实验报告

报告标题：实验数据准备

学号：19200312

姓名：成文涛

日期：2022/10/24

# 一、实验环境

1. 操作系统：

Window10

2. 数据库管理软件（含版本号）：

Postgresql4

3. 其他工具：

Pycharm

# 实验内容及其完成情况

自动生成记录的方法在Student、Course和SC中分别插入100万、1万和1000万条记录。

**表1**:Student表的结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sno | Sname | Ssex | Sage | Sdept |
| 100000000 | 1000000 | 男 | 20 | CS |
| ... | ... | ... | ... | ... |

**表2**:Course表的结构

|  |  |  |  |
| --- | --- | --- | --- |
| Cno | Cname | Cpno | Ccredit |
| 1 | 数据库 | 5 | 4 |
| ... | ... | ... | ... |

**表3**:SC表的结构

|  |  |  |
| --- | --- | --- |
| Sno | Cno | Grade |
| 100000000 | 1 | 92 |
| ... | ... | ... |

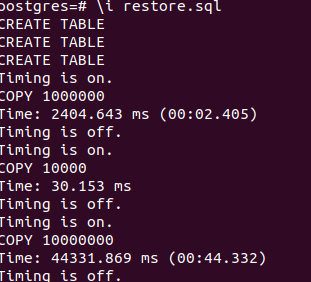
# 构造数据：

一种方法是利用sql的脚本构造数据，但是复杂的sql脚本对于我现在具有较高的难度。第二种是用爬虫从网网络上爬取大量的数据，这种数据往往是显得较为真实的数据，但爬虫技术的应用我也不是很熟悉。

鉴于第二次作业的实践，restore数据库时是以restore.sql等sql脚本和xxxx.dat文件的结合，我利用python生成了这些数据[1]。生成数据时应考虑实体完整性，要使得生成的Sno，Cno等主码是独一无二的，且Cpno等外码必须参照给定的表。

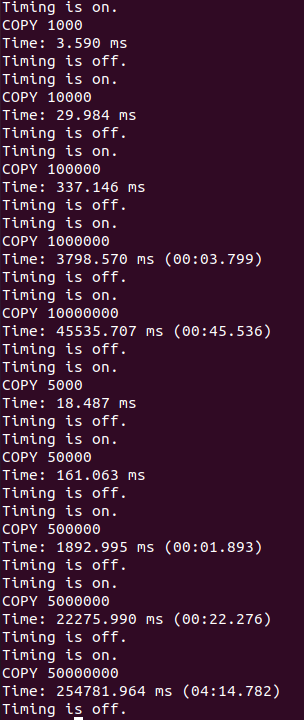
# 数据导入：

在Student、Course和SC中分别插入100万、1万和1000万条记录，编写sql脚本进行导入，结果见图1。

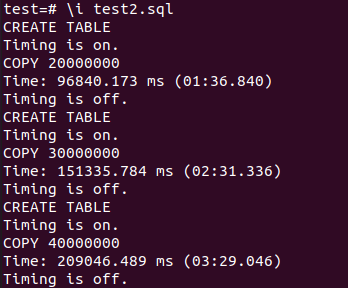


**图1**

接下来测试在不同数据量情况下完成插入记录的性能，插入的表的结构为Course表的结构。插入数量分别为1e3、1e4、1e5、1e4、1e5、1e6、1e7、5e3、5e4、5e5、5e6、2e7、3e7、4e7、5e7。结果见图2、3。

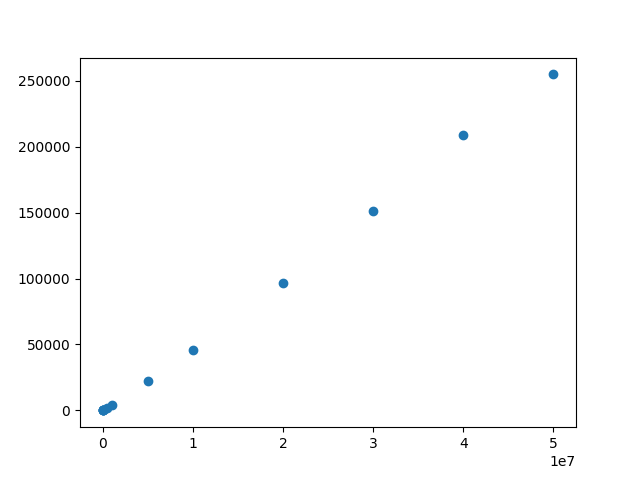


**图2**

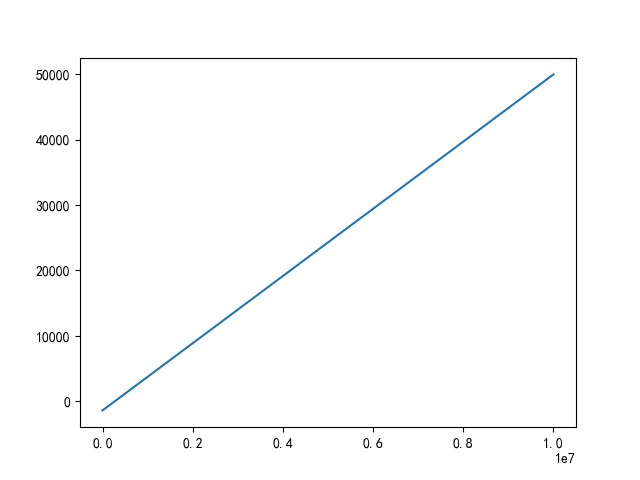


**图3**

把得出的结果画成散点图，得到图4。所用的时间与插入的条数应该成正比例的关系。多项式拟合为0.005133 x - 1377，见图5。

****

**图4**

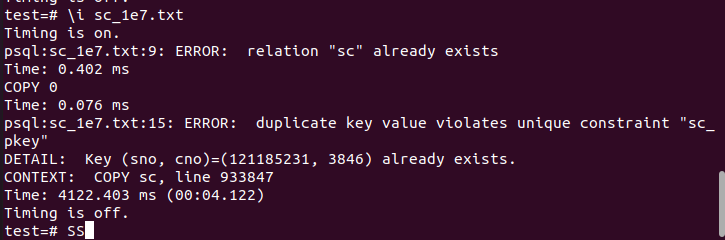


**图5**

# 三、实验总结

在做实验时，生成的数据太大，我第一次创的虚拟机分配的内存比较小，把我虚拟机的硬盘撑爆了，然后整个数据好像就坏了，总之是连接不上去了。后面就在网上学虚拟机重新分配空间的教程，因为对Linux的硬盘管理不是特别了解，结果在重分配的时候，把内核搞坏了，我猜测是删除sda5,然后重创sd5时出了问题。最后只能重新创建个虚拟机，重下postgresql，对Linux的磁盘管理的理解还不是很清楚。

若在生成随机sc\_data中出现多个相同主键，插入时发生完整性冲突，结果为postgresql的处理方式是全都不往里面插入。见图6。



**图6**

参 考 文 献

[1] python生成随机姓名，https://blog.csdn.net/qq\_41426326/article/details/91975774