**Java实训**

|  |  |  |
| --- | --- | --- |
| 课程名称：java第五次作业 | 班级：1806班 | 实训日期： |
| 姓名：蒋云濛 | 学号：20181855 | 指导老师：杨帆 |
| 1. 键盘录入一个字符，统计字符串中大小写字母及数字字符个数   import java.util.Scanner;  public class test{  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  int upCaseCount = 0;  int lowerCaseCount = 0;  int numberCount = 0;  System.out.print("键盘输入一个字符串：");  String str = scanner.next();  for (int i = 0;i < str.length();i++){  if (str.charAt(i)>='A' && str.charAt(i)<='Z'){  upCaseCount++;  }else if (str.charAt(i)>='a' && str.charAt(i)<='z'){  lowerCaseCount++;  }else if (str.charAt(i)>='0' && str.charAt(i)<='9'){  numberCount++;  }  }  System.out.println("大写字母个数："+upCaseCount);  System.out.println("小写字母个数："+lowerCaseCount);  System.out.println("数字个数："+numberCount);  }   1. 定义一个方法，把数组{1,2,3}按照指定个格式拼接成一个字符串   import java.util.Scanner;  public class test{  public static void main(String[] args) {  int[] arr = {1,2,3};  StringBuffer stringBuffer = new StringBuffer();  stringBuffer.append(arr[2]);  stringBuffer.append(arr[0]);  stringBuffer.append(arr[1]);  System.out.println(stringBuffer.toString());//312  }  }   1. 计算在 -10.8 到 5.9 之间，绝对值大于 6 或者小于 2.1 的整数有多少个？   public class test{  public static void main(String[] args) {  int count=0;  double min=-10.8;  double max=5.9;  for(int i=(int)min;i<max;i++) {  int abs = Math.abs(i);  if(abs>6 || abs<2.1){  System.out.println(i);  count++;  }  }  System.out.println("整数的个数为："+count);  }  }   1. String,stringBuffer,StringbBuilder三者之间的区别   String对象是一经创建就不可修改的。  StringBuffer对象new出来底层创建一个初始化容量为16的byte[]数组，通过调用append()方法进行字符串的拼接，StringBuffer是多线程安全的。  StringBuilder相比于StringBuffer是非线程安全的。   1. 验证for循环打印数字1-9999所需要使用的时间   public class test{  public static void main(String[] args) {  long time1 = System.currentTimeMillis();  for (int i = 1; i < 10000; i++) {  System.out.println(i);  }  long time2 = System.currentTimeMillis();  System.out.println("该打印方法执行完成所需时间为："+(time2-time1));  }  }   1. 计算出一个人已经出生了多少天   public class test{  Scanner scanner = new Scanner(System.in);  System.out.print("输入你的出生日期，格式为“yyyy-MM-dd”：");  String birthTime = scanner.next();  System.out.println("格式为yyyy-MM-dd的出生日期："+birthTime);  SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");  Date birthTime2 = sdf.parse(birthTime);  System.out.println("格式为系统默认的格式为："+birthTime2);  long birthTimeMillis = birthTime2.getTime();  System.out.println("1970-01-01 00:00:00 000到你出生的毫秒数："+birthTimeMillis);  Date nowTime = new Date();  long nowTimeMillis = nowTime.getTime();  System.out.println("1970-01-01 00:00:00 000到现在的毫秒数："+nowTimeMillis);  long birthMillis = nowTimeMillis-birthTimeMillis;  System.out.println("你出生到现在总共"+(birthMillis/1000/60/60/24)+"天啦！");  }   1. 完成斗地主的发牌 import java.util.ArrayList; import java.util.Collections; import java.util.List; public class Test{  //构造函数初始化  public Test() {  init1();  System.out.println(poxBox);//生成牌  Collections.shuffle(poxBox);  System.out.println(poxBox);//洗牌  init2();  }  //生成牌  //扑克牌共54张，4种花色，9个数字（2-10）+4个字母（A，J，Q，K）+大小王各1张  ArrayList<String> poxBox = new ArrayList<>();  public void init1(){  ArrayList<String> colors = new ArrayList<>();  colors.add("♦");  colors.add("♥");  colors.add("♣");  colors.add("♠");  ArrayList<String> numbers = new ArrayList<>();  for (int i = 2; i < 11; i++) {  numbers.add(i+"");  }  numbers.add("A");  numbers.add("J");  numbers.add("Q");  numbers.add("K");  poxBox.add("☠");  poxBox.add("☺");  for (String number:numbers){  for (String color:colors){  poxBox.add(color+number);  }  }  }  //发牌  public void init2(){  ArrayList<String> player1 = new ArrayList<>();  ArrayList<String> player2 = new ArrayList<>();  ArrayList<String> player3 = new ArrayList<>();  ArrayList<String> bottomCard = new ArrayList<>();  for (int i = 0; i < poxBox.size() ; i++) {  String card = poxBox.get(i);  if (i>50){  bottomCard.add(card);  }else if (i%3==0){  player1.add(card);  }else if (i%3==1){  player2.add(card);  }else if (i%3==2){  player3.add(card);  }  }  System.out.println(bottomCard);  System.out.println(player1);  System.out.println(player2);  System.out.println(player3);  }   public static void main(String[] args) {  new Test();  } } | | |