工厂模式和建造者模式的区别

工厂模式关注的是一个产品整体，生产出的产品应该具有相似的功能和架构，直接定义一个产品的总体特征，然后构建实现工厂直接实例一个整体不可分割的产品。

建造者模式关注的是将产品整体拆开成不同部分，不同的部分可以按照自定义来构建一个产品，灵活性较高。

例子：

如要制造一个超人，如果使用工厂方法模式，直接产生出来的就是一个力大无穷、能够飞翔、内裤外穿的超人；而如果使用建造者模式，则需要组装手、头、脚、躯干等部分，然后再把内裤外穿，才能创造一个超人。

小米只生产红米note7----工厂模式（由实现类决定具体实例）

小米生产各种型号手机，但实际的生产交给不同代工厂代工 ---抽象工厂模式（建立超级工厂，包含多个工厂）

富士康将手机分为各种模块生产组装，构成一台手机 ---建造者模式（将类多维度化）

富士康对各种模块进行采购，相机用sony,电池用三星等 ---桥接模式（将多维度关联单独的类）

富士康建立流水线，不同段负责不同责任，包括装电池，屏幕贴合等，每段检测是否属于自己的责任，属于就操作；否则不操作，然后交给下一段（不纯）或者结束（纯）

---责任链模式

富士康生产中美充电器的转接头，用于调整电压---适配器模式（兼容不同接口）

手机安装android或者Apple os（桥接模式），其中文件系统呈树状结构，并且对文件夹和文件能同一操作---组合模式

组合模式定义了如何将容器对象和叶子对象进行递归组合，使得客户在使用的过程中无须进行区分，可以对他们进行一致的处理，叶子对象和组合对象实现相同的接口

谷歌的原生android功能不符合中国国情，小米在android外面利用开发接口添加额外功能，比如在任何操作后面添加广告弹出---装饰器模式（许向一个现有的对象添加新的功能，同时又不改变其结构，储存原来的实例）

手机翻墙的VPN软件，知道youtube网址，但是不能访问，通过VPN访问youtube,然后返回显示数据—代理模式（代理模式和代理模式很类似，代理模式拥有控制权，并且完全实现被代理类的所有方法，于是要实现被代理类的接口，并且还要注入被代理类的实例）

装饰器模式为了增强功能，而代理模式是为了加以控制。

打开APP时，只要点击图标，就可以开启指定APP，但是操作系统内部执行了启动线程，加载资源，显示出事页面等复杂操作 ---外观模式（隐藏系统复杂性，对外显示可实现功能接口）

打开微信，关注公众号，一旦公众号分心内容，那么所有关注的用户都能第一时间收到信息---观察者模式（类中一对多存在依赖关系，一发生改变，所有依赖的类同时改变）

原型模式：在new创建的代价比较大的时候，用于复制当前实例

过滤器模式：使用不同标准过滤一组对象

享元模型：重复利用已经创建的实例，共享实例

例子工厂中getinstance()函数

命令模式：将命令封装成对象，以便使用不同的请求、日志、队列等来参数化其他对象

命令模式和代理模式区别？

代理模式和被代理的类实现共同接口，对代理模式执行操作，相当于对被代理的类操作

命令模式，中间类不需要继承同一接口，令指定命令绑定指定操作。

解释器模型：

为特定文法进行解释，比如计算器，对加减乘除语法进行解释。

迭代器模式：

不需要知道类的底层实现从而实现对实例的遍历，iterator.

中介者模式：

利用中间平台处理不同类之间的通信。

备忘录模式：

所谓备忘录模式就是在不破坏封装的前提下，保存一个对象的某个状态，以便在适当的时候恢复对象。例子，浏览器中的后退键，保留浏览记录