**الگوی Prototype (ایجادی)**

**هدف : افزایش سرعت ایجاد Object از کلاس ها می باشد. ما در برنامه ممکن است کلاس هایی داشته باشیم که جز کلاس های سنگین باشند و روند ساختن آن به طول بیانجامد و یا کلاس هایی که برای ساختن آن نیاز به یکسری config باشد.**

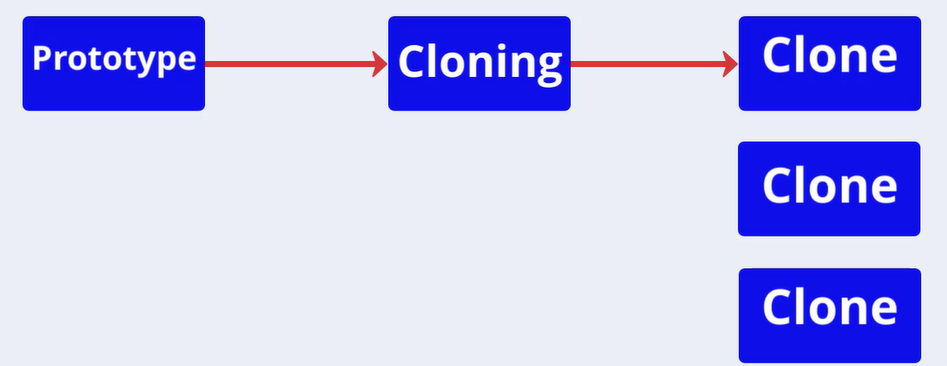
**روال کار این الگو به این صورت است که اولین Objectیی را که میسازیم را در حافظه ذخیره میکند و در هربار که نیاز به ایجاد Object جدیدی باشد از همان Objectدر حافظه استفاده میکند.**

****

**اولین Object ساخته شده Prototype میگوییم و به عمل کپی سازی از prototype ، cloning میگوییم و به object انتهایی که از کپی ساخته شده Clone میگوییم.**

**مزایا :**

1. **سرعت نمونه سازی به شدت افزایش می یابد**
2. **اگر یک آبجکت داشته باشیم بدون اینکه بدانیم نام کلاس آن چیست میتوانیم آبجکت جدید بسازیم.**
3. **اضافه و حذف Prototype در زمان اجرا**
4. **استفاده از Prototype به جای کلاس در زمان اجرا**
5. **میتوانیم چندین clone بسازیم که هیچ کدام با هم ارتباطی ندارند.**

****

1. **جلوگیری از تکرار کدها**

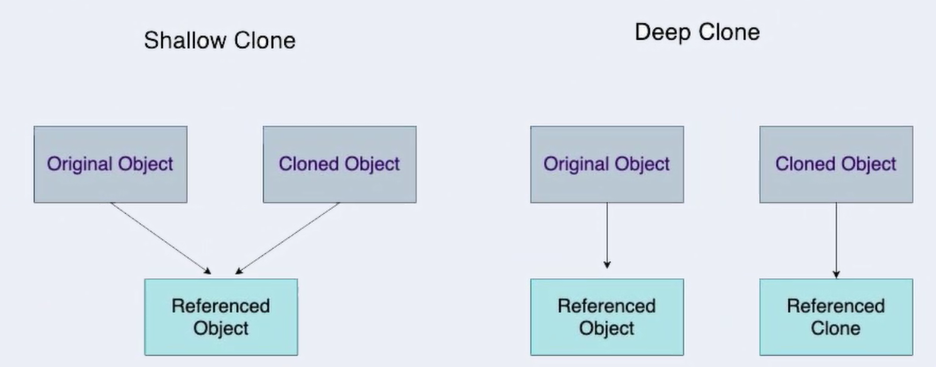
**معایب**

1. **پیاده سازی پیچیده منطق clone**
2. **مشکل Dispose اشیا**

**بررسی تفاوت Shallow Copy و Deep Copy**

**وقتی ما از آبجکت یک clone میگیریم این کپی به دو صورت انجام میشود :**

1. **کپی کم عمق Shallow Copy**
2. **کپی عمیقDeep Copy**

****

**نکات :**

1. **به ازای یک کلاس میتوانیم چندین پیاده سازی داشته باشیم**
2. **عملیات clone پیاده سازی سختی دارد.**
3. **کلاینت ها شاید نیاز داشته باشند که نسخه های کپی شده را شخصی کنند.**
4. **اگر تعداد Prototype ها زیاد بود میتوانیم از PrototypeManager استفاده کنیم/**

**چه زمانی از این الگو استفاده کنیم**

1. **افزایش سرعت نموه سازی از کلاس های سنگین**
2. **کاهش تعداد کلاس های استفاده شده در سیستم**
3. **پنهان کردن برخی از کلاس ها از دید کاربر**