

[https://blog.csdn.net/fly\\_tom/article/details/79515797](https://blog.csdn.net/fly_tom/article/details/79515797)

//项目备注

/\*

\*该类的功能以及特点描述：超市管理系统的 User的实体类

\*该类是否被编译测试：否

\*@see(与之相关联的类)： 来源：

\* 中间：

\* 去处：

\*开发公司或单位：成007个人

\*版权：成007

\*@author(作者)：成007

\*@since (该文件使用的jdk)：JDK1.8

\*@version (版本)：1.0

\*@数据库查询方式：mysql+Hibernate

\*@date(开发日期)：2018/5/20

\*改进：

\*最后更改日期：

\*/

1.快速生成映射文件

2.逆向建表

解决方法：

现在数据库中建表，myelipse连接数据库，通过Hibernate face工具自动生成映射文件和实体类。

### 3.awt表格和文本域的关联

```
JTable table=new JTable();
```

//用model给table填充数据，vector给model填充数据，vector给vector填充数据

//因为第一参数接收的是二维的参数，所以产生了vector给vector填充数据

```
Vector<String>columnNames=new Vector<String>();
```

```
columnNames.add("商品编号");
```

```
columnNames.add("名称");  
columnNames.add("规格");  
columnNames.add("单位");  
columnNames.add("数量");  
columnNames.add("售价");  
columnNames.add("折扣");  
columnNames.add("金额");
```

```
//定义二维部分
```

```
Vector<Vector<String>>tableValue =new Vector<Vector<String>>();//定  
义表格数据向量
```

```
//定义一行的数据类容
```

```
Vector<String>row1=new Vector<String>();  
row1.add("1");  
row1.add("2");  
row1.add("3");  
row1.add("4");  
row1.add("5");  
row1.add("6");  
row1.add("7");  
row1.add("8");  
tableValue.add(row1);
```

```
//修改table(增加)即可
```

```
DefaultTableModel model = new  
DefaultTableModel(tableValue,columnNames);  
table.setModel(model);  
JScrollPane scrollPane = new JScrollPane(table);  
tabbedPane.add(scrollPane);  
contentPane.add(tabbedPane);
```

### 获取鼠标选定的表中元素

```
int[] selection = table.getSelectedRows();
```

```
for (int i = 0; i < selection.length; i++) {  
    selection[i] = table.convertRowIndexToModel(selection[i]);  
}
```

```
int col = jTable1.getSelectedColumn();//获取选中的列号  
int row = jTable1.getSelectedRow();//获取选中的行号  
DefaultTableModel model = (DefaultTableModel) jTable1.getModel();//获取  
defaulttablemodel  
Object val = model.getValueAt(row, col);//根据行号和列号，获取某个单元格的值  
//其他  
model.removeRow(index);//删除某行  
model.setValueAt(value, row, col);//修改某单元格的值
```

#### 4.时间实时获取并显示在文本框上

```
private String DEFAULT_TIME_FORMAT = "yyyy-MM-dd hh:mm:ss";  
private String time;  
private int ONE_SECOND = 1000;
```

//调用这个内部方法

```
/**  
 * 这个方法创建 a timer task 每秒更新一次 the time  
 * 参数三 单位是1000毫秒 即设置的刷新闻隔  
 */  
public void configTimeArea() {  
    Timer tmr = new Timer();  
    tmr.scheduleAtFixedRate(new JLabelTimerTask(), new Date(), ONE_SECOND);  
}  
/**  
 * Timer task 更新时间显示区
```

```

    *
    */
//这是一个内部类
protected class JLabelTimerTask extends TimerTask {
    SimpleDateFormat dateFormatter = new SimpleDateFormat(
        DEFAULT_TIME_FORMAT);

    @Override
    public void run() {
        time = dateFormatter.format(Calendar.getInstance().getTime());
        textField_5.setText(time);
    }
}

```

## 5.监听一个文本框的变化

```

textField_1.addKeyListener(new KeyAdapter() {
    @Override
    public void keyReleased(KeyEvent e) {

        String text=textField_1.getText().trim();

        textField.setText(time+text);

    }
});

```

## 6.zxing 识别条形码

<https://blog.csdn.net/txadf/article/details/52152924>