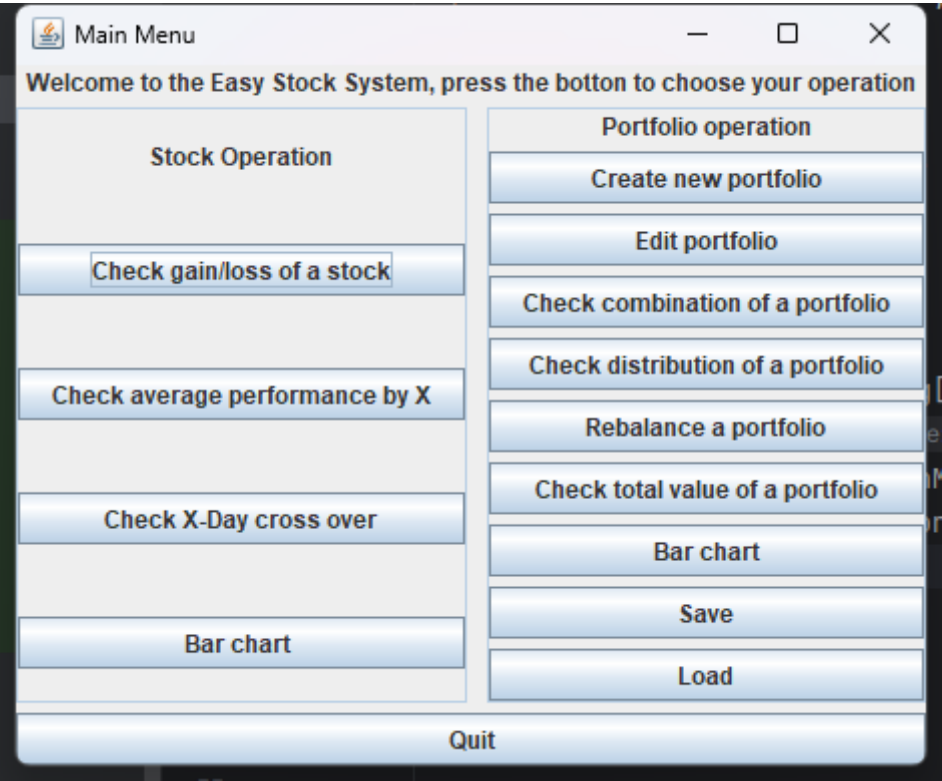


具体交互逻辑：

当程序开始运行时，应打开主菜单：



（最基本的制作已完成）

对于左侧的按键，打开后无需用户进行选择 只需要用户进行输入，参考设计：

此图对应check gain/loss of a stock。重要：基本山所有的功能都需要用户输入日期，日期分为三种：开始日期，结束日期和普通日期（代表某个时间点）。为了保证程序的简易上手，所有日期均应该分成三部分，年-月-日。

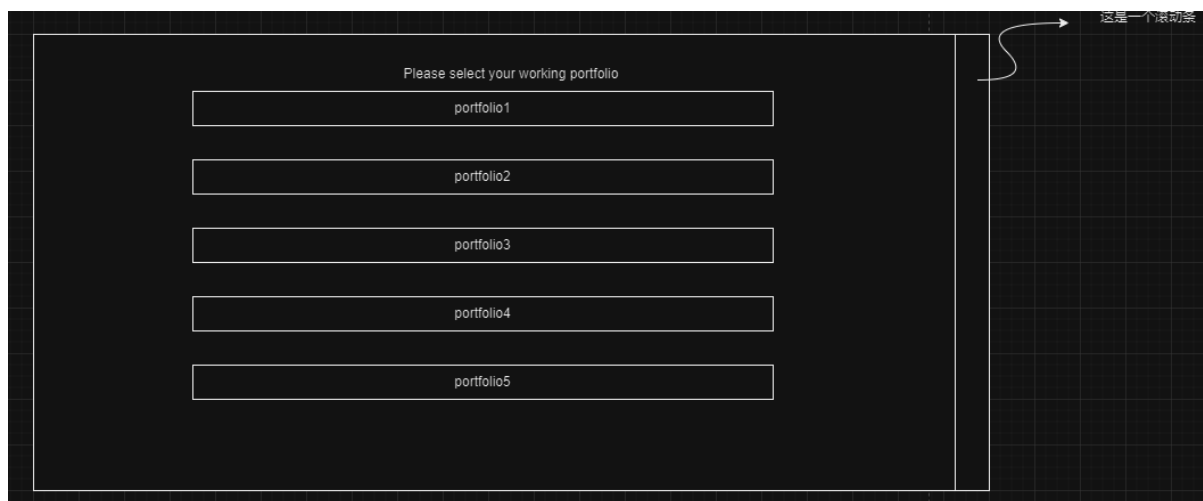
在MVC结构种，controller 需要检查最基本的用户输入。以check gain/loss功能为例，在model接口中该方法的参数为(String stockSymbol, String startDate, String endDate). 其中在controller中应该检查startDate和endDate是否符合yyyy-MM-dd的格式，可以参考目前已有的controller中私有方法checkDate

