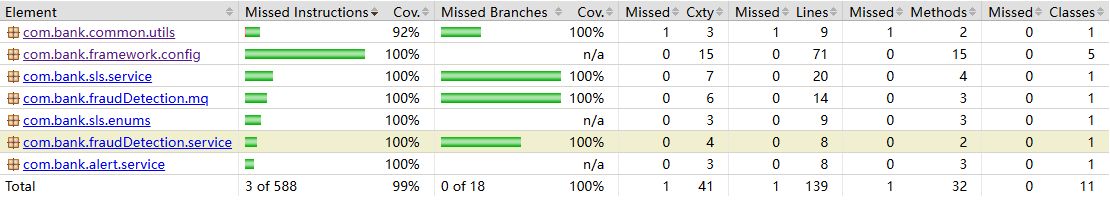
代码覆盖率测试报告

下表详细列出了每个包的覆盖情况：



* **Missed Instructions Cov.:** 未覆盖的指令占比（百分比形式）。
* **Missed Branches Cov.:** 未覆盖的分支占比（百分比形式）。n/a表示该包无分支可覆盖。
* **Missed Cnty:** 未覆盖的圈复杂度条目数。
* **Missed Lines:** 未覆盖的代码行数。
* **Missed Methods:** 未覆盖的方法数。
* **Missed Classes:** 未覆盖的类数

### 关键发现

1. **整体高覆盖率：**

* 总体 **指令覆盖率为 99%** (588条指令中仅3条未覆盖)，表明测试执行了绝大多数的程序指令。
* 总体 **分支覆盖率为 100%** (18个分支中0个未覆盖)，表明所有分支逻辑都被测试到。
* 类、方法、行覆盖率都非常高（大部分包为100%）。

1. **无分支的包：**

* com.bank.framework.config,com.bank.sls.enums, com.bank.alert.service 这三个包被标记为 **Branches n/a**。这表明这些包中的代码（可能是配置类、枚举类等）不包含任何条件分支语句（如 if, switch），因此分支覆盖率指标不适用。它们在其他指标（指令、行、方法、类）上都达到了100%。

1. **其他包：**

* 所有其他包 (com.bank.sls.service, com.bank.fraudDetection.mq, com.bank.fraudDetection.service) 在**所有适用的覆盖率指标**上都达到了**100%** (包含分支覆盖率)，表现优异。

### 总结与建议

* **项目整体代码覆盖率水平非常高（99%指令，100%分支），测试覆盖情况非常良好。**
* **无分支包：** com.bank.framework.config, com.bank.sls.enums, com.bank.alert.service 已实现完全覆盖。
* **其他高覆盖包：** com.bank.sls.service, com.bank.fraudDetection.mq, com.bank.fraudDetection.service 也达到了全面覆盖的要求。

## 附加说明

* 本报告基于提供的图片数据和描述生成。原始工具（如 JaCoCo）的报告可能包含更多细节（如点击包名跳转到源码详情）。
* “Missed Cnty (圈复杂度条目) 通常指未被测试覆盖到的独立决策路径数量。
* 高覆盖率是质量的一个重要指标，但并非唯一指标。测试用例的有效性和对业务逻辑的覆盖同样重要。