

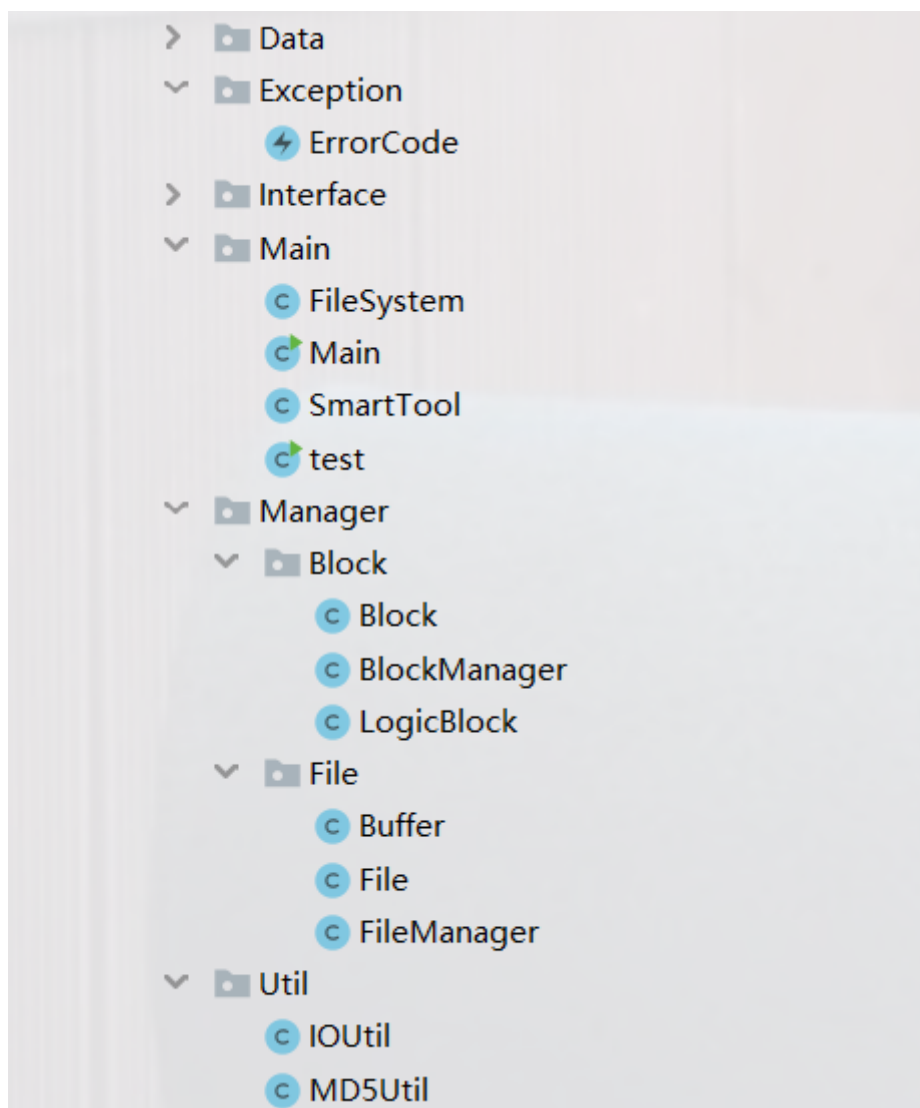
# FileSystem 说明文档

---

姓名：邵荟媛

学号：19302010077

## 代码结构



**Data**——存储文件的meta和块的meta、data

**Exception**——ErrorCode

**Interface**——需要实现的接口

**Main**

FileSystem——将FileManager和BlockManager结合成一个文件系统，并恢复当前的meta信息

smart工具——实现smart\_cat\smart\_hex\smart\_write\smart\_copy\smart\_ls工具

Main——初始化filesystem，实现控制台操作

## **Manager.Block**

Block——对block的读、写

Blockmanager——新建block、根据索引得到相应的block、list当前管理的所有块的信息

Logicblock:

实现duplication,维护一个Block的数组列表

logicBlock number设置为2即一个logicblock对应两个物理block，进行读写时，随机选取一个物理block

## **Manager.File**

Buffer——维护一个byte数组和size

File——对文件的读写、游标的移动、改变文件size

Filemanager——根据id或者filename新建file，根据id或filename得到相应的file，list当前所有文件的信息

## **Util**

IOUtil ——利用哈希表实现对meta文件的读和写

MD5Util——利用md5算法对byte数组进行加密

## **理解**

用户通过filemanager进行文件的获取和新建，得到文件后可以通过File对文件进行读和写等操作，当文件close之后LogicBlock会随机向多个blockmanager中写入block进行物理存储，由block manager对其中的block进行管理

## **ErrorCode**

异常	异常码	异常产生的原因
IO EXCEPTION	1	进行file的meta和block的meta、data的读写时
CLASS NOT FOUND	2	进行读meta时之前存进去的类的文件移动了位置或者被删除，导致找不到类
MD5 FAILED	3	加密checksum时MD5算法出错
CHECKSUM CHECK FAILED	4	读块的data时加密得到的checksum与保存的checksum不一致
INVALID CURSOR	5	移动后的新游标位置小于0或者超过文件的size
WRONG PARAM	6	输入参数长度或范围错误
FILE NOT FOUND	7	getFile时找不到文件
FILE ALREADY EXISTS	8	新建文件时已经有了id或name相同的文件
EOF	9	已经读到文件末尾
BLOCK NOT FOUND	10	文件逻辑块中没有相应的块
BLOCK BROKEN	11	读不到相应块的内容
INDEX OUT OF BOUND	12	索引超过界限
SYSTEM ERROR	13	创建文件失败、存入块的数据长度超出块的size
UNKNOWN	1000	未知错误

## bonus

### buffer

对于每个file维护一个Buffer对象，每次读文件先判断buffer是否为空，若为空则从block中读取相应数据到buffer，再从buffer中读取；写文件先重新设置buffer的大小再进行写入。所有读写都在file close的时候写入block，若写入过程中发生异常，则meta内容不会修改

### 持久化

在初始化一个文件系统的时候会读取file和block的meta内容，从而恢复上一个文件系统所保存的内容

### 命令行

实现了命令行操作

```
Welcome to Your File System
Now you have 3 file managers and 3 block managers
Here are the operations:
'new [FM index] [file name]': create a new file
'read [FM index] [file name] [length](option)': read a file, if you don't enter the length, it will read the full file
'write [FM index] [file name]': write a file
'save [FM index] [file name]': save the changes
'cursor [FM index] [file name]': show the cursor position
'move [FM index] [file name] [offset] [where]': move the cursor, curr=0, head=1, tail=2
'setSize [FM index] [file name] [new size]': reset the file size
'detail [FM index] [file name]': show the details of a file
'smart_cat [FM index] [file name]': read file data directly
'smart_hex [FM index] [block index]': read block data with hex
'smart_write [FM index] [file name] [index]': move the cursor and write the data
'smart_copy [src FM index] [src file name] [dst FM index] [dst file name]': copy the file
'smart_ls': show the file system's structure
'quit': quit the file system
>>>>
```