# 实验任务：数据可视化

本项目中使用Flask框架将数据从Mysql中读出，使用Echarts进行图表展示。

表 3‑2 Mysql中的表结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 表名 | 字段 | 数据类型 | 备注 |
| 1 | 借款金额分布表：  loan\_amount | zone | varchar(50) | 借款金额区间 |
| num | Int | 人数 |
| 2 | 借款等级分布表：  loan\_grade | grade | varchar(5) | 借款等级 |
| subgrade | varchar(5) | 借款子等级 |
| num | Int | 人数 |
| 3 | 借款等级与借款金额关系表：  grade\_amount | grade | varchar(5) | 借款等级 |
| amount | varchar(50) | 借款金额区间 |
| num | Int | 人数 |
| 4 | 借款金额与工作年限关系表：  amount\_wkyear | wkyears | varchar(50) | 工作年限区间 |
| amount | varchar(50) | 借款金额区间 |
| num | Int | 人数 |
| 5 | 借款金额与年收入关系表：  amount\_salary | salary | varchar(50) | 年收入区间 |
| amount | varchar(50) | 借款金额区间 |
| num | Int | 人数 |
| 6 | 借款金额与房屋所有权状态关系表：  amount\_house | house | varchar(50) | 房屋状态 |
| amount | varchar(50) | 借款金额区间 |
| num | Int | 人数 |

## 步骤一：创建项目

**在Pycharm中创建项目FlaskTest，项目结构如下：**

1. 在该项目中创建文件夹static，将Echarts的js文件放入该文件夹下；
2. 在该项目中创建文件夹templates，用于保存网页文件；
3. 在根目录下创建python文件app.py；

**整个项目结构如下：**

文本

描述已自动生成

图 3‑5 项目结构

使用pip install指令安装pymysql，numpy包。

## 步骤二：借款金额分布可视化

本实验将loan\_amount表中的数据用柱状图展示出来，因此需要在app.py文件中引入flask包，并创建路由，引入Echarts插件，进行可视化展示。

### 在copy.py文件中，加入以下代码：

# 导入Flask，render\_template类

from flask import Flask, render\_template

import pymysql, numpy as np

# Flask类接收一个参数\_\_name\_\_

app = Flask(\_\_name\_\_)

# 装饰器的作用是将路由映射到视图函数

# 借款金额分布——柱状图

@app.route('/1')

def bar():

data = GetData()

print(data)

xData = []

yData = []

for i in data:

xData.append(i[0])

yData.append(i[1])

return render\_template("bar.html", data={"xData": xData, "yData": yData})

pass

def GetData():

# 连接数据库

connect = pymysql.connect(

host="localhost",

user="root",

password="@wsw0420",

database="test",

charset="utf8"

)

cursor = connect.cursor()

# 1借款金额分布

cursor.execute("SELECT \* FROM loan\_amount ORDER BY CAST(SUBSTRING\_INDEX(zone,'-',1) AS SIGNED);")

# 从mysql提取数据到data

data = cursor.fetchall()

print(data)

cursor.close()

connect.close()

return [i for i in data]

# Flask应用程序实例的run方法启动WEB服务器

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

### 在templates文件夹中新建html文件，文件名为bar.html，填入以下代码：

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="UTF-8">  
 <title>借款金额分布</title>  
 <script src="../static/echarts.min.js"></script>  
</head>  
  
<body>  
 <div id="root" style="width: 1000px;height: 600px"></div>  
 <script>  
 root = ***document***.getElementById("root")  
 echart = echarts.init(root)  
 echart.setOption({  
 title :{  
 text :"借款金额分布图",  
 left:"center",  
 textStyle:{  
 fontSize:"200%"  
 }  
 },  
 legend:{  
 left:"right",  
 top:"center",  
 },  
 xAxis:{  
 type:"category",  
 data:{{data.xData|safe}},  
 name:"借款金额：美元"  
 },  
 yAxis:{  
 name:"人数"  
 },  
 series:[{  
 name:"人数",  
 type:"bar",  
 label:{  
 show:true,  
 position:'top'  
 },  
 data:{{data.yData}}  
 }]  
 })  
 </script>  
</body>  
  
</html>

### 运行程序，在浏览器中打开网址<http://127.0.0.1:5000/1>，可以看到可视化结果如下图所示：

截图里有图片

描述已自动生成

图表, 条形图

描述已自动生成

图 3‑6 借款金额分布可视化结果图

## 步骤三：借款等级分布可视化

loan\_grade表中有三个字段，分别是借款等级、子等级、人数，本实验要将每个等级的子等级显示出来，需要使用堆叠条形图进行可视化。

### 在app.py中加入如下代码：

# 借款等级分布图——堆叠条形图

@app.route("/2")

def double\_bar():

data = GetData()

