# java程序设计编程参考

\* 本文档只提供代码片段 运行结果截图需要自己做

作业一

1.准备以下任何一个Java程序的集成开发环境，并开发一个简单的Java Application程序，在屏幕上输出: Hello Java World！提交程序代码和运行结果截图，并进行必要的说明

代码示例:

|  |
| --- |
| package com.joker.donghua;  public class Demo {  public static void main(String[] args) {  System.out.println("Hello Java World！");  }  } |

1. 编写一个Java Applet程序，使之能在浏览器中显示： Hello Java Applet World！提交Java程序代码、嵌入小程序的HTML文档和运行结果截图

HelloWorldApplet.java

|  |
| --- |
| package donghua;  import java.applet.\*;  import java.awt.\*;  public class HelloWorldApplet extends Applet{  @Override  public void paint(Graphics g){  g.drawString("Hello Java Applet World！", 5, 30);//绘制文本  }  } |

HelloWorldApplet.html

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html xmlns="http://www.w3.org/1999/xhtml">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  <title>applet</title>  </head>  <body>  <applet code="donghua.HelloWorldApplet.class" width="100" height="100" codebase=".."></applet>  </body>  </html> |

1. 编写Java Application程序，定义一个矩形类RECT: 类的属性成员包括矩形的宽width和高 height，成员方法包括带参数和不带参数的构造函数、两个属性的get方法和set方法、计算面积的方法 Area（）。在main（）定义RECT的对象，使用它的一些方法进行一些操作。提交程序代码和程序运行结果截图

代码示例:

RECT .java

|  |
| --- |
| package com.joker.donghua;  public class RECT {  /\*\*  \* 矩形-宽  \*/  private float width;  /\*\*  \* 矩形-高  \*/  private float height;  public float getWidth() {  return width;  }  public void setWidth(float width) {  this.width = width;  }  public float getHeight() {  return height;  }  public void setHeight(float height) {  this.height = height;  }  /\*\*  \* 无参构造  \*/  public RECT() {  }  /\*\*  \* 有参构造  \*  \* @param width  \* @param height  \*/  public RECT(float width, float height) {  this.width = width;  this.height = height;  }  /\*\*  \* 面积计算方法  \*  \* @return  \*/  private float Area() {  return width \* height;  }  public static void main(String[] args) {  // 使用无参构造创建对象 并 计算面积  RECT rect = new RECT();  rect.setHeight(1);  rect.setWidth(2);  System.out.println("使用无参构造创建对象 并 计算面积:" + rect.Area());  // 使用有参构造创建对象 并 计算面积  RECT rect2 = new RECT(2, 3);  System.out.println("使用有参构造创建对象 并 计算面积:" + rect2.Area());  }  } |

1. 使用Scanner类的对象，从键盘输入两个接收两个数，然后计算这两个数的乘积。提交程序代码和程序运行结果截图

ScannerTest.java

|  |
| --- |
| package com.joker.donghua;  import java.util.Scanner;  public class ScannerTest {  public static void main(String[] args) {  Scanner scan = new Scanner(System.in);  // 从键盘接收数据  // next方式接收字符串  System.out.println("请输入第一个数字：");  // 判断是否还有输入  float firstNum = scan.nextFloat();  System.out.println("请输入第一个数字：");  float secondNum = scan.nextFloat();  System.out.println("请输入第一个数字：" + (firstNum \* secondNum));  scan.close();  }  } |

1. 编写一个GUI程序，界面上包含一个按钮，点击按钮后在界面上画一个矩形。提交程序代码和程序运行结果截图

|  |
| --- |
| package com.joker.donghua;  import javax.swing.\*;  import java.awt.\*;  public class GuiTest {  public static void main(String[] args) {  // 1. 创建一个顶层容器（窗口）  JFrame jf = new JFrame("GuiTest");  // 设置窗口大小  jf.setSize(250, 250);  // 把窗口位置设置到屏幕中心  jf.setLocationRelativeTo(null);  // 当点击窗口的关闭按钮时退出程序  jf.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);  // 2. 创建中间容器（面板容器） 使用默认的布局管理器  JPanel panel = new JPanel();  // 3. 创建一个基本组件（按钮），并添加到 面板容器 中  JButton btn = new JButton("GuiTestButton");  btn.addActionListener(e -> {  Graphics graphics = panel.getGraphics();  Graphics2D g2d = (Graphics2D) graphics.create();  g2d.setRenderingHint(RenderingHints.KEY\_ANTIALIASING, RenderingHints.VALUE\_ANTIALIAS\_ON);  g2d.setColor(Color.GRAY);  // 绘制一个矩形: 起点(30, 20), 宽80, 高100  g2d.drawRect(85, 50, 80, 100);  });  panel.add(btn);  // 4. 把 面板容器 作为窗口的内容面板 设置到 窗口  jf.setContentPane(panel);  // 5. 显示窗口，前面创建的信息都在内存中，通过 jf.setVisible(true) 把内存中的窗口显示在屏幕上。  jf.setVisible(true);  }  } |

