**الگوی Memento**

**در برخی موراد نیاز داریم اشیا را ذخیره و بعد از مدتی آن ها را بازیابی کنیم. اطلاعات موجود یک شی را ، در یک حافظه جانبی ذخیره کنیم، در ادامه بتوانیم در مواقع مورد نیاز آن شی را به همان صورت بازیابی کنیم.(همانند Redo و Undo در الگوی Command )**

**یا در بازی های کامپیوتری وضعیت اشیا را در یک بازی کامپیوتری ذخیره کنیم.**

**نکات:**

1. **کسی بجز originator نباید به اطلاعات Memento دسترسی داشته باشد:**

* **Nested Class**
* **تعریف Memento به صورت internal تعریف شود.(Memento و originator در یک Assembly تعریف شود که امکان دسترسی مستقیم به Memento امکان پذیر نباشد.)**
* **رمز نگاری اطلاعات در Originator**

1. **Originator هم میتواند مسئول ذخیره اطلاعات در یک فضای External باشد.**

مزایا:

1. به راحتی میتوانیم یک Snapshot از اشیا ایجاد کنیم و بعدا بازیابی کنیم.
2. با وجود کلاس caretaker کار ساده تر شده است.

معایب:

1. **اگر** Snapshot را در RAM ذخیره کنید بعد از مدتی حافظه زیادی اشغال می شود.

**چه زمانی از این الگو استفاده می شود:**

1. **فرض کنید در حال طراحی یک اپلیکیشن هستید و میخواهید وضعیت اپلیکیشن را ذخیره کنید که بعد از کرش به حالت قبل برگردد.**
2. **زمانی که میخواهید برای سیستم Check Point تعریف کنید.**
3. **زمانی که میخواهید قابلیت save برای بازی ها پیاده سازی کنید.**

**الگو های مرتبط**

* **الگوی Command**