学号 E21714067 专业 软件工程 姓名 石 月

实验日期 2020/4/19 教师签字 成绩

实验报告

【实验名称】 MVC实验

【实验内容】

利用基本MVC模式实现下图所示的应用程序，程序主要功能如下：

当用户在图形化用户界面上拖动表示圆半径的标尺或者直接填写相应的半径值时，自动显示圆的半径、直径、周长和对应球的表面积；并在图形界面上画出圆。

代码主要逻辑如下：

CircleModel表示MVC中的M（Model），在本程序中是表示圆的模型，具有半径和观察者列表两个成员变量，有对应的get和set方法，此外还有计算直径、周长、圆面积、球面积、球体积的方法，以及登记、移除观察者，通知观察者的方法，当模型中的数据改变时，由控制器调用，通知登记过的各个观察者改变各自的视图。

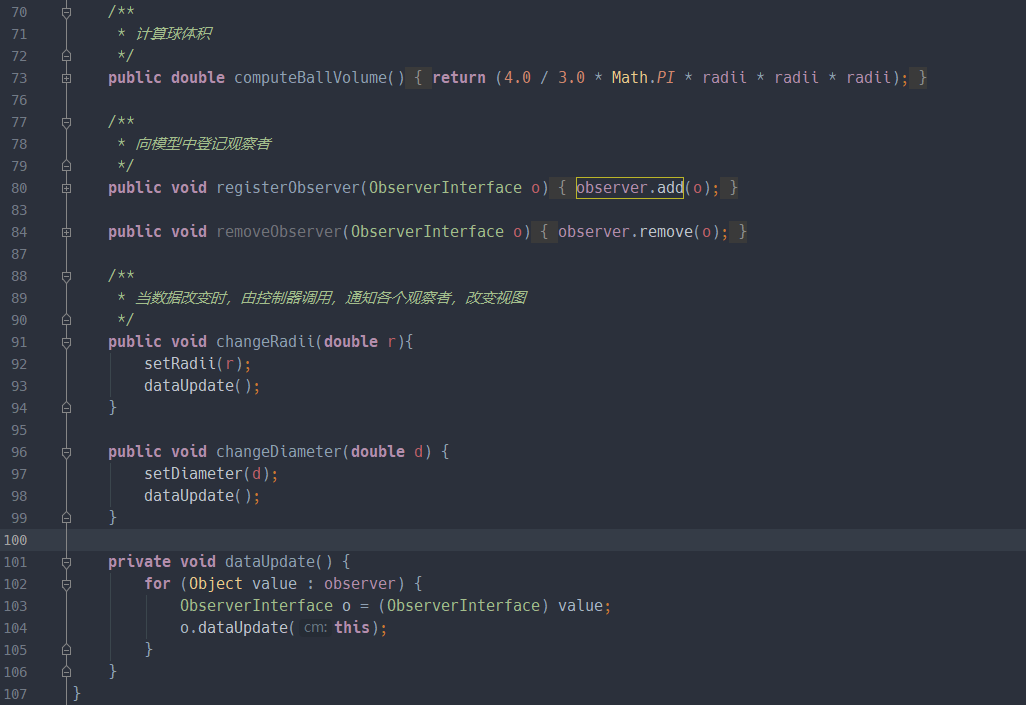
Controller表示MVC中的C（Controller），也就是控制器，拥有模型的实例和主界面的实例，负责调用模型中的登记观察者方法向模型（主题）中登记主界面中的各个观察者，还具有执行操作、改变状态等方法，当半径的数值或滑动条改变时，来控制模型。

View包中有许多个视图，即MVC中的V（View），都实现了util包中的ObserverInterface接口，因此这些视图类都是模型的观察者，当模型的数据改变时，它们都会收到通知，然后执行各自update()方法来改变自己的视图，该方法就会调用模型的对应方法进行计算数据，然后用以更新视图。

