## 方法名称：listToExcel

### 方法描述：

集合导出excel

### 方法参数：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 注释 |
| list | List | 要导出的集合 |
| fieldMap | LinkedHashMap | 展示的字段名称 |
| sheetName | String | sheet名称 |
| out | OutputStream |  |
| lockpass | String | 展示的字段名称 |
| protectpass | String | 保护密码 |
| protectMap | LinkedHashMap | 需要保护列的集合 |

### 使用案例：

**public** **class** User {

**private** Integer id;

**private** String name;

**private** Integer age;

**private** String sex;

**get set...**

/\*\*

\* 模拟获取列表

\* **@return**

\*/

**public** List<User> getList(){

List<User> list = **new** ArrayList<>(10);

**for** (**int** i = 1; i <= 10; i++) {

list.add(**new** User(i, "张"+i, (**int**)(Math.*random*()\*100), i%2==0?"男":"女"));

}

**return** list;

}

}

@Test

**public** **void** testListToExcel() **throws** FileNotFoundException {

List<User> list = **new** User().getList();

//sheet名称

String sheetName = "用户信息";

//对应取值属性和表头信息

LinkedHashMap<String, String> fieldMap = **new** LinkedHashMap<>(10);

fieldMap.put("id", "ID");

fieldMap.put("name", "姓名");

fieldMap.put("age", "年龄");

fieldMap.put("sex", "性别");

OutputStream out = **new** FileOutputStream("D:\\user.xlsx");

//要保护的列

LinkedHashMap<String, String> protectMap = **new** LinkedHashMap<>(1);

protectMap.put("name", "姓名");

ExcelPoiUtil.*listToExcel*(list, fieldMap, sheetName, out, "111", "222",protectMap);

}

## 方法名称：excelToList

### 方法描述：

读取excel到集合

### 方法参数：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 注释 |
| in | InputStream |  |
| sheetName | String | sheet名称 |
| entityClass | Class |  |
| fieldMap | String | 展示的字段名称 |
| password | String | 文件密码 |
| protectMap | LinkedHashMap | 需要保护列的集合 |

### 使用案例：

/\*\*

\* 集合导入excel

\* ExcelPoiUtil.excelToList方法测试

\* **@throws** FileNotFoundException

\*/

@Test

**public** **void** testExcelToList() **throws** FileNotFoundException {

FileInputStream in = **new** FileInputStream(**new** File("D:\\user2.xlsx"));

//sheet名称

String sheetName = "用户信息";

//表头信息和对应取值属性

LinkedHashMap<String, String> fieldMap = **new** LinkedHashMap<>(10);

fieldMap.put("ID", "id");

fieldMap.put("姓名", "name");

fieldMap.put("年龄", "age");

fieldMap.put("性别", "sex");

List<User> list = ExcelPoiUtil.*excelToList*(in, sheetName, User.**class**, fieldMap, "111", "222");

**for** (User user : list) {

System.***out***.println(user);

}

}

## 方法名称：excelToList

### 方法描述：

读取全部sheet页面数据 可将一页中数据分成不同对象，注每个sheet页格式需要相同

### 方法参数：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 注释 |
| in | InputStream |  |
| password | String | 文件密码 |
| protectMap | LinkedHashMap | 需要保护列的集合 |
| parts | SheetPart... | 模板组成部分 |

### 使用案例：

/\*\*

\* 导出全部sheet页面数据 可将一页中数据分成不同对象

\* **@throws** FileNotFoundException

\*/

@Test

**public** **void** testExcelToList2() **throws** FileNotFoundException {

FileInputStream in = **new** FileInputStream(**new** File("D:\\user3.xlsx"));

//表头信息和对应取值属性

LinkedHashMap<String, String> fieldMap = **new** LinkedHashMap<>(10);

fieldMap.put("ID", "id");

fieldMap.put("姓名", "name");

fieldMap.put("年龄", "age");

fieldMap.put("性别", "sex");

