**الگوی Abstract Factory**

**در الگوی FactoryMethod یک Factory ایجاد می شود و کارش تولید نمونه از کلاس برنامه است و هر کلاینتی نیاز به نمونه ای از کلاس ها داشته باشد از Factory ایجاد شده دریافت و استفاده میکند.**

**اما مشکل این روش در این بود که فقط میتواند یک جنس از محصولات را میتوانست تولید کند.مثلا در مثالی که پیاده سازی شد ما فقط میتوانستیم پیاده سازی های SmsManager را از آن دریافت کنیم که برای برنامه های بزرگ کفایت نمیکند و مجبور به ایجاد تعداد زیادی FactoryMethod میشویم.**

**برای رفع این مشکل AbstractFactory ارائه شد که یک Factory بزرگ و محصولات بیشتر (در خود تعداد زیادی FactoryMethod میتواند داشته باشد.)**

**نام دیگر این الگو کیت میباشد.**

نکات:

1. متد های یجاد نمونه میتوانند پارامتر ورودی داشته باشد.کلاینت بتواند خودش پارامتر ارسال کند و نمونه مورد نظر خودش را ایجاد کند.
2. استفاده از الگوی Prototype به ازای هر نمونه از Product میتوانیم یک ProtoType داشته باشیم.
3. استفاده از interface و abstract class
4. استفاده از singleton بر روی AbstractFactory
5. استفاده از تکنیک IOC برای اتصال سست

مزیا

1. جلوگیری از اتصال محکم بین کد کلاینت و کلاس های Concrete نداشته باشیم.
2. رعیت اصل SRP
3. رعایت اصل Open/Closed

معایب

1. افزودن محصول جدید به سادگی نیست و نیاز به کدنویسی و تغییر زیاد دارد.
2. افزودن پیچیدگی کد به دلیل ارتباط زیاد بین کلاس ها

چه زمانی از این الگو استفاده کنیم

1. زمانی که محصولات مختلفی داریم

الگوهای مرتبط به الگو Abstract Factory

1. Factory Method
2. Prototype
3. Singleton