

```
package com.state.money;
```

```
public abstract class AbstractState implements State {
```

```
    protected static final RuntimeException EXCEPTION = new RuntimeException("操作流程不允许");
```

```
    //抽象类，默认实现了 State 接口的所有方法
```

```
    //该类的所有方法，其子类(具体的状态类)，可以有选择的进行重写
```

```
    @Override
```

```
    public void checkEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void checkFailEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void makePriceEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void acceptOrderEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void notPeopleAcceptEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void payOrderEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
    public void orderFailureEvent(Context context) {  
        throw EXCEPTION;  
    }
```

```
    @Override
```

```
public void feedBackEvent(Context context) {  
    throw EXCEPTION;  
}  
}
```