//要保护的列

LinkedHashMap<String, String> protectMap = **new** LinkedHashMap<>(1);

protectMap.put("name", "姓名");

SheetPart sp1 = **new** SheetPart(User.**class**, "u1", fieldMap, 0, 1, 5);

SheetPart sp2 = **new** SheetPart(User.**class**, "u2", fieldMap, 0, 5, 7);

List<Map<String, List>> list = ExcelPoiUtil.*excelToList*(in, "111", "222", sp1,sp2);

//打印返回的数据

**for** (Map<String, List> map : list) {

List<User> us1 = map.get("u1");

**for** (User user : us1) {

System.***out***.println(user);

}

System.***out***.println("------------------------------------");

List<User> us2 = map.get("u2");

**for** (User user : us2) {

System.***out***.println(user);

}

}

}

## 方法名称：excelToList

### 方法描述：

读取全部sheet页面数据 可将一页中数据分成不同对象，注每个sheet页格式需要相同

### 方法参数：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 注释 |
| out | OutputStream |  |
| lockpass | String | 文件密码 |
| protectpass | String | 保护密码 |
| sVos | SheetVo... | sheet 的 对象 包含sheetName 和 相关part |

### 使用案例：

/\*\*

\* 多sheet多部分数据导出到excel dataToExcel测试

\* **@throws** FileNotFoundException

\*/

@Test

**public** **void** testDataToExcel() **throws** FileNotFoundException {

List<User> list = **new** User().getList();

//对应取值属性和表头信息

LinkedHashMap<String, String> fieldMap = **new** LinkedHashMap<>(10);

fieldMap.put("id", "ID");

fieldMap.put("name", "姓名");

fieldMap.put("age", "年龄");

fieldMap.put("sex", "性别");

OutputStream out = **new** FileOutputStream("D:\\user4.xlsx");

//要保护的列

LinkedHashMap<String, String> protectMap = **new** LinkedHashMap<>(1);

protectMap.put("name", "姓名");

SheetPart sp1 = **new** SheetPart("我是标题1", fieldMap, protectMap, list, 5);

SheetPart sp2 = **new** SheetPart("我是标题2", fieldMap, protectMap, list, list.size());

SheetVo sv1 = **new** SheetVo("用户信息1", sp1,sp2);

ExcelPoiUtil.*dataToExcel*(out, "111", "222", sv1);

}

## 方法名称：dataToExcel

### 方法描述：

多sheet多部分数据导出到excel

### 方法参数：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 注释 |
| out | OutputStream |  |
| lockpass | String | 文件密码 |
| protectpass | String | 保护密码 |
| sVos | SheetVo... | sheet 的 对象 包含sheetName 和 相关part |

### 使用案例：

/\*\*

\* 多sheet多部分数据导出到excel dataToExcel测试

\* **@throws** FileNotFoundException

\*/

@Test

**public** **void** testDataToExcel() **throws** FileNotFoundException {

List<User> list = **new** User().getList();

//对应取值属性和表头信息

LinkedHashMap<String, String> fieldMap = **new** LinkedHashMap<>(10);

fieldMap.put("id", "ID");

fieldMap.put("name", "姓名");

fieldMap.put("age", "年龄");

fieldMap.put("sex", "性别");

OutputStream out = **new** FileOutputStream("D:\\user4.xlsx");

//要保护的列

LinkedHashMap<String, String> protectMap = **new** LinkedHashMap<>(1);

protectMap.put("name", "姓名");

SheetPart sp1 = **new** SheetPart("我是标题1", fieldMap, protectMap, list, 5);

SheetPart sp2 = **new** SheetPart("我是标题2", fieldMap, protectMap, list, list.size());

SheetVo sv1 = **new** SheetVo("用户信息1", sp1,sp2);

ExcelPoiUtil.*dataToExcel*(out, "111", "222", sv1);

}

## 测试实例代码附件：

 