

# 说明

下面这份“指令文档”默认你已经在 `D:\subjectResource\softwareDesign\pj\demo` 下执行了程序：

```
cd D:\subjectResource\softwareDesign\pj\demo
mvn exec:java
# 或
java -cp target/classes com.editor.App
```

终端里会看到提示：

```
文本编辑器 v1.0
输入 'exit' 退出程序
```

```
>
```

以下所有命令都是在这个提示符下输入，用来专门调试 `ready.txt` 和 `ready.xml`。

# 一、调试 ready.txt 的指令

## 1.1 初始化与基本编辑

# 若没有 ready.txt，可以先创建一个带日志的缓冲区

```
> init ready.txt with-log
```

新缓冲区已创建: ready.txt

# 追加几行文本

```
> append "Hello World"
```

文本已追加

```
> append "Ths is a smple txt line."
```

文本已追加

# 查看所有内容

```
> show
```

1: # log

2: Hello World

3: Ths is a smple txt line.

## 1.2 插入 / 删除 / 替换

```
# 在第2行第7列插入 "Beautiful "  
> insert 2:7 "Beautiful "  
文本已插入
```

```
# 查看结果  
> show  
1: # log  
2: Hello Beautiful World  
3: Ths is a smple txt line.
```

```
# 删除第3行第5列开始的 2 个字符  
> delete 3:5 2  
文本已删除
```

```
# 替换第3行第1列开始的 3 个字符为 "This"  
> replace 3:1 3 "This"  
文本已替换
```

```
# 查看调整后的内容  
> show
```

## 1.3 撤销 / 重做

```
# 撤销上一步替换  
> undo  
已撤销
```

```
# 再撤销一次  
> undo  
已撤销
```

```
# 重做  
> redo  
已重做
```

## 1.4 日志与编辑器列表 / 目录树

# 显示当前打开的文件及编辑时长

```
> editor-list
```

# 打印当前目录树（确认文件存在）

```
> dir-tree .
```

# 查看 ready.txt 的日志

```
> log-show
```

# 如日志未开启，可手动开启：

```
> log-on ready.txt
```

日志已启用：ready.txt

## 1.5 保存 / 关闭 / 重新打开（测试统计模块）

# 保存当前活动文件（ready.txt）

```
> save
```

文件已保存：ready.txt

# 关闭 ready.txt

```
> close
```

文件已关闭：ready.txt

# 再次加载 ready.txt（编辑时长会根据实验要求重置）

```
> load ready.txt
```

文件已加载：ready.txt

# 再次查看编辑器列表和时间

```
> editor-list
```

## 1.6 针对 ready.txt 的拼写检查

```
> spell-check
```

拼写检查结果：

第3行，第5列："smple" -> 建议：simple

如想对指定文件检查（不是当前活动）：

```
> spell-check ready.txt
```

## 二、调试 ready.xml 的指令

### 2.1 初始化 / 加载 XML

#### 情况 A：已有 ready.xml 文件

确保 ready.xml 满足要求（有 XML 声明、唯一 id 等），例如：

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore id="root">
  <book id="book1" category="COOKING">
    <title id="title1" lang="en">Everyday Itallian</title>
    <author id="author1">Giada De Laurentiis</author>
  </book>
</bookstore>
```

在编辑器中加载：

```
> load ready.xml
文件已加载：ready.xml
```

#### 情况 B：从命令行新建 ready.xml

```
> init ready.xml
新缓冲区已创建：ready.xml
```

```
# 默认会创建一个 root[id="root"] 的根元素
> xml-tree
root [id="root"]
```

## 2.2 追加子元素 append-child

```
# 在根元素 root 下追加一个 book 元素
> append-child root book book1 category="COOKING"
子元素已追加
```

```
# 在 book1 下追加 title 和 author
> append-child book1 title title1 lang="en"
子元素已追加
> append-child book1 author author1
子元素已追加
```

```
# 修改 title1 文本
> edit-text title1 "Everyday Itallian"
元素文本已修改
```

```
# 查看树形结构
> xml-tree
```

## 2.3 insert-before / edit-id / delete-element

```
# 在 book1 前插入一个新的 book2
> append-child root book book2 category="CHILDREN"
子元素已追加
```

```
# 在 book2 里面追加 title2
> append-child book2 title title2 lang="en"
子元素已追加
> edit-text title2 "Harry Pottre"
元素文本已修改
```

```
# 在 title2 前插入一个 year 元素
> insert-before title2 year year2
元素已插入
```

```
# 修改元素 ID
> edit-id year2 yearNew
元素ID已修改
```

```
# 删除元素（例如删除 author1）
> delete-element author1
元素已删除
```

```
# 撤销删除
> undo
已撤销
```

## 2.4 xml-tree 输出与保存

```
# 查看 ready.xml 的完整树
> xml-tree
# 或指定文件
> xml-tree ready.xml
```

```
# 保存 XML 文件
> save ready.xml
文件已保存: ready.xml
```

## 2.5 XML 日志启用方式

### 方式 A：通过根元素属性

如果 ready.xml 中根元素写成：

```
<bookstore id="root" log="true">
```

则 load ready.xml 时会自动启用日志：

```
> load ready.xml  
文件已加载：ready.xml  
> log-show
```

### 方式 B：命令行开启/关闭

```
> log-on ready.xml  
日志已启用：ready.xml  
  
> log-off ready.xml  
日志已关闭：ready.xml
```

## 2.6 针对 ready.xml 的拼写检查

ready.xml 中如果有拼写错误（例如 Itallian、Pottre），可以这样检查：

```
> edit ready.xml  
已切换到文件：ready.xml  
  
> spell-check  
拼写检查结果：  
元素 title1: "Itallian" -> 建议：Italian  
元素 title2: "Pottre" -> 建议：Potter
```

也可以显式指定文件名：

```
> spell-check ready.xml
```



### 三、同时调试 ready.txt 和 ready.xml 的指令串（推荐手动复制执行）

# 1. 打开两个文件

```
> load ready.txt  
> load ready.xml
```

# 2. 切换到 ready.txt 做文本编辑与拼写检查

```
> edit ready.txt  
> append "Ths is a smple line."  
> spell-check
```

# 3. 切换到 ready.xml 做 XML 结构编辑

```
> edit ready.xml  
> append-child root book book1 category="COOKING"  
> append-child book1 title title1 lang="en"  
> edit-text title1 "Everyday Itallian"  
> xml-tree  
> spell-check
```

# 4. 查看编辑器列表和各自编辑时长

```
> editor-list
```

# 5. 保存并退出

```
> save all  
> exit
```