# Arrays

1. asList

将一个数组(**变长参数的语法糖实现就是数组**)转变成一个List(确切的来说是ArrayList)，注意这个List是定长的，企图添加或者删除数据都会报错

public static <T> List<T> asList(T... a) { return new ArrayList<>(a); }

1. sort

对数组进行排序。适合byte,char,double,float,int,long,short等基本类型，还有Object类型（**实现了Comparable接口**），如果提供了比较器Comparator也可以适用于泛型。

int a[] = new int[]{1,9,5,4,6,4,7,1}; Arrays.sort(a);

1. toString

数组转化为字符串

String str[] = {"s2","s4","s1","s3"}; Arrays.sort(str); System.out.println(Arrays.toString(str));

1. binarySearch

binarySearch通过**二分查找法**对已排序（譬如经过Arrays.sort排序，且按照升序进行排序。如果**数组没有经过排序，那么检索结果未知**）的数组进行查找。适合byte,char,double,float,int,long,short等基本类型，还有Object类型

1. copyOf和copyOfRange

数组拷贝，底层采用System.arrayCopy（native方法）实现。

String str[] = {"s2","s4","s1","s3"}; String str2[] = Arrays.copyOf(str, str.length);

String str2[] = Arrays.copyOfRange(str,1,3);

1. equals和deepEquals

equals：判断两个数组的每一个对应的元素是否相等, deepEquals适用于数组

String str1[] = {"s2","s4","s1","s3",null}; String str2[] = Arrays.copyOf(str1, str1.length); System.out.println(Arrays.equals(str1, str2));

1. fill

填充值

String str[] = {"s2","s4","s1","s3",null}; Arrays.fill(str, "s5");

1. toString和deepToString

toString数组转字符串,deepToString递归实现。

1. hashCode和deepHashCode

计算数组的hashcode

数组与其他容器区别：

效率（随机访问最快）、类型（容器会丢掉类型）和保存基本类的能力（容器只能存object，基础类型自动转型）

Char a [] = {‘1’};

a.length

数组与泛型

Peel<Banna> peels;peels = new peel<Banana>[10];//illegal不能如此实例化

Peel la = new peel[10]; peels =(peel<Banana>) peels;//unchecked warning

# String

1. length

长度

char chars[]={'a','b'.'c'}; String s=new String(chars); int len=s.length();

1. charAt

截取一个字符

ch="abc".charAt(1); 返回'b'

1. toCharArray()

返回字符数组

1. getChars()

截取多个字符void getChars(int sourceStart,int sourceEnd,char target[],int targetStart)

String s="this getChars method."; char buf[]=new char[20]; s.getChars(10,14,buf,0);

1. substring

截取字符串。

String substring(int startIndex)，String substring(int startIndex,int endIndex)

1. toLowerCase()，toUpperCase()

转换为大/小写

1. trim()

去掉两边空格

1. replace

替换

String replace(char original,char replacement);String s="Hello".replace('l','w');

1. substring

截取字符串。

String substring(int startIndex)，String substring(int startIndex,int endIndex)

1. toLowerCase，toUpperCase

转换为大/小写

构造方法

public String(char[] var1)

public String(String var1)

public String(StringBuffer var1)

public String(StringBuilder var1)

+=是java语言唯一重载的操作符

# Class

Class.forName(xxx.xx.xx.class);

下为某个具体类的class

1、getName()

2、newInstance()

3、getClassLoader()

4、getComponentType() 返回表示数组组件类型的 Class。

5、getSuperclass() 返

6、isArray() 判定此 Class 对象是否表示一个数组类

7、method getMethod(String var1, Class... var2) //参数1方法名 参数2，参数的class 方法名+参数确定方法

8、method []getMethods()

9、Constructor<T> getConstructor(Class... var1)

10、Field getField(String var)

Field、constructor、method都是java.lang.reflection下的类

Method: Object invoke(Object var1, Object... var2)

# Collections

Collections则是集合类的一个工具类/帮助类，其中提供了一系列静态方法，用于对集合中元素进行排序、搜索以及线程安全等各种操作。

Collections.Shuffling(list)//混排

Collections.reverse(list)//反转

Collections.sort(list);//排序

Collections.fill(li,"aaa");//替换所有元素

Collections.copy(list,li);//拷贝后面一个参数是目标列表 ,前一个是源列表 Collections.min(list)

Collections.max(list)

比较大小的成员必须实现comparable