

类名: MainUI

说明: 主体界面, 用于管理策略和书本

1、属性:CardLayout cl;

说明: 用于切换AddBookUI和StrategyUI

2、方法:JFrame createTpyeFrame()

说明: 用于生产书类管理界面, 用于增加新的书类

3、方法:JMenu createFileMenu()

说明: 用于生成主UI的菜单, 菜单里有初始化、购买书和管理书类的功能

类名: AddBookUI

说明: 管理图书界面, 用于增加、删除、查阅图书信息

1、属性:DefaultTableModel tableModel

说明: 用于更方便的管理表格

2、属性:JTable BookTable=BookCatalogTable.createTable(tableModel);

说明: 存储所有图书数据的表格

3、方法:void addType(String typename)

说明: 用于将新增的类型加入下拉框中

4、方法: void AddBook()

说明: 根据UI信息增加图书

5、方法: void DelBook()

说明: 根据UI信息删除图书

6、方法:void InitBook()

说明: 初始化图书

类名: StrategiesUI

说明: 生产管理优惠策略的界面, 用于增加、删除、更改、查阅优惠策略, 同时刷新所有图书表中图书信息

1、属性:JTable StrategyTable

说明: 存储所有策略的表格

2、属性:DefaultTableModel tableModel

说明: 更方便于管理表格

3、方法: void update(int selectedRow)

说明: 更新表格中某一行的策略zs信息

4、方法: void addSimpleStrategy()

说明: 根据UI信息增加策略

5、方法: void DelStrategy()

说明: 根据UI信息删除策略

6、方法:void InitBook()

说明: 初始化策略

类名: BuyUI

说明: 用于生产购买书的界面, 同时给购物车传送信息

1、属性:ShoppingCarUI subframe=new ShoppingCarUI(clientname);

说明: 附属于BuyUI的购物车界面

2、属性:DefaultTableModel tableModel

说明: 更方便于管理表格

3、方法: `JTable BookTable=BookCatalogTable.createTable(tableModel);`

说明: 显示所有的图书信息

4、方法: `void BuyBook()`

说明: 购买在UI中选中的图书, 在购物车中更新

5、方法: `boolean ValidInput(int copies)`

说明: 判断输入的购买个数是否合法

类名: ShoppingCarUI

说明: 用于生成购物车界面及管理购物车

1、属性:JTable ShopTable

说明: 购物表, 显示购买书的信息和数量

2、属性:DefaultTableModel tableModel

说明: 更方便于管理表格

3、方法: `JTable BookTable=BookCatalogTable.createTable(tableModel);`

说明: 显示所有的图书信息

4、方法: `void AddBook(String isbn,String title,int copies,double price)`

说明: 根据传来的参数, 计算单次购买的钱, 更新总价和购物表

5、方法: `int hasSame(String isbn,String title,double price)`

说明: 判断购物车中是否有同类商品。如果购物车中有同类, 则直接将数量加上去。

类名: TypeFactory

说明: 用于管理书类

1、属性:HashMap<Integer, String> booktypename

说明: 存储对应的书类ID和书类名称之间的关系

2、属性:String getBookTypeName(int booktype)

说明: 输入书类名, 传出书类ID

3、方法: setBookTypeName(int booktype,String name)

说明: 新增书类

4、方法: void initJB(JComboBox jb)

说明: 用于刷新选择书类的下拉框

5、方法: `void InitType()`

说明: 初始化类别

- 6、方法: **boolean** hasTypeID(**int** id)
说明: 判断是否存在这个书类ID

类名: PricingStrategyFactory

说明: 用于处理策略的增加、修改、删除

- 1、属性:PricingStrategyFactory instance= new PricingStrategyFactory();
说明: 一个PricingStrategyFactory类的实例, 用于单例模式
- 2、属性:HashMap<Integer, String> strategiename
说明: 策略的书类ID到策略名称之间的投影
- 3、属性: HashMap<Integer,IPricingStrategy> strategies
说明: 策略的书类ID到策略本身之间的投影
- 4、方法:void addStrategy(int booktype,IPricingStrategy ips,String strname)
说明: 将策略增加存储进数据结构
- 5、方法:**void** deleteStrategy(**int** booktype)
说明: 将策略从数据结构中删除
- 6、方法:void updateStrategy(int booktype,IPricingStrategy ips,String strname)
说明: 更新策略
- 7、方法:IPricingStrategy getPricingStrategy(int booktype)
说明: 通过书类别获取相应策略
- 8、方法:IPricingStrategy createSimpleStrategy(int strategytype,double num)
说明: 创建一个简单策略
- 9、方法:IPricingStrategy createCompositeStrategy(ArrayList<IPricingStrategy> strategyArray)
说明: 创建一个组合策略
- 10、方法:**boolean** hasbooktype(**int** booktype)
说明: 是否含有用于此类书的策略
- 11、方法:int getTypeTnt(String name)
说明: 通过策略名得知策略对应的书类别
- 12、方法:**static double** getTotal(**double** price,**int** type)
说明: 通过书的单价和种类, 计算出其优惠后的单价

