

第三次课外作业

编程题:

- 1、使用 `input()` 函数, 输入一个包含 10 自然数的列表, 输出这些自然数的平均值。(提示: 可使用 `eval()`、`sum()` 和 `len()` 函数)
- 2、使用列表推导式创建一个列表, 用 `random` 库的子函数(`randint` 或 `randrange` 或 `choice`) 随机生成 10 个 1000 以内的自然整数。
 - (1) 使用排序函数 `sorted()` 对该列表中的数据按降序排列。
 - (2) 使用排序函数 `sorted()` 对该列表中的数据按字符串类型升序排列。
 - (3) 使用排序函数 `sorted()` 对该列表中的数据按字符串的长度降序排列。
- 3、使用 `input()` 函数, 输入一个包含若干个自然数的列表, 输出这些整数的乘积, 如输入 `[-8,2,4]`, 输出 `-64`。(提示: 可使用 `functools` 库中的 `reduce()` 函数, 也可以使用遍历循环进行累乘运算)
- 4、通过列表推导式, 使用 `random` 库中的 `randrange()` 函数, 随机生成一个包含 15 个 50 以内自然数的列表, 输出由其中奇数组成的新列表。例如: 输入 `[1,3,4,5,6,7,2]`, 输出 `[1,3,5,7]`。
- 5、通过列表推导式, 使用 `random` 库中的 `choice()` 函数, 从 `[1,20]` 中随机生成 50 个自然数的列表, 输出列表中最大值所对应的索引下标。(提示: 最大值可能会重复多个, 参考上课案例, 可使用 `count()` 和 `index()` 两个函数循环求解, 也可使用 `enumerate()` 函数遍历序列解包求解)
- 6、通过列表推导式, 使用 `random` 库中的 `choice` 方法, 使用 `string` 库中 `ascii_letters` 英文字符串, 从 52 个大小写英文字符串中随机抽取 10000 个字符, 用字典统计出每个字符被抽取的次数, 并输出出现次数最多的前五个字母及出现的次数。(提示: 可使用字典的 `get()` 方法求解字符出现的次数, 也可以使用字典推导式生成, 然后再对字典进行排序求解)
- 7、某学院有四个年级(2020,2021,2022,2023), 每个年级分别有“临床医学”、“儿科学”和“影像学”三个专业, 每个专业有三个自然班(01,02,03), 每个自然班有 30 名学生, 共计 1080 名学生, 每个学生的信息包括: 学号(`stid` 由 10 位数字构成, 1-4 位是入学年度, 5-6 位是学院专业编号, 7-8 位是同年级专业班级编号, 9-10 位是班级学生的序号, 如“2021110125”)、姓名(`stname`, 如“张国庆”)、班级名(`bjname`, 如“临床医学 2021-02 班”), 电话(`phonenum`, 如“13801239811”)及高等数学(`math`)、英语(`eng`)、计算机(`compu`)三门课程的成绩。

数据源: 要求导入 `faker` 及 `random` 库, 通过列表推导式生成学号、姓名、班级名, 电话, 三门课程的成绩可使用 `random` 库的子库从 `[45,100]` 范围内随机抽取。

- (1) 使用 `zip()` 函数将学号(`stid`)、姓名(`stname`)、班级名(`bjname`)、电话(`phonenum`)及高等数学(`math`)、英语(`eng`)、计算机(`compu`)三门课程的成绩, 共七个数据集成到一个 `zip` 对象容器中, 为了便于操作, 可转为列表对象 `stinfomation`。
- (2) 列表中包含着 1080 名的学生信息, 每个学生信息的 7 个数据保存在一个元组对象中。若要再为每个学生信息增添一个总分(`total`)数据, 其值是三门课程成绩之和, 请编程实施。(提示: 可参考上课讲解, 先求解每个学生的三门课程总分, 再将每个学生的总分加入到学生信息的元组中)
- (3) 若要按总分升序对列表中的学生信息进行排序, 根据排序结果, 输出总分最高的前三名学生的学号、姓名、班级名和总分四项数据, 每名学生的数据占一行。

输出格式: 学号: xxxxxxxxxxx 姓名: xxxxxx 班级名: xxxxxxxx 总分: xxx

- (4) 如对总分进行降序排序时, 总分相同, 再按 `math` 成绩降序排列, 输出总分最低的前三名学生的姓名、高等数学和总分数据, 输出格式同上。
- (5) 思考: 能否用切片方法求解出每个班级总分最高(或最低)的学生信息数据?

作业要求: 请同学们在 jupyter notebook 环境中完成以上内容, 每题的题目存放在(code 代码)cell 上方的(markdown 标记) cell 中。每题的代码可以根据完成的功能放在多个 cell 中, 每个 cell 中的程序代码要能正确的运行。