是: </td> <!-- 第一行的第一个单元格 -->
<td> {{today}} </td> <!-- 第一行的第二个单元格 -->
</tr> <!-- 第一行结束 -->
<tr>
<td>欢迎</td>
<!-- 以红色加粗的方式显示登录用户 -->
<td> <font style='color:red; font-weight: bold;'> {{userID}}</font>使用图书借阅系统! </td>
</tr>
</table>
```





欢迎 **李晓峰** 使用图书借阅系统！

今日：2022-04-24

[图书信息录入](#)

[学生信息录入](#)

[借阅信息查询](#)

[图书查询与借阅](#)

[借阅排行榜查询](#)

书 名

出版时间

查 找

预 约

### 库存图书列表

选择

ISBN

书名

图书类型

出版时间

库存数量





# 课后小结

- 信息系统的数据呈现
  - 确定需要输出的数据
    - 聚合查询
    - GROUP BY子句
    - TOP子句
    - ORDER BY子句
  - 数据的输出样式
    - html
    - CSS







# 谢谢大家

