·计算机科学与技术专业

**课程项目设计报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **课程名称：** | **Java程序设计** | | | | | |
| **年 级：** | **2018级** | | | | | |
| **学 号：** | **1811030118** | | | | | |
| **学生姓名：** | **彭佳林** | | | | | |
| **指导教师：** | **方 静** | | | | | |
| **完成时间：** | **2020** | **年** | **04** | **月** | **15** | **日** |

|  |
| --- |
| **一、程序设计目标与要求** |
| 编写一个矩形Rectangle类，这个类拥有：   1. 成员变量  * len(私有，浮点型)//存放矩形的长 * width(私有，浮点型)//存放矩形的宽  1. 构造方法  * 无参构造方法Rectangle()//将长和宽初始化为0 * 有参构造方法Rectangle(len, width)//创建Rectangle对象时将长和宽进行初始化  1. 编写长和宽的set和get方法，能够设置并获取长和宽的值。  * getLen()//获取矩形的长 * setLen()//设置矩形的长 * getWidth()//获取矩形的宽 * setWidth()/设置矩形的宽  1. 其他成员方法  * double getArea()//获取矩形的面积 * double getPerimeter()//获取矩形的周长 * void show()//将矩形的周长和面积输出到屏幕上   编写应用程序，创建Rectangle类的对象，分别显示矩形的长、宽、面积、周长，然后再分别修改矩形的长和宽，计算并分别显示矩形的长、宽、面积、周长。 |
| **二、程序设计内容** |
| import java.util.Scanner;  class Rectangle  {  private double len;  private double width;  Rectangle()  {  len=0;  width=0;  }  Rectangle(double len,double width)  {  this.len=len;  this.width=width;  }  void getLen()  {  System.out.print("矩形长为："+len+"\n");  }  void setLen(){  System.out.print("请输入矩形的长：");  Scanner lenth=new Scanner(System.in);  double i=lenth.nextDouble();  len=i;  }  void getWidth()  {  System.out.print("矩形宽为："+width+"\n");  }  void setWidth(){  System.out.print("请输入矩形的宽：");  Scanner width1=new Scanner(System.in);  double j=width1.nextDouble();  width=j;  }  double getArea()  {  double s=len\*width;  return s;  }  double getPerimeter()  {  double c=2\*(len+width);  return c;  }  void show()  {  double s=getArea();  double c=getPerimeter();  System.out.print("矩形的面积为："+s+"\n");  System.out.print("矩形的周长为："+c+"\n");  }  }  public class peng  {  public static void main(String[] args)  {  Rectangle p1=new Rectangle(10,20);  p1.getLen();  p1.getWidth();  p1.show();  p1.setLen();  p1.setWidth();  p1.getLen();  p1.getWidth();  p1.show();  }  } |
| **三、程序实验结果** |
|  |
| **四、程序设计总结** |
| 此次实验包含了private、this、类的构造方法的使用，将类中的成员变量私有化，通过构造方法传递调用，很好的体现出了java语言的封装性。每次实验不是为了完成任务，而是为了在自己学习了某些知识点之后，加以运用，从而使自己掌握该知识，此次实验原理及实现代码较为简单，主要是为了掌握java的封装性质。 |