

```
package com.template.improve;

//抽象类, 表示豆浆
public abstract class SoyaMilk {

    //模板方法, make, 模板方法可以做成final, 不让子类去覆盖.
    final void make() {

        select();
        if(customerWantCondiments()) {
            addCondiments();
        }
        soak();
        beat();

    }

    //选材料
    void select() {
        System.out.println("第一步: 选择好的新鲜黄豆 ");
    }

    //添加不同的配料, 抽象方法, 子类具体实现
    abstract void addCondiments();

    //浸泡
    void soak() {
        System.out.println("第三步, 黄豆和配料开始浸泡, 需要3小时 ");
    }

    void beat() {
        System.out.println("第四步: 黄豆和配料放到豆浆机去打碎 ");
    }

    //钩子方法, 决定是否需要添加配料
    boolean customerWantCondiments() {
        return true;
    }
}
```