一、今天做了什么？

1. 深入学习了Git的使用，特别是关于“创建与合并分支”的内容，通过实践操作加深了对Git版本控制的理解。
2. 参加了关于“如何把坏代码改成好代码”的课堂学习，掌握了代码重构的基本原则和技巧。
3. 与团队成员进行了讨论，确定了通过随机数的形式进行分组小程序的设计方案，并开始着手修改“摇一摇”程序，以更好地实现随机分组的功能。
4. 使用微信开发者工具继续小程序的开发，根据团队讨论的结果对程序进行了相应的调整和优化。

二、今天最有收获的点。

今天最大的收获是在Git的学习中掌握了分支管理的技巧，这对于未来的项目协作和版本控制将起到至关重要的作用。同时，在课堂上学到的代码重构知识也让我对如何提升代码质量有了更深刻的认识。

三、今天解决了什么问题？解决方案。

1. 在深入学习Git的分支管理功能时，我遇到了如何高效合并分支并处理可能出现的合并冲突的问题。这对于保持代码库的整洁和团队协作的顺畅至关重要。

**解决方案**：我首先通过Git的官方文档和多个高质量的在线教程进行了系统学习，特别是针对git merge和git rebase命令的详细用法和最佳实践。通过实践，我掌握了如何选择合适的合并策略，以及如何手动解决合并冲突。此外，我还学习了使用Git的图形界面工具（如GitKraken或SourceTree）来辅助分支管理，这些工具提供了更直观的界面来查看和管理分支，大大提高了工作效率。

1. 在修改小程序的“摇一摇”功能时，我发现原有的程序逻辑较为简单，无法直接满足我们新提出的随机分组需求。这要求我们必须对程序进行重构，以引入更复杂的逻辑来处理随机数的生成和分组。

**解决方案**：我首先分析了原程序的逻辑结构，并确定了需要修改的关键点。随后，我设计了一套新的算法来生成随机数并进行分组。这个算法考虑了多种因素，如分组数量、每个组的人数限制等，以确保分组的公平性和随机性。最终，我将新算法集成到小程序中，并进行了多次测试以验证其效果。

四、今天还有什么问题没解决，接下来的大致计划。

1. 虽然掌握了Git的基本操作，但在实际项目中如何高效利用Git进行团队协作还需要进一步学习和实践。计划在接下来的时间里，通过参与更多的项目来积累经验。
2. 小程序的修改和优化工作仍在进行中，特别是随机分组功能的实现还需要进一步调试和完善。计划与团队成员保持紧密沟通，及时解决遇到的问题，并争取在预定时间内完成开发任务。