# IO流(序列流)

## 什么是序列流

序列流可以把多个字节输入流整合成一个, 从序列流中读取数据时, 将从被整合的第一个流开始读, 读完一个之后继续读第二个, 以此类推.

## 整合多个

**public** **static** **void** demo2() **throws** FileNotFoundException, IOException {

FileInputStream fis1 = **new** FileInputStream("a.txt");

FileInputStream fis2 = **new** FileInputStream("b.txt");

SequenceInputStream sis = **new** SequenceInputStream(fis1, fis2);

FileOutputStream fos = **new** FileOutputStream("c.txt");

**int** b;

**while**((b = sis.read()) != -1) {

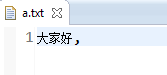
fos.write(b);

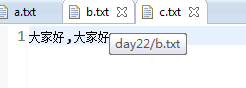
}

sis.close(); //sis在关闭的时候,会将构造方法中传入的流对象也都关闭

fos.close();

}





# 内存输出流

**public** **static** **void** demo2() **throws** FileNotFoundException, IOException {

FileInputStream fis = **new** FileInputStream("e.txt");

ByteArrayOutputStream baos = **new** ByteArrayOutputStream(); //在内存中创建了可以增长的内存数组

**int** b;

**while**((b = fis.read()) != -1) {

baos.write(b); //将读取到的数据逐个写到内存中

}

System.*out*.println(baos.toString()); //将缓冲区的内容转换为了字符串,在输出语句中可以省略调用toString方法

fis.close();

}

# Properties的概述和作为Map集合的使用

A:Properties的概述

\* Properties 类表示了一个持久的属性集。

\* Properties 可保存在流中或从流中加载。

\* 属性列表中每个键及其对应值都是一个字符串。

22.15\_IO流(Properties的特殊功能使用)(了解)

\* A:Properties的特殊功能

\* public Object setProperty(String key,String value)

\* public String getProperty(String key)

\* public Enumeration<String> stringPropertyNames()

\* B:案例演示

\* Properties的特殊功能

**public** **static** **void** demo2() {

Properties prop = **new** Properties();

prop.setProperty("name", "张三");

prop.setProperty("tel", "18912345678");

//System.out.println(prop);

Enumeration<String> en = (Enumeration<String>) prop.propertyNames();

**while**(en.hasMoreElements()) {

String key = en.nextElement(); //获取Properties中的每一个键

String value = prop.getProperty(key); //根据键获取值

System.*out*.println(key + "="+ value);

}

}

###22.16\_IO流(Properties的load()和store()功能)(了解)

\* A:Properties的load()和store()功能

\* B:案例演示

\* Properties的load()和store()功能

**public** **static** **void** demo3() **throws** IOException, FileNotFoundException {

Properties prop = **new** Properties();

prop.load(**new** FileInputStream("config.properties")); //将文件上的键值对读取到集合中

prop.setProperty("tel", "123456");

prop.store(**new** FileOutputStream("config.properties"), **null**);//第二个参数是对列表参数的描述,可以给值,也可以给null

System.*out*.println(prop);

}