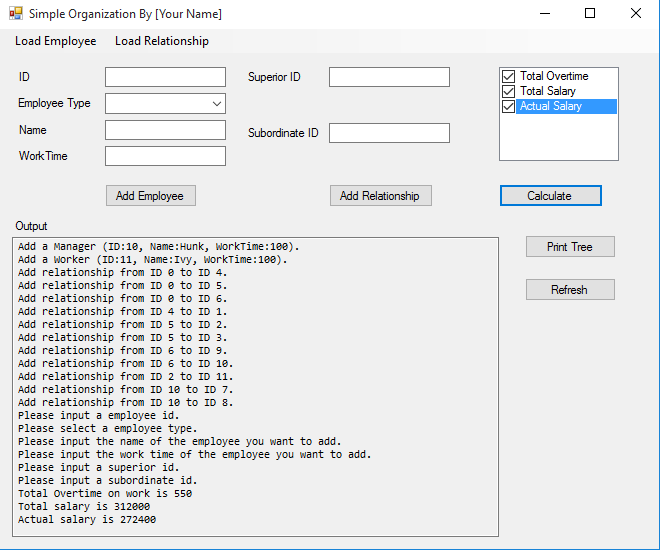
通过学习C#及面向对象设计模式的教程，编写一个基本功能如下图所示的公司管理程序，并在窗口左上角注明姓名。



基本要求如下：

1. **通过填写表单并点击Add Employee按钮，完成添加Employee操作。**

该公司有且仅有四种Employee类型，级别从高到底依次为：Director, Senior Manager, Manager, Worker。每类Employee都有ID，Name，WorkTime这三个属性。其中ID是非负整数，是Employee的唯一标识；Name是Employee姓名；WorkTime是非负整数，表示Employee的当月工作时长。

1. **通过填写表单并点击Add Relationship按钮，完成添加Employee间的上下级关系。**

该公司架构呈树状结构:

1. 公司有且仅有一名Director。
2. 除Director没有上级，其他所有Employee有且仅有一位直属上级，且其职位级别比自己的职位级别高。
3. **点击菜单上的Load Employee和Load Relationship可以分别读取employee.txt和relationship.txt文件批量导入信息。**

employee.txt的每一行是待添加的employee信息，遵循以下格式：

id, employee type, name, work time

relationship.txt的每一行是待添加的relationship信息，遵循以下格式：

superior id, subordinate id

1. **使用设计者模式实现对公司所有Employee信息的统计并输出结果，包括：总的工作时间Total Worktime，所有Employee未扣税工资的总和Total Salary，所有Employee税后工资的总和Actual Salary。**
2. Director:

普通时薪300元，加班时薪600元，Work time>30则算作加班。税收计算简化为其工资的80%

1. Senior Manager:

普通时薪200元，加班时薪400元，Work time>40则算作加班。税收计算简化为其工资的80%

1. Manager:
2. Worker：