

## HDFS客户端操作 --- IO流操作

### HDFS文件上传

```
@Test
public void putFileToHDFS() throws IOException, InterruptedException,
URISyntaxException {
    // 1 获取文件系统
    Configuration configuration = new Configuration();
    FileSystem fs = FileSystem.get(new URI("hdfs://hadoop102:9000"),
configuration, "zhutiansama");

    // 2 创建输入流
    FileInputStream fis = new FileInputStream(new File("e:/data.txt"));

    // 3 获取输出流
    FSDataOutputStream fos = fs.create(new Path("/data.txt"));

    // 4 流对拷,关键
    IOUtils.copyBytes(fis, fos, configuration);

    // 5 关闭资源
    IOUtils.closeStream(fos);
    IOUtils.closeStream(fis);
    fs.close();
}
```

### HDFS文件下载

```
// 文件下载
@Test
public void getFileFromHDFS() throws IOException, InterruptedException,
URISyntaxException{
    // 1 获取文件系统
    Configuration configuration = new Configuration();
    FileSystem fs = FileSystem.get(new URI("hdfs://hadoop102:9000"),
configuration, "zhutiansama");

    // 2 获取输入流
    FSDataInputStream fis = fs.open(new Path("/data.txt"));

    // 3 获取输出流,注意Input是从哪里输入, Output是输出到哪里
    FileOutputStream fos = new FileOutputStream(new File("e:/data.txt"));

    // 4 流的对拷
    IOUtils.copyBytes(fis, fos, configuration);

    // 5 关闭资源
    IOUtils.closeStream(fos);
    IOUtils.closeStream(fis);
}
```

```
fs.close();  
}
```

## 定位文件读取

### 1. 下载第一块文件数据【文件如果过大就会分块存储，所以需要分块读取】

```
@Test  
public void readFileSeek1() throws IOException, InterruptedException,  
URISyntaxException{  
    // 1 获取文件系统  
    Configuration configuration = new Configuration();  
    FileSystem fs = FileSystem.get(new URI("hdfs://hadoop102:9000"),  
configuration, "zhutiansama");  
  
    // 2 获取输入流  
    FSDataInputStream fis = fs.open(new Path("/hadoop-2.7.2.tar.gz"));  
  
    // 3 创建输出流  
    FileOutputStream fos = new FileOutputStream(new File("e:/hadoop-  
2.7.2.tar.gz.part1"));  
  
    // 4 流的拷贝  
    byte[] buf = new byte[1024];  
  
    for(int i =0 ; i < 1024 * 128; i++){  
        fis.read(buf);  
        fos.write(buf);  
    }  
  
    // 5关闭资源  
    IOUtils.closeStream(fis);  
    IOUtils.closeStream(fos);  
}
```

### 2. 下载第二块文件数据

```
@Test  
public void readFileSeek2() throws IOException, InterruptedException,  
URISyntaxException{  
    // 1 获取文件系统  
    Configuration configuration = new Configuration();  
    FileSystem fs = FileSystem.get(new URI("hdfs://hadoop102:9000"),  
configuration, "zhutiansama");  
  
    // 2 打开输入流  
    FSDataInputStream fis = fs.open(new Path("/hadoop-2.7.2.tar.gz"));  
  
    // 3 定位输入数据位置  
    fis.seek(1024*1024*128);  
  
    // 4 创建输出流
```

```
FileOutputStream fos = new FileOutputStream(new File("e:/hadoop-2.7.2.tar.gz.part2"));

// 5 流的对拷
IOUtils.copyBytes(fis, fos, configuration);

// 6 关闭资源
IOUtils.closeStream(fis);
IOUtils.closeStream(fos);
}
```

### 3.合并两块文件

按住shift打开cmd，对数据进行合并  
type hadoop-2.7.2.tar.gz.part2 >> hadoop-2.7.2.tar.gz.part1  
合并完成后，将hadoop-2.7.2.tar.gz.part1重新命名为hadoop-2.7.2.tar.gz即可