接口：IPricingStrategy

- 1、方法: `double getSubTotal(double price);`
说明：为策略提供的接口，用于计算出优惠价格
- 2、方法: `public void add(IPricingStrategy ips);`
说明：主要为组合策略提供的接口，用于在组合策略中增加子策略

类名：NoiscountStrategy

- 1、方法: `double getSubTotal(double price);`
说明：为策略提供的接口，用于计算出优惠价格

类名：PercentaeStrategy

- 1、属性: `int discountPercentage;`
说明：折扣百分比
- 2、方法: `double getSubTotal(double price);`
说明：返回打折后单价

类名：FlatRateStrategy

- 1、属性: `double discountPerBook;`
说明：单本书便宜的价格
- 2、方法: `double getSubTotal(double price)`
说明：返回减少折扣后单价。

类名：CopositeStrategy

说明：是一种组合策略

- 1、属性: `ArrayList<IPricingStrategy> strategies`
说明：存有其子数列的数组
- 2、方法: `double getSubTotal(double price)`
说明：返回最优子策略折扣后单价
- 3、方法: `void add(IPricingStrategy ips)`
说明：在组合策略中加入子策略

类名：BookCatalogTable

说明：用于管理所有的图书表格

- 1、属性: `LinkedList<TablePair> tableList`
说明：存有所有生成的图书表格，需要更新时就遍历这张表
- 2、方法: `JTable createTable(DefaultTableModel tableModel)`
说明：用于生成图书表格，程序中所有图书表都在这生成，并被存在链表中
- 3、方法: `void addTable(double price,String title,int type,String isbn)`
说明：在所有图书表中增加一个图书信息
- 4、方法: `void deleteTable(String isbn)`
说明：删除所有图书表中号码为isbn的图书

- 5、方法: `void updateTable()`
说明: 更新所有的图书表

类名: BookCatalog

说明: 保存管理所有图书信息

- 1、属性: `LinkedList<BookSpecification> books`
说明: 存储有图书信息的链表
- 2、方法: `BookSpecification getBookSpecification(String isbn)`
说明: 通过isbn号查找到相应图书
- 3、方法: `void addBook(BookSpecification bk)`
说明: 增加图书至链表中
- 4、方法: `void initTable(DefaultTableModel dtm)`
说明: 用链表中的信息初始化图书表
- 5、方法: `void updateTable(DefaultTableModel dtm)`
说明: 更新单个图书表, 用于辅助图书表管理
- 6、方法: `boolean hasbooktype(int type)`
说明: 判断是否含有这一类的图书
- 7、方法: `boolean hasisbn(String isbn)`
说明: 判断是否含有此isbn号的图书

类名: BookSpecification

说明: 一个书本的基本信息

- 1、属性: `double price`
说明: 书本价格
- 2、方法: `String title`
说明: 书本标题
- 3、属性: `int type`
说明: 书本类别
- 4、方法: `String isbn`
说明: 书本isbn号
- 5、属性: `double getPrice()`
说明: 获取价格
- 6、方法: `String getIsbn()`
说明: 获取isbn号
- 7、属性: `String gettypename()`
说明: 获取书本类别名
- 8、方法: `String gettitle()`
说明: 获取书本标题

- 9、方法: `int gettype()`
说明: 获取书本类别

类名: TablePair

说明: 用于将表格和表格模型联系在一起, 方便管理

- 1、属性: `JTable table`
说明: 一个表格
- 2、属性: `DefaultTableModel tmodel`
说明: 一个表格模组

类名: Controller

说明: 用于传递操作的控制器

- 1、方法: `void addBook(BookSpecification bk)`
说明: 在数据结构中增加书本
- 2、方法: `void deleteBook(String isbn)`
说明: 在数据结构中删去书本
- 3、方法: `void addSimpleStrategy(int booktype, IPricingStrategy ips, String strname)`
说明: 在数据结构中增加策略
- 4、方法: `void deleteStrategy(int booktype)`
说明: 在数据结构中删除策略

下页为界面展示:

主界面：

MainUI

Menu(M)

| 策略号 | 策略名称 | 策略类型 | 适用类型 | 折扣百分比/每... |
|-----|-----------------|------------|---------|------------|
| 0 | Science Disc... | Percentage | Science | 20 |
| 1 | Social Discount | Percentage | Social | 33 |
| 2 | Math Discount | FlatRate | Math | 10 |

--BookType--

--StrType--

--StrName--

AddSimpleStrategy

CreateCopos

AddCopositeStrategy

Update

Delete

Switch

MainUI

Menu(M)

| isbn | title | type | strategy | price |
|--------|-------------|---------|-----------------|-------|
| SCI-1 | Nature | Science | Science Disc... | 70.0 |
| SCO-1 | socialism | Social | Social Discount | 66.6 |
| Math-1 | FastCompute | Math | Math Discount | 180.6 |

isbnname
type

Science

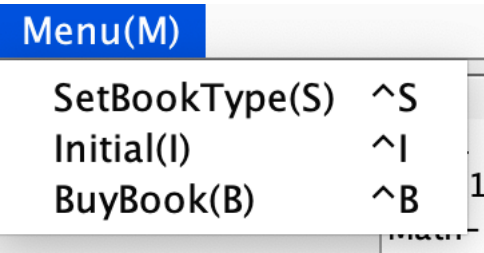
price

ADDDELETESWITCH

主界面有两个分界面，一个为策略管理界面，一个为图书管理界面，按Switch按钮进行切换。
策略管理界面功能：增加、修改、删除简单及复杂策略。用鼠标选中要删除或修改的策略。
图书管理界面功能：增加或删除书本。用鼠标选中要删除的书本。

注：界面对输入进行判断及处理，非法的输入会被拒绝，鲁棒性较高。价格栏中输入非数字会被自动屏蔽。

左上角菜单有三个功能：设置书本类别，初始化策略、书本和类别、购买书本。可使用快捷键。

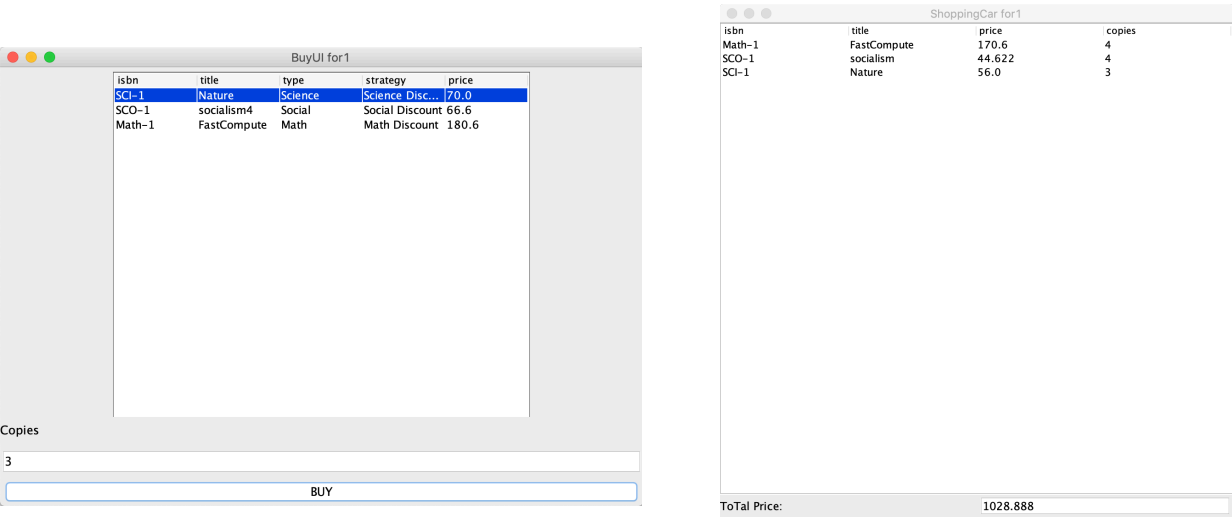


设置类别界面如下,用于增加新的书类别。



ID若重复，添加会被拒绝。ID框输入的非数字会被自动屏蔽。

买书页面及购物车如下：



打开买书页面后会自动生成一个匹配的购物车页面。你可以打开多个买书页面，可以同时运作，标题会显示ID，匹配的两个页面ID相同。用鼠标选中书类，输入书量，点击购买。购物车自动更新。

自动同步更新：当策略、书本被删除、修改或增加时，所有的买书页面会被同步更新。（如：某书策略本来是Discount 01，我将Discount 01策略删除后，购买页面中这本书的策略被刷新成default，即没有打折策略。某本书被删除后，所有购买页面中的这本书被同步删除。）