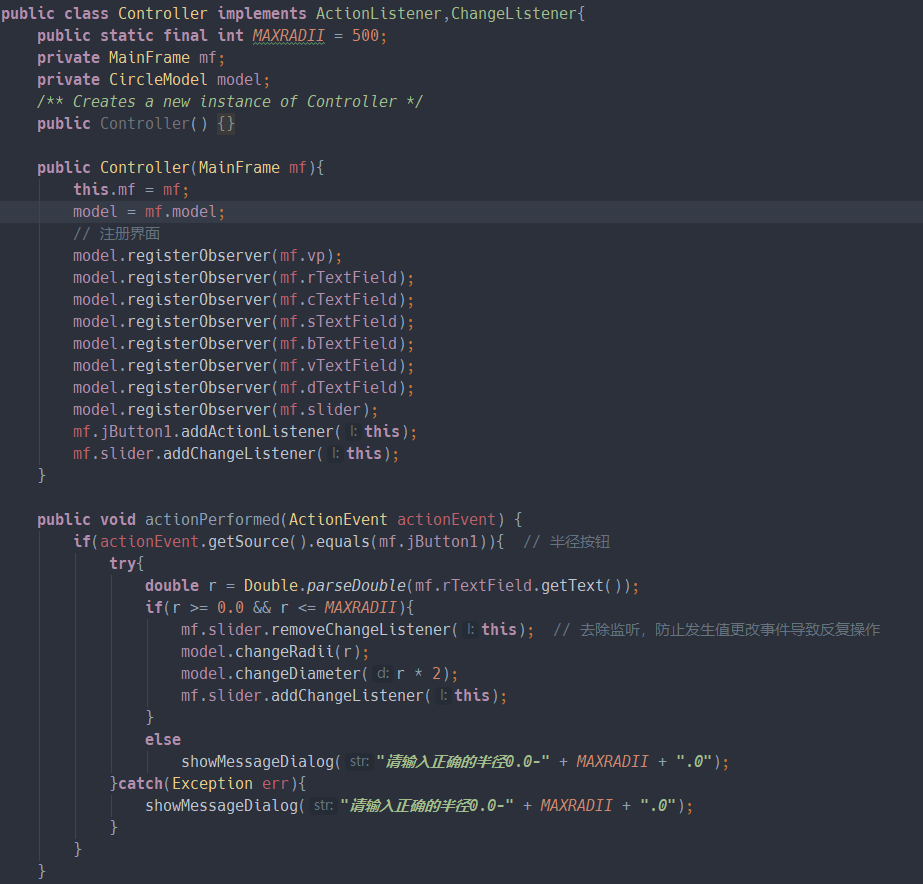
MainFrame是主界面，用于展示一个Swing的界面，相当于是提供一个显示的容器。

