

说明

- 本作业限时 2 天完成；
- 你可以选用擅长的语言完成，包括但不限于 C、C++、Java、C#、Javascript、Python、Scala 等；
- 可使用第三方库简化代码（如时间、集合操作等）
- 作业的输入和输出必须和题目的要求严格一致；
- 作业至少能够证明针对范例中的输入，能够得到范例中的输出。
- 单元测试非必须，但属于加分项，建议提供。
- 作业完成后必须附上 Readme 纯文本文档（推荐使用 markdown 排版）。文档中应当描述如何运行作业中的单元测试或作业中的代码来证明作业的正确性（至少针对范例输入能够得到范例输出）。

题目-作业

小明是一个羽毛球爱好者。考虑到有很多人喜欢打羽毛球但找不到球友，他打算成立羽毛球俱乐部，不定期组织羽毛球活动。活动可以是每周的任一时段，每次时间为2小时或3小时。参加活动的人每次付给小明30元，小明向球馆支付总费用。

小明每次会根据报名人数决定向球馆预定若干球场，通常会按6个人一个场地预定。但小明发现报名人数多数情况并不是6的整倍数，到底该不该为多出来的人预订场地，让小明很头疼。

经过一段时间的摸索，小明找到了一种让大多数人满意的方式：

- 记参加一次活动的总人数为 M
- 记 T 为可以立即确定的场地数，其值为 $M/6$ （整除）。例如 M 为 6 人，则 T 为 1； M 为 5，则 T 为 0
- 记 x 为多出来的人，其值为 $M \% 6$ （求模）。比如 M 为 6，则 x 为 0； M 为 8，则 x 为 2（提醒：按求模的定义， x 的值小于 6）。

则订场策略如下：

| T | X | 订场策略 |
|-------|----------|-------|
| 0 | <4 | 取消活动 |
| 0 | ≥ 4 | 共订1个场 |
| 1 | 任意 | 共订2个场 |
| 2 或 3 | ≥ 4 | 多订一个场 |
| >3 | 任意 | 不多订场 |

比如只来了3个人，则取消活动；来了5个人，则订一个场；来了7个人，则订2个场；来了29个人，订4个场；等等。

同时，小明选择的羽毛球馆每块场地的收费标准如下：

1. 周一到周五：
 - 9:00 ~ 12:00 30元/时
 - 12:00 ~ 18:00 50元/时
 - 18:00 ~ 20:00 80元/时
 - 20:00 ~ 22:00 60元/时
2. 周六及周日

- 9:00 ~ 12:00 40元/时
- 12:00 ~ 18:00 50元/时
- 18:00 ~ 22:00 60元/时

每过一段时间，小明会根据活动情况计算收入。请编写程序帮助小明计算收入。

要求

实现以下函数或方法，例如：

- Javascript: `generateSummary(input)`，参数 `input` 是如下面范例所示的输入（字符串），返回值为生成的汇总信息（字符串）。
- Java: `String generateSummary(String input)`
- C#: `string GenerateSummary(string input)`

注意，你可以对该函数的实现进行任意的设计。例如，划分子函数或创建新的类型。

输入为字符串，其格式定义如下

- 每一行代表一次活动，其格式为 {活动时间 yyyy-MM-dd HH:mm~HH:mm} {人数}
- 每一行的时间不会有重叠，其时间必然为整小时。

范例：

```
2016-06-02 20:00~22:00 7
2016-06-03 09:00~12:00 14
2016-06-04 14:00~17:00 22
2016-06-05 19:00~22:00 3
2016-06-06 12:00~15:00 15
2016-06-07 15:00~17:00 12
2016-06-08 10:00~13:00 19
2016-06-09 16:00~18:00 16
2016-06-10 20:00~22:00 5
2016-06-11 13:00~15:00 11
```

输出为字符串，针对上述范例，应当有如下的输出

[Summary]

```
2016-06-02 20:00~22:00 +210 -240 -30
2016-06-03 09:00~12:00 +420 -180 +240
2016-06-04 14:00~17:00 +660 -600 +60
2016-06-05 19:00~22:00 +0 -0 0
2016-06-06 12:00~15:00 +450 -300 +150
2016-06-07 15:00~17:00 +360 -200 +160
2016-06-08 10:00~13:00 +570 -330 +240
2016-06-09 16:00~18:00 +480 -300 +180
2016-06-10 20:00~22:00 +150 -120 +30
2016-06-11 13:00~15:00 +330 -200 +130
```

```
Total Income: 3630
Total Payment: 2470
Profit: 1160
```

其中：

- 每一行代表一次活动的收支情况。格式为 {活动日期: yyyy-MM-dd} {活动时间段: HH:mm~HH:mm} {收入: +数字} {支出: -数字} {结余}
- 其中，结余的部分如果大于 0 则格式为 +数字，如果小于 0 则格式为 -数字，如果等于 0 则直接为 0。
- 活动的顺序与输入保持一致。