

2022 北京冬奥会 信息管理系统

数据库项目实验总结

郑懿 未央-水木 01 2020012859

李晗 未央-水木 01 2020012856

一、 遇到的问题及解决方案

1. 用户输入合法性的保证

我们对于用户输入合法性的保证是通过提示和警告这两种方式来完成。

首先，在新用户注册的时候，我们通过在输入框中加入灰色的提示性字符来对用户的输入进行提示。并且我们对“性别”这一属性采用了下拉选择框的形式，以防止用户有误输入



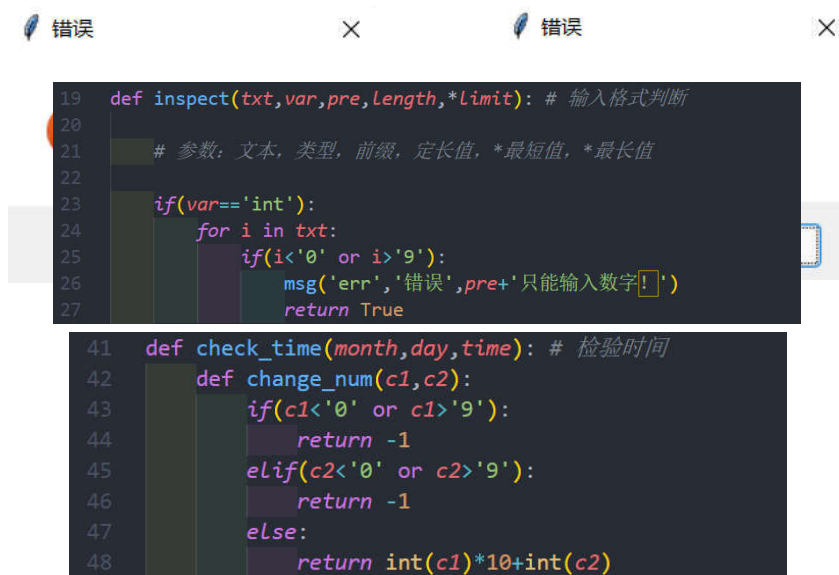
的情况发生。

如果用户确实输入了非法的输入，例如将年龄设置过大或是两次密码输入不一致，则我们的程序都会发出警告。

我们设置了专门的函数 `inspect()` 和 `check_time()` 来进行输入合法性的判断，从而避免了在每一处需要判断合法性的输入位置都重复编写函数的冗余操作，从而大大提高了程序的可读性。

2. 商品和门票销量数据的可视化

在设计之初，我们就设想了对商品和门票销量数据进行可视化操作，从而为管理者提供更多信息。



实际设计过程中，对于门票，由于其本身的属性就包括了总量（Total）和余量（Remain），

```
select mno, event, total - remain as sold from match order by sold desc
```

因此只需要一条简单的 SQL 语句即可实现对门票销量的提取。

但是对于商品，由于在数据库设计时仅有存量（Remain）这一属性，因此对其销量的读取较为复杂，需要采用 Group by 语句以及 sum 函数进行聚合搜索，在这一过程中我们对 Group by 的使用出现了问题，在开始时我们并没有将 Iname 属性（下图中红框圈出的部分）加入 Group by 中，从而导致一直报错“非 Group by 语句”，在搜索后我们发现所有 select 后的关键词（除了聚合函数外）都需要添加到 Group by 之后，从而解决了这一问题，也丰富了对于 Group by 语句的使用经验。

```
select iname,ino, sum(quantity) from(select iname, itemsale.ino, quantity
from itemsale ,item where item.ino=itemsale.ino)
group by ino,iname order by sum(quantity) desc
```

