

# 一、创建项目，实现EasyExcel对Excel写操作

## 1、pom中引入xml相关依赖

```
1 <dependencies>
2     <!-- https://mvnrepository.com/artifact/com.alibaba/easyexcel -->
3     <dependency>
4         <groupId>com.alibaba</groupId>
5         <artifactId>easyexcel</artifactId>
6         <version>2.1.1</version>
7     </dependency>
8 </dependencies>
```

## 2、创建实体类

设置表头和添加的数据字段

```
1 @Data
2 public class Stu {
3
4     //设置表头名称
5     @ExcelProperty("学生编号")
6     private int sno;
7
8     //设置表头名称
9     @ExcelProperty("学生姓名")
10    private String sname;
11
12 }
```

## 3、实现写操作

(1) 创建方法循环设置要添加到Excel的数据

```

1 //循环设置要添加的数据，最终封装到list集合中
2 private static List<DemoData> data() {
3     List<DemoData> list = new ArrayList<DemoData>();
4     for (int i = 0; i < 10; i++) {
5         DemoData data = new DemoData();
6         data.setSno(i);
7         data.setName("张三"+i);
8         list.add(data);
9     }
10    return list;
11 }

```

## (2) 实现最终的添加操作（写法一）

```

1 public static void main(String[] args) throws Exception {
2     // 写法1
3     String fileName = "F:\\11.xlsx";
4     // 这里 需要指定写用哪个class去写，然后写到第一个sheet，名字为模板 然后文件流会自动关闭
5     // 如果这里想使用03 则 传入excelType参数即可
6     EasyExcel.write(fileName, DemoData.class).sheet("写入方法一").doWrite(data());
7 }

```

## (3) 实现最终的添加操作（写法二）

```

1 public static void main(String[] args) throws Exception {
2     // 写法2，方法二需要手动关闭流
3     String fileName = "F:\\112.xlsx";
4     // 这里 需要指定写用哪个class去写
5     ExcelWriter excelWriter = EasyExcel.write(fileName, DemoData.class).build();
6     WriteSheet writeSheet = EasyExcel.writerSheet("写入方法二").build();
7     excelWriter.write(data(), writeSheet);
8     /// 千万别忘了finish 会帮忙关闭流
9     excelWriter.finish();
10 }

```