# print(data)

fData = []

vData = []

grade = [1, 2, 3, 4, 5]

for j in range(0, 5):

for i in range(j \* 7, j \* 7 + 7):

vData.append(data[i][2])

# print(fData[j])

fData.append({"category": grade[j], "value": vData})

vData = []

return render\_template("double\_bar.html", data=fData)

pass

**注意：代码需要粘贴到app.run方法前**

**将获取数据的语句改为如下代码：**

# 2借款等级分布 需要注释掉上一条命令

cursor.execute("SELECT \* FROM loan\_grade ORDER BY RIGHT(subgrade,1),grade")

### 在template文件夹中新建double\_bar.html文件，在文件中加入如下代码：

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>借款等级分布</title>

<script src="../static/echarts.min.js"></script>

</head>

<body>

<div id="root" style="width: 1000px;height: 600px">

</div>

<script>

root = document.getElementById("root")

echart = echarts.init(root)

var series1=[];

var length={{data|tojson}}.length

for(var i=0;i<length;i++){

series1.push({

name:i+1,

type:'bar',

stack:'总量',

label:{

show:true,

position:'insideRight'

},

data:{{data|tojson}}[i].value

})

}

echart.setOption({

title :{

text :"借款等级分布图",

left:"center",

textStyle:{

fontSize:"200%"

}

},

tooltip: {

trigger: 'axis',

axisPointer: {

type: 'shadow'

}

},

legend:{

left:"center",

top:"bottom",

data:['1','2','3','4','5'],

name:'子等级'

},

xAxis:{

type:'value',

name:'人数'

},

yAxis:{

type:'category',

data:['A','B','C','D','E','F','G'],

name:'等级'

},

series:series1

})

</script>

</body>

</html>

### 在浏览器中输入地址<http://127.0.0.1:5000/2>，查看结果如下图：

图表

描述已自动生成

图 3‑7借款等级分布可视化图

## 步骤四：借款等级与借款金额的关联关系可视化

表grade\_amount中包含三个字段借款等级、借款金额区间、人数，也需要使用堆叠条状图将数据进行可视化，显示每个借款金额区间内的各借款等级的人数分布。

### 在app.py文件中加入以下代码：

注意：代码需要粘贴到app.run方法前

# 借款等级与借款金额分布图——堆叠条形图

@app.route("/3")

def grade\_amount():

data = GetData()

fData = []

vData = []

grade = ['A', 'B', 'C', 'D', 'E', 'F', 'G']

for j in range(0, len(grade)):

for i in range(j \* 5, j \* 5 + 5):

vData.append(data[i][2])

# print(fData[j])

fData.append({"category": grade[j], "value": vData})

vData = []

return render\_template("grade\_amount.html", data=fData)

pass

**将获取数据处的代码改为：**

# 3借款等级与借款金额的关系 需要注释掉上一条命令

cursor.execute("SELECT \* FROM grade\_amount ORDER BY grade,CAST(SUBSTRING\_INDEX(amount,'-',1) AS SIGNED)")

### 在template文件夹中新建grade\_amount.html文件，文件内容写入以下代码：

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>借款等级与借款金额关系</title>

<script src="../static/echarts.min.js"></script>

</head>

<body>

<div id="root" style="width: 1000px;height: 600px">

</div>

<script>

root = document.getElementById("root")

echart = echarts.init(root)

console.log({{data|tojson}})

var series1=[];

var zone=[];

var length={{data|tojson}}.length

for(var i=0;i<length;i++){

zone.push({{data|tojson}}[i].category)

}

for(var i=0;i<length;i++){

series1.push({

name:zone[i],

type:'bar',

stack:'总量',

data:{{data|tojson}}[i].value

})

}

console.log(series1)

echart.setOption({

title :{

text :"借款等级与借款金额关系图",

left:"center",

textStyle:{

fontSize:"200%"

}

},

tooltip: {

trigger: 'axis',

axisPointer: {

type: 'shadow'

}

},

legend:{

left:"center",

top:"bottom",

data:zone,

name:'等级',

show:true

},

xAxis:{

type:'value',

name:'人数'

},

yAxis:{

type:'category',

data:['0-1000','1000-5000','5000-10000','10000-20000','20000-50000'],

name:'借款金额区间'

