

Java2 Final Project 评分标准		最终得分	
	姓名	学号	贡献占比
分组情况			

项目		数据部分评分标准	√	可视化得分
Basic Requirements (60 分) 数据部分得分： 只要展示了相应数据即可得全分 可视化部分得分参考： 1、仅仅使用表格展示数据 - 0% 2、使用了可视化图形，但不能让用户直观地获得想要的答案 - 50% 3、使用了直观的可视化图形呈现数据内容 - 100% 注意，并非所有数据都需要进行可视化呈现，请在每个板块中选择合适的数据进行可视化。	Developers 数据 8 分 + 可视化 2 分 (10 分)	展示了 developers 的总数 (4 分)		
		展示了 commit 数量前几位的 developers 信息 (4 分)		
	Issues 数据 16 分 + 可视化 4 分 (20 分)	展示了 open 的 issue 数量 (4 分)		
		展示了 close 的 issue 数量 (4 分)		
		展示了对 issue 解决时间的典型处理，如平均值、极值差、方差等 (8 分)		
	Releases & Commits 数据 22 分 + 可视化 8 分 (30 分)	展示了 release 的总数 (2 分)		
		展示了 release 间的 commit 数量 (10 分)		
		展示了 commit 的时间分布 (10 分)		

项目		评分标准	得分
Advanced Requirements (12 分)	Multiple repositories (3 分)	可以在前端页面中展示不同的 repo，并查看对应信息及视图 (展示两个及以上 repo 即可得 3 分)	
	REST services (4 分)	Web 服务器可以提供至少三种不同的 RESTful API endpoint (3 分) 通过 API 得到的数据是处理后的数据 (1 分) 展示时，可以通过浏览器直接访问 endpoint 或采用其他合适的方式。不要求可以通过外网访问，只在本地即可。	
	Issue topics (5 分)	对当前 repo 中的 issues 的 title, comment, description 等文本内容进行了收集 (1 分) 文本内容的分析处理 (2 分) 结果的可视化呈现 (2 分)	
Data Collection & Storage (10 分)	使用 Java 爬取了数据，并对数据进行了存储 (10 分) 如果使用其他语言爬取数据，此项不得分 请在报告中简要介绍使用爬虫的框架和思路		
Web Framework (10 分)	使用了 SpringBoot 作为项目框架 (10 分) 如果使用其它后端框架，此项不得分		
Frontend (5 分)	前端可以呈现可视化图形，进行用户交互操作，评分占比参考如下： 前端页面呈现了直观的可视化图形，具备一定美观度 - 2 分 前端页面有易于操作的用户交互系统 - 2 分 前端页面整体的布局和呈现简洁美观 - 1 分		
Documentation (3 分)	报告打分参考如下： 报告中包括了所有要求的内容 - 1 分 报告整体清晰、美观，易于阅读 - 1 分 project 整体的 insight 部分具有一定实际意义 - 1 分		