【实验代码】

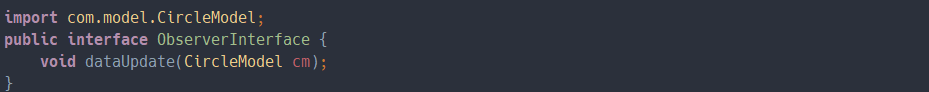
CircleModel.java



Controller.java



Observer.java



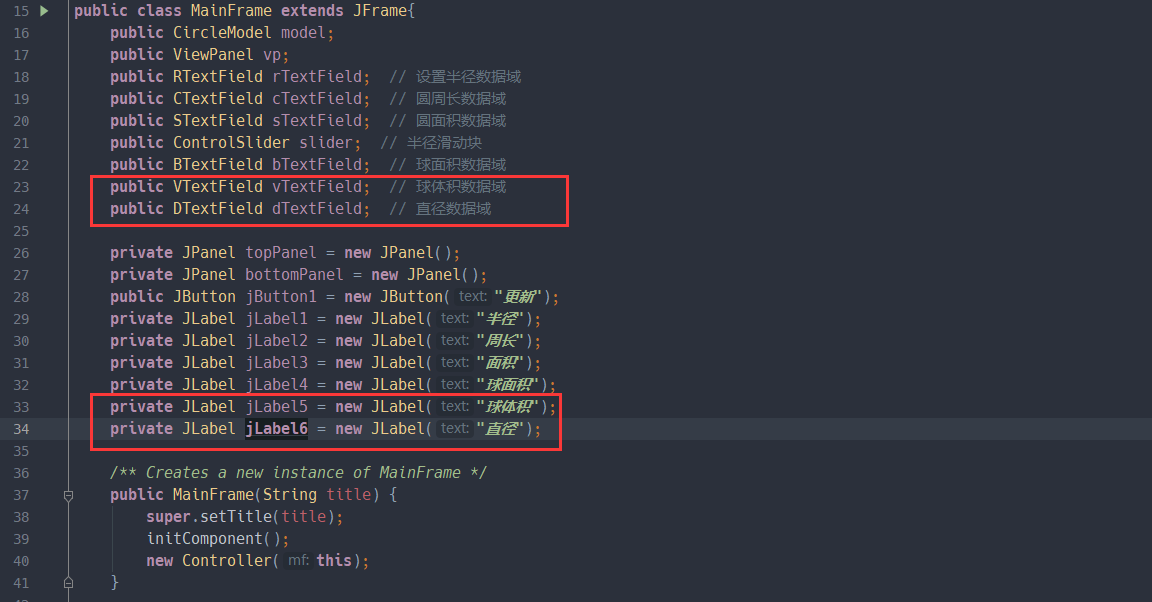
增加的视图：球体积VTextField.java

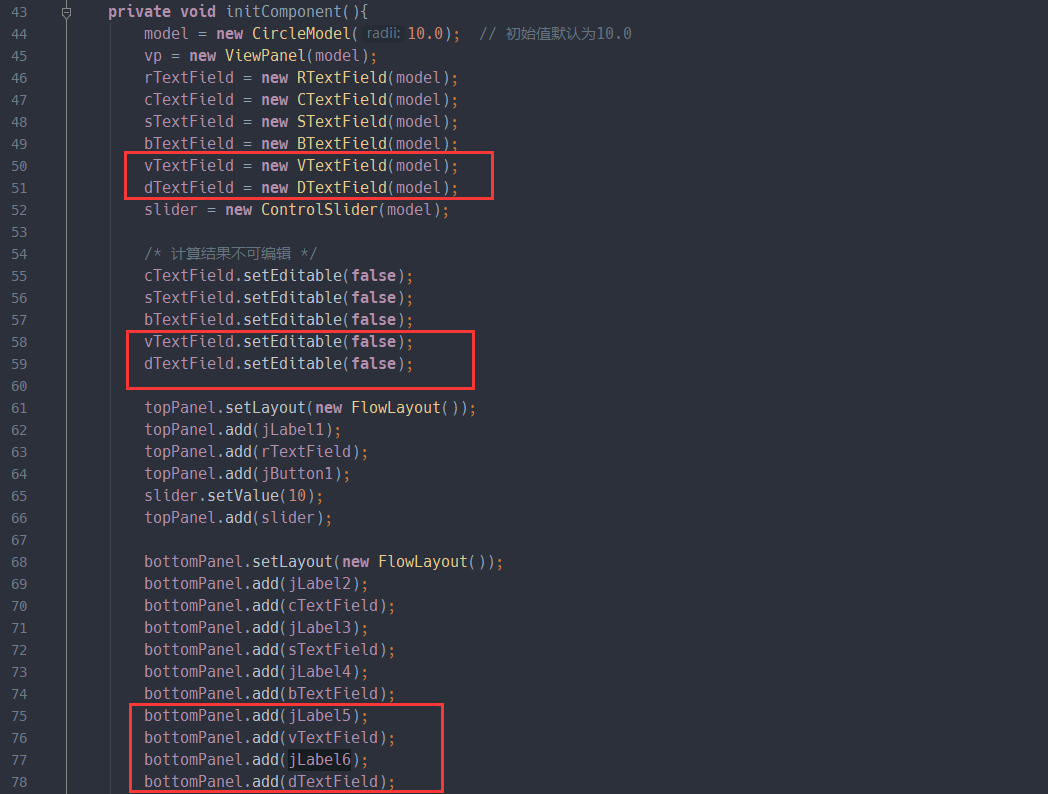


直径TextField.java