对于数据的可视化部分，我们调用了 matplotlib.pyplot 包，制作了降序排列的柱状统计

```
583 def ticket_summary_data(*arg):
584     #coding:utf-8
585     import matplotlib.pyplot as plt
586     plt.rcParams['font.sans-serif']=['SimHei'] # 用来正常显示中文标签
587     plt.rcParams['axes.unicode_minus']=False # 用来正常显示负号
588     # 有中文出现的情况，需要u'内容'
589     data = ticket_statistical()
590     # data形式为（比赛编号，比赛名称，销量）
591     # print(data[0][1])
592     name_list = []
593     sold_list = []
594     for i in range(0,len(data)):
595         name_list.append(data[i][1])
596         sold_list.append(data[i][2])
597     fig = plt.bar(range(len(sold_list)), sold_list,tick_label = name_list)
598     # 添加数据标签 就是矩形上面的数值
599
600
601     def add_labels(rects):
602         for rect in rects:
603             height = rect.get_height()
604             plt.text(rect.get_x() + rect.get_width()/2, height+0.01*height, '%.0f'%height, ha='center')
605             rect.set_edgecolor('black')
606     add_labels(fig)
607     plt.title('门票销量统计图')
608     plt.show()
```

图，使销量结果十分直观地呈现。

3. 登录界面出现的 bug 及其修复

我们发现，在群众申请成为志愿者但未被分配任务的状态下，群众的登录会出现问题，仔细找寻原因后，我们发现这是由于 select 语句返回的任务（assign）属性都是空的，从而使得检查函数缺失参数而报出错误，于是我们考虑了这一特殊状态，进行了如下修复，从而解决了这一问题。

```
else:
    if(assign==None):
        res = [state, account, name, sex, age, ('无','无')]
        return res
```

4. 志愿者审批界面出现的 bug 及其修复

在调试中我们发现，假如志愿者审批界面有多个待审批的志愿者，如果同时选中进行审批，理论上是可行的，但实际操作中报错。检查发现，当我们读取多个志愿者信息元组时，序号应为 0、1、2...但实际操作时，当操作完序号为 0 的信息时，我们将其删去了，这就导致此后操作的信息元组序号出现问题，检查后，我们修复了此问题。

5. 编译时由于数据库版本原因产生的问题及其解决

由于在当时安装数据库时我们二人的版本不一，分别为 64 位客户端和 32 位客户端系统，因此编译时产生了问题，上网搜索后，了解到可以通过安装在 32 位客户端的计算机上安装 Instant Client 来使得其可以运行 64 位的程序。

```
3 #32位的Oracle系统可以通过安装instantclient并运行下面两行代码成功运行在64位的python环境，记得修改路径！
4 import os
5 os.environ['path'] = r'D:/Codefield/CODE_python/instantclient_21_3'
```

6. 代码共同协作

采用 GitHub 进行了代码的合作编写，为以后的开发积累更多经验。

代码已经开源在 GitHub: <https://github.com/LiXiaoHan0/Database>

7. UI 设计

界面 UI 部分我们从窗体到内部的控件都是采用代码语句实现的,而不是像 PowerBuilder 中采用点击控件和添加的方式实现。我们采用了 Python 中的 GUI 模块 Tkinter 进行了 UI 界面的设计,实现了较为完善的功能,并且在登录界面中加入了带有冬奥元素的图片,使得该



程序与主题更加契合。

二、收获与感受

距离 2022 年北京冬奥会开幕已不到 40 天。作为东道主,北京和张家口正在紧锣密鼓地做着最后冲刺准备,即将迎接世界各地的冰雪健儿。众多外媒也将目光汇聚中国,热切关注这场冰雪盛会的来临。在新冠肺炎疫情持续蔓延的特殊背景下,国际社会期待北京冬奥会顺利举办,给世界带来更多活力与信心。在这样的背景下,良好的赛场保障和志愿服务就显得尤为重要。我们作为清华大学的学生,自然也应当尽自己的一份力,这也是我们选择这一实验题目的原因。李晗同学作为冬奥会志愿者团队的一员,也努力想让我们的程序尽可能接近真实情况,因此我们做了很多的调试,在时间紧张的期末复习阶段努力抽出时间完善程序,当真正完成的时候也非常有成就感。

数据库的课程看似只有三学分,但实则包罗万象,从关系代数到 SQL 语句,从数据库内部的组织到数据库外部的操纵,在课上都有所涉及。如果仅是用一些书面作业作为巩固,则多少有些纸上谈兵的意味。因此,我们非常感谢老师能够提供这一次大实验的机会,不仅让我们实际动手操练了课上的知识,更是锻炼了我们的综合能力。实验的过程固然是漫长的,但这也是最真实的项目开发体验,在此感谢老师和助教对我们的指导和帮助。