1. 在当前任务节点审批通过时，查询下一步流转可以分配的所有角色的用户集合，查询已经分配给这些用户的任务量并根据任务量排序，然后把当前任务分配给其中已经分配到任务量最小的用户。实现根据工作负荷安排工作。
2. SpringCloud在代码开发上更注重服务性，日志更注重系统间的交互，异常链更注重异常系统间传递，管理更注重自动化，故障排除更注重有迹可寻
3. 实体类要参与进行比较必须实现Comparable接口，重写chompare方法；

参与集合类排序，还可以通过传入Comparator的方法。

一般实现的比较方法，要满足几个特性：自身相等，对称性，传递性等。

1. Serializable接口，作为标识对象可以被序列化的接口，不需要实现任何方法；

java中的 ==比较都是对象引用比较；

默认equals方法也是比较的对象引用，除非对该方法进行了重写，比如String对每个字符按照字典序比较。

1. 有自己git，https://github.com/zhuzuoyu
2. 前后端分离开发，日志应该如何处理方便定位问题：

配置过滤器，打印请求url、参数信息

1. 同一个项目中可以使用多种持久化技术； 对于spring来说，事务管理器管理事务，在配置的方法中确保使用同一种持久化技术

n的排列组合：

首先把n个整数放到数组链表，传入比较器根据字典序排序；

把第一个元素作为前缀，剩余n-1个元素做为参数递归调用本方法，有n-1的阶乘个；

把第二个元素作为前缀，剩余n-1个元素作为参数递归调用本方法，有n-1的阶乘个；

。

。

。

传入list 参数收集 n-1阶乘个有序列

。

。

。

递归至只有一个元素返回。

收集到n的阶乘个元素列表

【1，2，3，4】

1 【2，3，4】

2 【1，3，4】

3 【1，2，4】

4 【1，2，3】

1 2 【3，4】

1 3 【2，4】

1 4 【2，3】

1 2 3 【4】

1 2 4 【3】