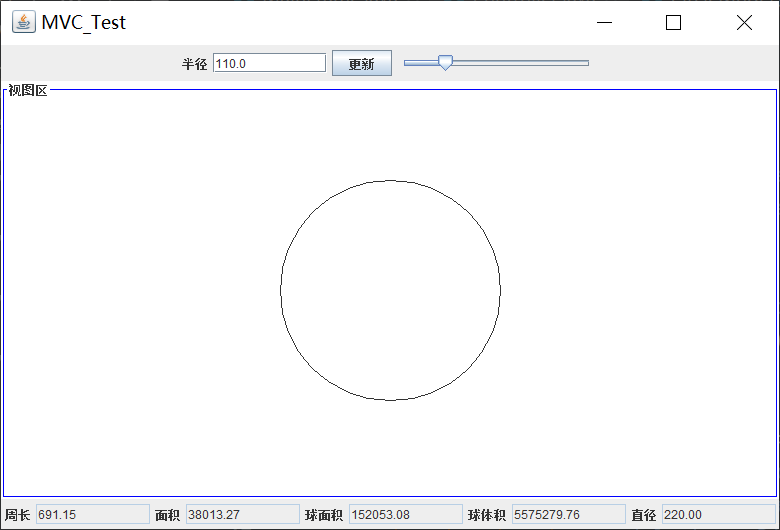


MainFrame.java中增加的部分



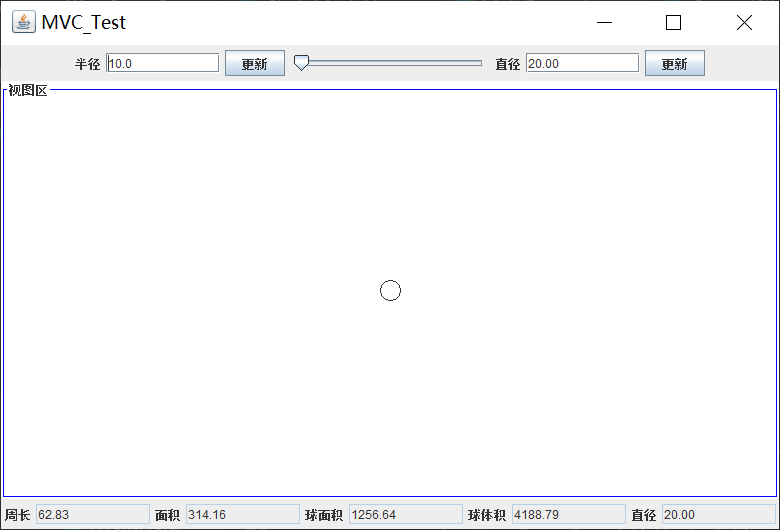


【实验结果】



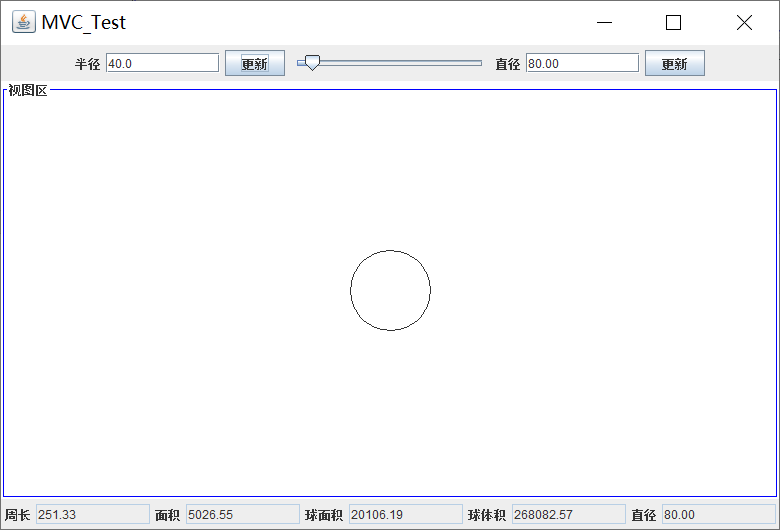
对程序加以改进，使得能够在界面上通过改变半径或直径的方式来调整圆的大小，即在程序面板上方增加一个输入直径的textField，能够通过该处的直径值来控制圆。

