

面向对象Java—实验报告

****

**实验四： Java API**.

**姓 名：**   **於俊涛**

**学 号： 202007020625**

**专业班级： 计算机203**

**系 别： 计算机系**

**学 院： 电子信息与人工智能学院**

**实验四 Java API**

1. **实验目的**

在集成开发环境下，通过阅读Java API文档来使用合适的常用类及方法解决实际问题。

1. **实验目标**

1.能够通过阅读Java API文档来灵活运用Java API中的一些常用类（例如String、StringBuffer、System、Runtime、Math、Random等）来解决实际问题。

2.能够灵活运用用Java语言的常用集合类（ArrayList、Map、Collections、Array等）来解决实际问题。

1. **实验内容**

**3.1 实验环境**

Eclipse IDE for Java Developers

**3.2 具体实验内容**

1.定义一个方法，根据用户身份证号码计算其年龄并输出。

2.采用Random类或者Math类随机生成一组随机数，存入List集合类(ArrayList、LinkedList等都可以)，并利用Iterator接口便利集合。

3.分别统计字符串中出现的大写字母、小写字母个数，以及出现的数字字符以及其他字符的个数。

4. 编写一个程序,实现字符串大小写的转换并倒序输出。具体要求：

（1）使用for循环将字符串从最后一个字符开始遍历。

（2）遍历的当前字符如果是大写字符，就使用toLowerCase()方法将其转换为小写字符，反之则使用toUpperCase()方法将其转换为大写字符。

（3）定义一个StringBuffer对象，调用append()方法依次添加遍历的字符，最后调用StringBuffer对象的toString( )方法，并将得到的结果输出。

注意：倒叙序出要单独写一个有参的方法，在主函数里调用该方法完成测试。

1. **实验过程分析**

**4.1 实验步骤**

1.定义一个方法，根据用户身份证号码计算其年龄并输出。

实验代码：

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.Scanner;

class parse{

public parse(String identifys){

date[0]= Integer.parseInt(identifys.substring(6,10));

date[1]=Integer.parseInt(identifys.substring(10,12));

date[2]=Integer.parseInt(identifys.substring(12,14));

get\_now\_date();}

private void get\_now\_date(){

Date date=new Date();

SimpleDateFormat dateFormat= new SimpleDateFormat("yyyy-MM-dd");

String Date=dateFormat.format(date);

System.out.println("当前日期为："+Date);

now[0]= Integer.parseInt(Date.substring(0,4));

now[1]=Integer.parseInt(Date.substring(5,7));

now[2]=Integer.parseInt(Date.substring(8,10));}

private boolean check(){ return now[0]>=date[0]&& date[0]>1900 &&(date[1]>0&&date[1]<13);}

public int Figure(){

if (check()){

int age= now[0]-date[0]-1;

if(now[1]<date[1]){age--; }

else if(now[1]==date[1]){

if(now[2]<date[2]){

age--;}

else if(now[2]>date[2]){age++;}}

if(now[1]>date[1]){age++;}

return age;}

return -1;}

private int date[]=new int[3];

private int now[]=new int[3];}

public class identify {

public static void main(String[] args) {

Scanner cin=new Scanner(System.in);

String identifys=cin.nextLine();

if (identifys.length()!=18){

System.out.println("身份证号输入错误！");

return;}

parse Person=new parse(identifys);

if (Person.Figure()>0){

System.out.println("年龄为："+Person.Figure());

return;}

else {

System.out.println("身份证号输入错误！");

return;}}}

结果输出：

610424200110101304

当前日期为：2022-6-20

年龄为：21

2.采用Random类或者Math类随机生成一组随机数，存入List集合类(ArrayList、LinkedList等都可以)，并利用Iterator接口便利集合。

实验代码：

import java.util.ArrayList;

import java.util.Iterator;

import java.util.Random;

public class Randtest {

public static void main(String[] args) {

Iterator<Integer> integers = generateRandomNumber().iterator();

while (integers.hasNext())

System.out.println(integers.next());}

public static ArrayList<Integer> generateRandomNumber () {

Random random = new Random();

ArrayList<Integer> arrayList = new ArrayList<>();

for (int i = 0; i < 10; i++)

arrayList.add(random.nextInt(1000));

return arrayList;}}

结果输出：

237

941

720

976

631

308

148

97

727

289

3.分别统计字符串中出现的大写字母、小写字母个数，以及出现的数字字符以及其他字符的个数。

实验代码：

import java.util.Scanner;

public class Six {

public static void main(String[] args) {

// TODO Auto-generated method stub

int lower=0,upper=0,number=0,space=0,other=0;

System.out.println("请输入字符串:");

String str;

Scanner in=new Scanner(System.in);

str=in.nextLine();

for(int i=0;i<str.length();i++)

{

if(Character.isLowerCase(str.charAt(i))) {lower++;}

else if(Character.isUpperCase(str.charAt(i))) {upper++;}

else if(Character.isDigit(str.charAt(i))) {number++;}

else if(Character.isSpace(str.charAt(i))) {space++;}

else other++;

}

System.out.println("小写字母个数："+lower);

System.out.println("大写字母个数："+upper);

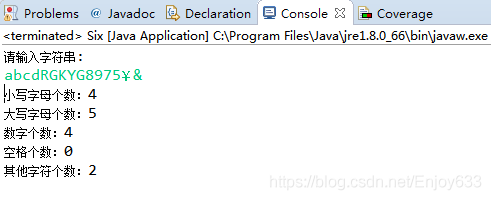
System.out.println("数字个数："+number);

System.out.println("空格个数："+space);

System.out.println("其他字符个数："+other);

}

}

结果输出：

4. 编写一个程序,实现字符串大小写的转换并倒序输出。具体要求：

（1）使用for循环将字符串从最后一个字符开始遍历。

（2）遍历的当前字符如果是大写字符，就使用toLowerCase()方法将其转换为小写字符，反之则使用toUpperCase()方法将其转换为大写字符。

（3）定义一个StringBuffer对象，调用append()方法依次添加遍历的字符，最后调用StringBuffer对象的toString( )方法，并将得到的结果输出。

注意：倒叙序出要单独写一个有参的方法，在主函数里调用该方法完成测试。

实验代码：

import java.util.Locale;

public class reverse {

public static void main(String[] args) {

String origin = "ITcastHeiMa";

StringBuffer stringBuffer = new StringBuffer("");

for (int i = origin.length() - 1, j = 0; i >= 0; i--, j++) {

char c = origin.charAt(i);

if (c <= 'z' && c >= 'a') {

stringBuffer.append((c + "").toUpperCase(Locale.ROOT));

}

if (c <= 'Z' && c >= 'A') {

stringBuffer.append((c + "").toLowerCase(Locale.ROOT));

}

}

System.out.println(stringBuffer.toString());

}

}

结果输出：

AmIEhTSACti

**4.2 错误分析**

1.在输入身份证号之后返回的年龄为-1，

分析原因：

问题产生源于未通过check函数的检查，最终发现now时间未被初始化get\_now\_date初始化函数未被调用。

解决方案：

在初始化date的同时调用get\_now\_date初始化函数初始化now.问题解决！

1. **实验总结**

在本次实验中通过阅读Java API文档学习Java API中的一些常用类（例如String、StringBuffer、System、Runtime、Math、Random等）来解决实际问题。并尝试使用Java语言的常用集合类（ArrayList、Map、Collections、Array等）来解决实际问题。