作业二

* 1. 编写一个Java程序，能从键盘上接收两个整数，然后计算两个数的乘积 。

代码片段:

|  |
| --- |
| package com.joker.donghua;  import java.util.InputMismatchException;  import java.util.Scanner;  public class Product {  public static void main(String[] args) {  Scanner cin = new Scanner(System.in);  try {  System.out.println("请输入第一个数字");  float firstNum = cin.nextFloat();  System.out.println("请输入第二个数字");  float secondNum = cin.nextFloat();  System.out.println("两个数字的乘积是:" + (firstNum \* secondNum));  } catch (InputMismatchException e) {  System.out.println("输入数据格式错，要求是数字！");  }  }  } |

2.编写求一个整数数组A[10,15,12,9,7]中最大元素max和元素之和sum的程序。

代码片段:

|  |
| --- |
| package com.joker.donghua;  public class MaxAndSum {  public static void main(String[] args) {  int[] numArr = {10, 15, 12, 9, 7};  int sum = 0;  int max = 0;  for (int item : numArr) {  sum = sum + item;  if (item > max) {  max = item;  }  }  System.out.println("数组之和sum=" + sum);  System.out.println("最大元素max=" + max);  }  } |

2.首先定义一个雇员类EMP，类的属性包括雇员编号ID，雇员名Name，年龄Age，性别Sex和专业Major，成员方法包括构造方法、属性的get方法和set方法，打印雇员信息的方法PrintEMP，然后在主类的main()函数中为5号计算机网络专业的23岁男性雇员王强定义一个相应的对象emp，并调用对象的PrintEMP ()方法在屏幕上显示其信息。

代码片段:

EmpTest.java

|  |
| --- |
| package com.joker.donghua;  public class EmpTest {  public static void main(String[] args) {  EMP emp = new EMP();  emp.setID(5);  emp.setName("王强");  emp.setAge(23);  emp.setSex("男");  emp.setMajor("计算机网络专业");  emp.PrintEMP();  }  } |

EMP.java

|  |
| --- |
| package com.joker.donghua;  public class EMP {  public EMP() {  }  public EMP(int ID, String name, int age, String sex, String major) {  this.ID = ID;  Name = name;  Age = age;  Sex = sex;  Major = major;  }  public void PrintEMP() {  System.out.println("雇员id:" + this.ID);  System.out.println("雇员名称:" + this.Name);  System.out.println("雇员年龄:" + this.Age);  System.out.println("雇员性别:" + this.Sex);  System.out.println("雇员专业:" + this.Major);  }  private int ID;  private String Name;  private int Age;  private String Sex;  private String Major;  public int getID() {  return ID;  }  public void setID(int ID) {  this.ID = ID;  }  public String getName() {  return Name;  }  public void setName(String name) {  Name = name;  }  public int getAge() {  return Age;  }  public void setAge(int age) {  Age = age;  }  public String getSex() {  return Sex;  }  public void setSex(String sex) {  Sex = sex;  }  public String getMajor() {  return Major;  }  public void setMajor(String major) {  Major = major;  }  } |

2.编写一个Applet程序DrawRec，以坐标点 (10,10)为左上角绘制宽为30，高为20的红颜色线矩形，然后结合本例说明开发一个Applet程序的过程。

代码片段:

DrawRec.java

|  |
| --- |
| package donghua;  import java.applet.Applet;  import java.awt.\*;  public class DrawRec extends Applet {  @Override  public void paint(Graphics g) {  g.setColor(Color.RED);  g.drawRect(10, 10, 30, 20);  }  } |

DrawRec.html

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html xmlns="http://www.w3.org/1999/xhtml">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  <title>apple</title>  </head>  <body>  <applet code="donghua.DrawRec.class" width="100" height="100" codebase=".."></applet>  </body>  </html> |

2.定义两个线程类ThreadA和ThreadB，其行为是分别显示“I am Thread A”和“I am Thread B”， 然后在main()函数中创建两个线程的对象，并分别启动两个线程。

代码片段:

ThreadA.java

|  |
| --- |
| package com.joker.donghua;  public class ThreadA extends Thread {  @Override  public void run() {  super.run();  System.out.println("I am Thread A");  }  } |

ThreadB.java

|  |
| --- |
| package com.joker.donghua;  public class ThreadB extends Thread {  @Override  public void run() {  super.run();  System.out.println("I am Thread B");  }  } |

ThreadTest.java

|  |
| --- |
| package com.joker.donghua;  public class ThreadTest {  public static void main(String[] args) {  ThreadA threadA = new ThreadA();  ThreadB threadB = new ThreadB();  threadA.start();  threadB.start();  }  } |