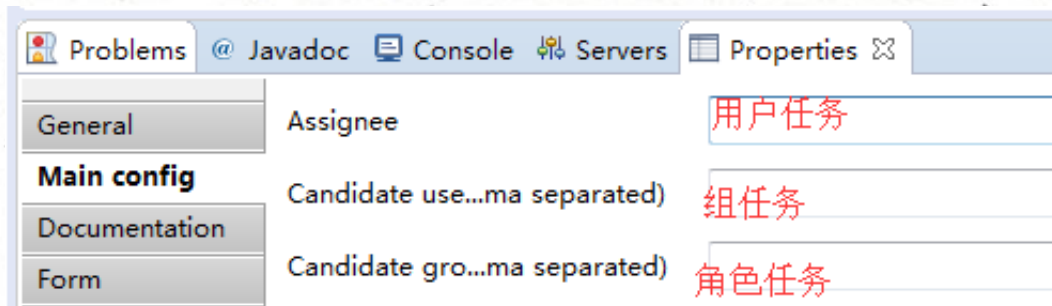


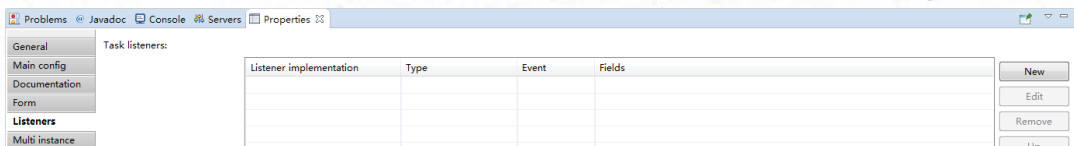
在activiti中，通过如下图可以分别指定用户任务，组任务，或者角色任务。



用户任务：

有三种方式

- 1 : 在Assignee地方直接写 用户的名字 ， 比如 ： 张三 ， 那么这个任务就只有张三可以处理
- 2 : 在Assignee地方填写一个变量 ， 比如\${userID} ， 那么在任务启动，或者设置的时候传递一个 流程变量 userID 即可。
- 3 : 如图点击new会弹出一个窗口，选择一个实现了org.activiti.engine.delegate.TaskListener 接口的实现类，可以在实现类中设置用户的ID.



```
@SuppressWarnings("serial")
```

```
public class TaskListenerImpl implements TaskListener {
```

```
    /**用来指定任务的办理人*/
```

```
    @Override
```

```
    public void notify(DelegateTask delegateTask) {
```

```
        //指定个人任务的办理人，也可以指定组任务的办理人
```

```
        //个人任务：通过类去查询数据库，将下一个任务的办理人查询获取，  
        然后通过setAssignee()的方法指定任务的办理人
```

```
        delegateTask.setAssignee("灭绝师太");
```

```
}  
  
}
```

组任务：

同用户任务一样三种方式。

角色任务：

需要通过流程引擎录入 角色，用户，角色用户关联关系

```
IdentityService identityService =  
processEngine.getIdentityService();//  
    //创建角色  
    identityService.saveGroup(new GroupEntity("总经理"));  
    identityService.saveGroup(new GroupEntity("部门经理"));  
    //创建用户  
    identityService.saveUser(new UserEntity("张三"));  
    identityService.saveUser(new UserEntity("李四"));  
    identityService.saveUser(new UserEntity("王五"));  
    //建立用户和角色的关联关系  
    identityService.createMembership("张三", "部门经理");  
    identityService.createMembership("李四", "部门经理");  
    identityService.createMembership("王五", "总经理");  
    System.out.println("添加组织机构成功");
```