},

series:series1

})

</script>

</body>

</html>

### 在浏览器中输入<http://127.0.0.1:5000/3>，显示结果如下：

图表, 条形图

描述已自动生成

图 3-8借款等级与借款金额的关联关系可视化图

## 步骤五：借款金额与工作年限、年收入的关联关系可视化

表amount\_wkyear中包含三个字段工作年限、借款金额区间、人数，也需要使用堆叠条状图将数据进行可视化，显示每个借款金额区间内的工作年限的人数分布。

1. 在Start.py文件中加入以下代码：

注意：代码需要粘贴到app.run方法前

# 借款金额与工作年限分布图——堆叠条形图

@app.route("/4")

def amount\_wkyears():

    data=GetData()

    fData=[]

    vData= []

    tmp=[]

    for i in data:

        tmp.append(i[0])

    wkyear = np.unique(tmp)

    for j in range(0, len(wkyear)):

        for i in range(j\*5,j\*5+5):

            vData.append(data[i][2])

        fData.append({"category":wkyear[j],"value":vData})

        vData=[]

    return render\_template("amount\_wkyears.html",data=fData)

    pass

将获取数据处的代码更改为：

# 4借款金额与工作年限的关系 需要注释掉上一条命令

    cursor.execute("SELECT \* FROM amount\_wkyear ORDER BY CAST( LEFT(wkyears,2) AS SIGNED),CAST(SUBSTRING\_INDEX(amount,'-',1) AS SIGNED)")

2. 在template文件夹中新建HTML文件，文件名为amount\_wkyears.html，文件内容写入以下代码：

<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <title>借款金额与工作年限关系图</title>

    <script src="../static/echarts.min.js"></script>

</head>

<body>

    <div id="root" style="width: 1000px;height: 600px"></div>

    <script>

        root = document.getElementById("root")

        echart = echarts.init(root)

        console.log({{data|tojson}})

        var series1=[];

        var zone=[];

        var length={{data|tojson}}.length

        for(var i=0;i<length;i++){

            zone.push({{data|tojson}}[i].category)

        }

        for(var i=0;i<length;i++){

            series1.push({

                name:zone[i],

                type:'bar',

                stack:'总量',

                data:{{data|tojson}}[i].value

            })

        }

        console.log(series1)

        echart.setOption({

            title :{

                text :"借款金额与工作年限关系图",

                left:"center",

                textStyle:{

                    fontSize:"200%"

                }

            },

             tooltip: {

                trigger: 'axis',

                axisPointer: {

                    type: 'shadow'

                }

            },

            legend:{

                left:"center",

                top:"bottom",

                data:zone,

                name:'等级',

                show:true

            },

            xAxis:{

                type:'value',

                name:'人数'

            },

            yAxis:{

                type:'category',

                data:['0-1000','1000-5000','5000-10000','10000-20000','20000-50000'],

                name:'借款金额区间'

            },

            series:series1

        })

    </script>

</body>

</html>

3. 在浏览器中输入<http://127.0.0.1:5000/4>，显示结果如下：

图表, 条形图

描述已自动生成

图 3‑9借款金额与工作年限关联关系可视化图

表amount\_salary中包含三个字段年收入区间、借款金额区间、人数，需要使用堆叠条形图进行可视化。步骤如下：

1. 在Start.py文件中加入以下代码：

注意：代码需要粘贴到app.run方法前

# 借款金额与年收入分布图——堆叠条形图

@app.route("/5")

def amount\_salary():

    data=GetData()

    fData=[]

    vData= []

    tmp=[]

    for i in data:

        tmp.append(i[0])

    salary = np.unique(tmp)

    salary[3],salary[4]=salary[4],salary[3]

    for j in range(0, len(salary)):

        for i in range(j\*5,j\*5+5):

            if(j>0):

                m=i-1

                vData.append(data[m][2])

            else:

                vData.append(data[i][2])

        fData.append({"category":salary[j],"value":vData})

        vData=[]

    return render\_template("amount\_salary.html",data=fData)

    pass

将获取数据处的代码更改如下：

# 5借款金额与年收入的关系 需要注释掉上一条命令

cursor.execute("SELECT \* FROM amount\_salary ORDER BY CAST(SUBSTRING\_INDEX(salary,'\_',1) AS SIGNED),CAST(SUBSTRING\_INDEX(amount,'\_',1) AS SIGNED)")