同时，“介绍功能”可以作为占位符存在，后续可以我自己更改内容，在StockInfoView中有两个String，menuTwo和portfolioSubMenu。这两个文本菜单中涵盖了每个按钮对应的功能介绍。如果您想直接完善“介绍功能”而不是放个text作为占位符的话，需要功能介绍和输入内容说明，例如gain/loss功能可以是Enter stock symbol; Enter start date; Enter end date to get the gain/loss information。

额外注意：gain/loss功能需要具体说明gain和loss，例如model方法中loss会返回一个负浮点数，controller需要处理这个数据并返回适当的内容。同时，当用户点击任何一个功能页面中的submit按钮后，无论是报错还是返回正常结果都应该有弹出的dialog框，并且当用户点击对话框中的确定或关闭后程序应该直接返回主界面

在右侧的portfolio的相关内容中：

具体的逻辑是，用户能够读取/保存他们在程序中的portfolio。可以将运行的程序参考成一个portfolio暂存器。而他们只能修改程序运行时暂存的portfolio。因此save功能可以让用户选择保存某一个portfolio（一次只能选一个），请注意 当程序中的某个portfolio名称和本地保存的目标路径中某个portfolio重复时，程序应先询问是否要覆盖(Yes/No).当用户选择Yes时再调取model中的方法进行保存。对于load功能也是类似的操作。（Save/Load）功能再StockInfoController中有私有方法，可以参考。同时每当按下某个按钮时都应该有对应的报错（如果异常被抛出），例如用户使用Save功能但是程序中并没有任何portfolio，或者使用load功能但是本地文件中并没有任何portfolio



对于这样的窗口，除了Load功能应该显示本地文件家中的portfolio（被保存成xml格式）的文件名，其他都应该显示程序中已有的portfolio的名字，如果程序中没有portfolio，那么应该返回一个对话框并显示报错（以免程序在此处陷入因无法读取程序运行中的空portfolio name list而卡住）

请注意mvc结构中关于view和controller的分开是很重要的，因此controller中应该只包含交互逻辑，具体的gui界面是从view中调用的

最后关于股票信息的获取，可以参考文字controller中的私有方法stockReader，该方法调用了名为alphaVantage的api（密钥已经获得，就在方法中），并将获取的数据保存在本地csv中。这也意味着程序需要在启动时创建好可能需要的本地文件夹（在公共方法start中）同时考虑到股票信息的时效性，当程序结束时所有的包含股票信息的csv文件都应该被删除，但不要删除portfolio文件夹中的xml文件

备注：

本程序只能使用java11默认jdk中的所有内容，包括gui设计应尽可能只是用swing组件和intelliJ自带的功能

本程序应遵守mvc结构下的思路，分离view和controller，并且用户每个行文都应该有适当的报错（合理的报错）

本程序不需要任何算法上的优化（可以loop套loop），但是应当注意减少代码重复度，例如相同的功能应该使用私有helper方法

无需给公共方法添加javadoc, 后续我会自己加