

UM06 报修处理与报表管理

报修处理功能

功能描述

报修处理功能旨在处理宿舍中学生提交的报修请求。管理员通过此功能能够查看所有未处理的维修请求，并根据请求内容决定是否处理该请求。处理后，系统会更新报修请求的状态，并在房间没有其他未处理报修的情况下，将房间状态更新为“正常”。该功能的核心在于确保维修请求得到及时的响应，同时合理更新宿舍房间状态。

具体步骤

1. 查询未处理的报修请求

首先，系统会查询数据库中的所有“未处理”状态的维修请求，并显示其详细信息，包括报修ID、学生ID、房间ID、报修种类、描述以及报修时间。这些信息供管理员参考，帮助其决定处理哪些报修请求。

2. 检查是否存在待处理报修请求

如果没有未处理的报修请求，系统会提示管理员当前没有待处理的报修请求，并直接退出处理流程。

3. 管理员选择报修请求

管理员通过输入报修ID来选择具体的报修请求。如果输入的ID无效或不存在，系统会进行提示并要求重新选择。

4. 确认处理报修请求

一旦管理员选择了有效的报修ID，系统会向管理员确认是否确认处理该报修请求。如果管理员选择“是”，系统将继续执行处理步骤，否则会取消操作。

5. 更新报修请求状态

如果管理员确认处理报修请求，系统将更新数据库中的报修请求状态为“已处理”，并记录处理时间。处理完成后，管理员可以继续进行其他操作。

6. 更新房间状态

根据报修请求的房间ID，系统会检查该房间是否还有其他未处理的维修请求。如果该房间还有待处理的报修请求，系统会将房间状态保持为“维修中”；否则，系统会将房间状态更新为“正常”。

遇到的问题与解决方案

• 问题一：查询未处理报修请求时出现结果为空

在查询未处理的维修请求时，可能会遇到没有待处理报修的情况。此时，系统应提示管理员

当前没有待处理请求。

解决方案：通过检查查询结果，如果返回值为空，则输出相应提示，避免系统误操作。

- **问题二：管理员输入无效的报修ID**

如果管理员输入了无效的报修ID（例如不存在的ID或已处理的ID），系统需要验证ID的有效性。

解决方案：通过SQL查询判断输入的报修ID是否存在且状态为“未处理”，若无效则提示管理员重新输入。

- **问题三：房间状态更新时的逻辑错误**

如果房间仍有未处理的报修请求，房间状态不应直接更新为“正常”。

解决方案：在更新房间状态之前，先通过查询检查该房间是否有其他未处理的维修请求。

从中获得的收获

- **数据验证的重要性：**在处理报修请求的过程中，确保输入数据和系统中的数据一致性非常重要。通过有效的查询和验证，可以避免因错误操作导致的系统异常。
- **逻辑思维的锻炼：**在开发过程中，如何处理不同状态之间的关系，尤其是在房间状态与报修请求之间的逻辑判断，增强了我的逻辑思维能力。
- **用户体验的提升：**通过设计简洁、清晰的操作流程，确保管理员能够快速、准确地完成任务，提高了系统的易用性。

报表管理功能

功能描述

报表管理功能主要用于生成宿舍的入住率报表。管理员可以根据时间范围（当前月或当前年）生成对应的入住率报表，帮助宿舍管理人员及时掌握宿舍的入住情况。该功能通过查询数据库，统计每个宿舍楼在特定时间范围内的入住人数，并计算入住率，最终以表格形式展示给管理员。

具体步骤

1. **选择时间范围**

管理员可以选择生成报表的时间范围：当前月或当前年。系统会提供选择界面，并根据用户输入的选择生成相应的报表。

2. **生成当前月的入住率报表**

如果选择当前月，系统会查询宿舍楼中所有房间在当前月份内的入住情况，统计每个宿舍楼的总房间容量，并计算入住率。入住率的计算方式为：已入住人数 / 总房间容量。

3. **生成当前年的入住率报表**

如果选择当前年，系统会查询宿舍楼中所有房间在当前年度内的入住情况，统计每个宿舍楼的总房间容量，并计算入住率。此过程与当前月的报表类似，只是时间范围不同。

4. **查询并展示报表结果**

生成SQL查询语句后，系统会执行查询操作并展示报表结果。管理员可以看到每个宿舍楼的

名称、已入住人数、总房间容量和入住率。报表结果会清晰地展示出来，供管理员分析和决策。

5. 操作提示

在报表生成过程中，系统会显示“正在查询中”的提示，查询完成后再显示报表结果。用户可以按任意键继续。

遇到的问题与解决方案

- **问题一：无效的时间范围选择**

如果管理员输入了无效的时间范围（如不选择时间范围或选择错误的选项），系统需要进行提示并要求重新输入。

解决方案：通过检查用户输入的时间范围是否有效，若无效，则提示重新输入。

- **问题二：查询结果为空**

在某些情况下，查询结果可能为空（例如没有入住记录）。这时，系统应确保能够正常处理空结果，并提示管理员相应信息。

解决方案：使用条件判断，确保查询结果为空时不会影响系统运行，且向管理员提供明确的反馈。

- **问题三：长时间查询操作**

由于报表生成涉及大量数据查询，可能会出现长时间查询的情况。

解决方案：通过在查询过程中添加进度提示（如“正在查询中”），提升用户体验，避免管理员误以为系统出现了问题。

从中获得的收获

- **数据统计与查询优化：**在处理报表生成时，通过优化SQL查询语句和合理的数据库索引，提升了查询效率，减少了长时间等待。
- **用户交互的设计：**通过提供简单明了的选择界面，使管理员能够轻松选择时间范围，避免因界面复杂而产生误操作。
- **动态提示与反馈机制：**在查询过程中加入进度提示，增强了系统的互动性和用户体验，让管理员在等待过程中感到更为流畅和舒适。

总结

通过实现报修处理和报表管理功能，**UserManager** 系统的功能得到了显著扩展。这两个功能不仅提升了管理员的工作效率，同时也增强了系统的交互性和可操作性。在开发过程中，我深刻体会到数据验证、逻辑设计、用户体验等多方面的重要性。通过合理的数据库查询和界面设计，能够使管理员高效、准确地处理任务，并进一步提高宿舍管理系统的综合水平。