2. 在template文件夹中新建HTML文件，文件名为amount\_salary.html，文件内容写入以下代码：

<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <title>借款金额与年收入关系图</title>

    <script src="../static/echarts.min.js"></script>

</head>

<body>

    <div id="root" style="width: 1000px;height: 600px"></div>

    <script>

        root = document.getElementById("root")

        echart = echarts.init(root)

        console.log({{data|tojson}})

        var series1=[];

        var zone=[];

        var length={{data|tojson}}.length

        for(var i=0;i<length;i++){

            zone.push({{data|tojson}}[i].category)

        }

        for(var i=0;i<length;i++){

            series1.push({

                name:zone[i],

                type:'bar',

                stack:'总量',

                data:{{data|tojson}}[i].value

            })

        }

        console.log(series1)

        echart.setOption({

            title :{

                text :"借款金额与年收入关系图",

                left:"center",

                textStyle:{

                    fontSize:"200%"

                }

            },

             tooltip: {

                trigger: 'axis',

                axisPointer: {

                    type: 'shadow'

                }

            },

            legend:{

                left:"center",

                top:"bottom",

                data:zone,

                name:'等级',

                show:true

            },

            xAxis:{

                type:'value',

                name:'人数'

            },

            yAxis:{

                type:'category',

                data:['0-1000','1000-5000','5000-10000','10000-20000','20000-50000'],

                name:'借款金额区间'

            },

            series:series1

        })

    </script>

</body>

</html>

3. 在浏览器中输入<http://127.0.0.1:5000/5>，显示结果如下：

图表, 日程表, 条形图

描述已自动生成

图 3‑10

借款金额与年收入的关联关系可视化图

## 步骤六：借款金额与房屋所有权状态的关联关系可视化

表amount\_house中包含三个字段房屋状态、借款金额区间、人数，本实验使用堆叠柱状图将数据进行可视化，显示每个借款金额区间内的各房屋所有权状态的人数分布。

1. 在Start.py文件中加入以下代码：

注意：代码需要粘贴到app.run方法前

# 借款金额与房屋所有权状态——堆叠柱状图

@app.route("/6")

def amount\_house():

    data=GetData()

    fData=[]

    vData= []

    tmp=[]

    for i in data:

        tmp.append(i[0])

    house = np.unique(tmp)

    for j in range(0, len(house)):

        # 由于“OTHER”状态下的“0-1000"没有值因此需要在该处增加数据0

        for i in range(j\*5,j\*5+5):

            if(j==1 and i==5):

                vData.append(0)

            if(j==1 and i>5):

                i=i-1

            if(j>1):

                i=i-1

            vData.append(data[i][2])

        fData.append({"category":house[j],"value":vData})

        vData=[]

    return render\_template("amount\_house.html",data=fData)

    pass

将获取数据处代码更改如下：

# 6借款金额与房屋所有权状态的关系 需要注释掉上一条命令

cursor.execute("SELECT \* FROM amount\_house ORDER BY house,CAST(SUBSTRING\_INDEX(amount,'\_',1) AS SIGNED)")

2. 在template文件夹中新建HTML文件，文件名为amount\_house.html，文件内容写入以下代码：

<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <title>借款金额与房屋所有权状态关系图</title>

    <script src="../static/echarts.min.js"></script>

</head>

<body>

    <div id="root" style="width: 1000px;height: 600px">

    </div>

    <script>

        root = document.getElementById("root")

        echart = echarts.init(root)

        var length={{data|tojson}}.length

        var house=[]

        var series1=[];

        for(var i=0;i<length;i++){

            house.push({{data|tojson}}[i].category);

        }

        var zone=['0-1000','1000-5000','5000-10000','10000-20000','20000-50000']

        for(var i=0;i<length;i++){

            series1.push({

                name:house[i],

                type:'bar',

                data:{{data|tojson}}[i].value,

                label:{

                    show:true,

                    position:'top'

                }

            })

        }

        echart.setOption({

            title :{

                text :"借款金额与房屋状态关系图",

                left:"center",

                textStyle:{

                    fontSize:"200%"

                }

            },

            legend:{

                left:"center",

                top:"bottom",

                data:house

            },

            xAxis:[{

                type:'category',

                data:zone

            }],

            yAxis:[{

                type:'value'

            }],

            series:series1

        })

    </script>

</body>

</html>

3. 在浏览器中输入<http://127.0.0.1:5000/6>，显示结果如下：

图表, 条形图

描述已自动生成

图 3‑11借款金额与房屋状态关联关系可视化图

## 备注：

备注：CAST：<https://www.yiibai.com/mysql/cast.html>