启动程序：

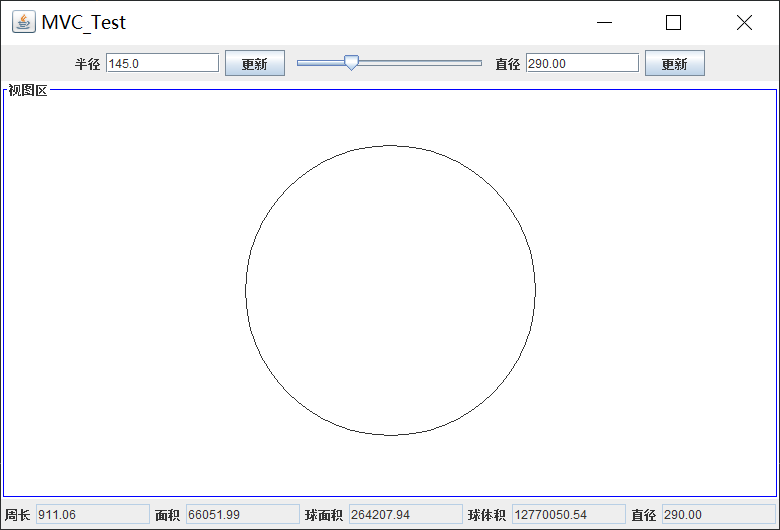


修改圆的大小的方式有3种：

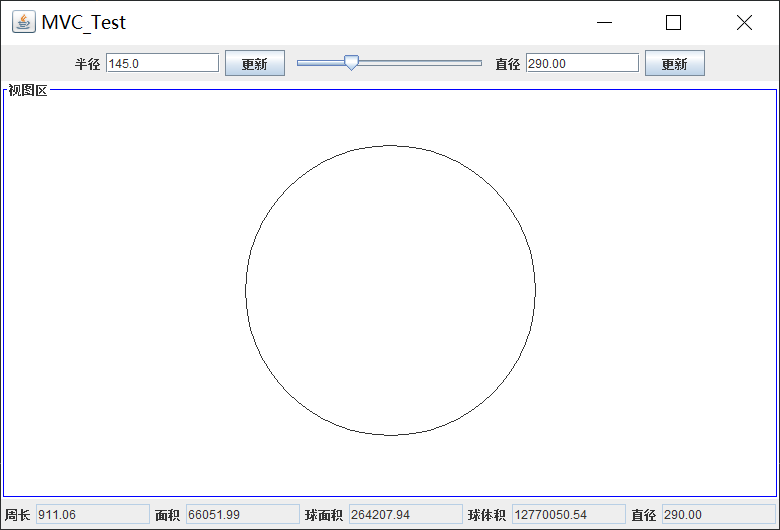
1. 输入半径修改圆的大小，点击半径旁边的“更新按钮”，直径和滑动条也随之更新：



1. 拖动滑动条修改圆的大小，两旁的半径与直径实时更新：



1. 输入直径修改圆的大小，点击更新，左边的半径和滑动条也随之更新：



通过这三种方式修改圆的大小，程序面板下方的五个相关数据均会对应改变。

【小结与体会】

通过本次实验了解了MVC架构，动手实践了MVC应用。MVC分为M模型，V视图，C控制器，实现了业务逻辑、数据、界面显示分离的方法来组织代码。MVC架构的重要组成部分就是观察者模式，也是JDK中使用最多的模式之一，比如在Swing中就大量使用了观察者模式。在观察者模式中，主题对象管理一些数据，当主题内的数据改变，就会通知观察者，而观察者已经注册了主题，以便在主题数据改变时能够收到更新。