

زبان برنامه‌نویسی سی‌شارپ (#C) به دلیل قابلیت‌های فراوان و انعطاف‌پذیری که ارائه می‌دهد، یکی از زبان‌های محبوب برای توسعه‌دهندگان محسوب می‌شود. یکی از قابلیت‌های ویژه در این زبان، کلاس‌های Sealed است. کلاس Sealed به توسعه‌دهندگان این امکان را می‌دهد که از ارث‌بری (inheritance) ناخواسته جلوگیری کرده و کنترل بیشتری روی ساختار و رفتار کدهای خود داشته باشند. در این مقاله، به‌صورت جامع و کاربردی به مفهوم کلاس Sealed در سی‌شارپ پرداخته و نحوه استفاده از آن را بررسی خواهیم کرد.

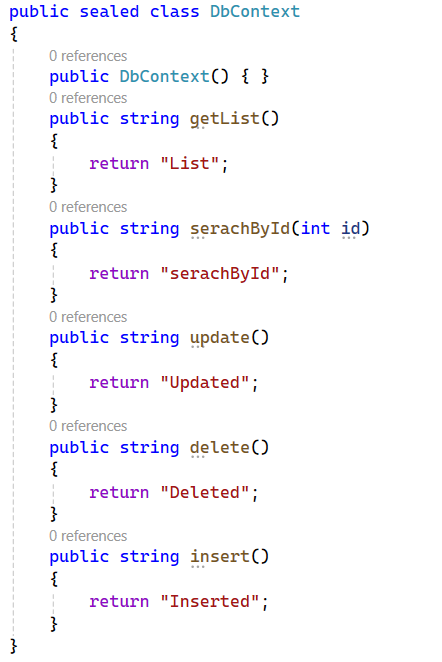
## مفهوم کلاس Sealed در سی‌شارپ

کلمه کلیدی sealed به معنی "مهر و موم شده" یا "قفل شده" است. هنگامی که کلاسی را با این کلمه کلیدی تعریف می‌کنیم، آن کلاس دیگر نمی‌تواند به‌عنوان پایه یا پدر برای کلاس‌های دیگر مورد استفاده قرار گیرد. به عبارت دیگر، هیچ کلاسی قادر به ارث‌بری از کلاس sealed نخواهد بود.

این ویژگی زمانی کاربرد دارد که می‌خواهیم از تغییرات ناخواسته در رفتار یا ساختار یک کلاس جلوگیری کنیم. در بسیاری از موارد، زمانی که یک کلاس را به‌صورت کامل طراحی کرده‌ایم و مطمئن هستیم که نیازی به گسترش آن نداریم، از sealed استفاده می‌کنیم تا این کلاس غیرقابل ارث‌بری باشد.

## نحوه استفاده از کلاس Sealed

برای استفاده از کلاس sealed کافیست در هنگام تعریف کلاس، کلمه کلیدی sealed را قبل از نام کلاس بنویسیم. به مثال زیر دقت کنید:



در این مثال، کلاس Vehicle به‌عنوان یک کلاس sealed تعریف شده است. بنابراین، هیچ کلاسی نمی‌تواند از Vehicle ارث‌بری کند. این کلاس در واقع به‌عنوان یک کلاس نهایی در نظر گرفته می‌شود که نمی‌توان آن را گسترش داد.

## کاربردهای کلاس Sealed

کلاس‌های sealed در موارد مختلفی می‌توانند مفید باشند. در اینجا به برخی از کاربردهای مهم اشاره می‌کنیم:

### ۱. جلوگیری از ارث‌بری ناخواسته

گاهی اوقات در پروژه‌های بزرگ ممکن است نخواهیم کلاس‌هایی که طراحی کرده‌ایم، توسط توسعه‌دهندگان دیگر به اشتباه یا ناخواسته ارث‌بری شوند. با استفاده از sealed می‌توانیم جلوی این اتفاق را بگیریم و مطمئن شویم که کلاس مورد نظر تغییر نخواهد کرد.

### ۲. بهبود کارایی برنامه

استفاده از کلاس‌های sealed می‌تواند منجر به بهینه‌سازی‌هایی توسط کامپایلر شود. وقتی کلاسی sealed است، کامپایلر مطمئن است که هیچ کلاس دیگری از آن ارث‌بری نخواهد کرد. این اطمینان به کامپایلر اجازه می‌دهد تا برخی بهینه‌سازی‌ها را انجام داده و سرعت اجرای برنامه را افزایش دهد.

### ۳. کنترل دقیق‌تر روی کلاس‌ها

با sealed کردن یک کلاس، می‌توانیم از تغییرات ناخواسته در رفتار آن جلوگیری کنیم. این قابلیت به ویژه در مواقعی که کلاس‌هایی داریم که قوانین یا رفتارهای خاصی را پیاده‌سازی کرده‌اند، بسیار مفید است.

## مزایای استفاده از کلاس Sealed

استفاده از کلاس‌های sealed دارای مزایای متعددی است که در اینجا به چند مورد اشاره می‌کنیم:

حفظ کنترل بر روی کلاس‌ها: با جلوگیری از ارث‌بری، کنترل بیشتری روی ساختار و رفتار کلاس‌ها خواهیم داشت.  
افزایش امنیت کد: استفاده از sealed از تغییرات ناخواسته و خطرناک در کلاس‌ها جلوگیری می‌کند و از خراب شدن کدهای حساس جلوگیری می‌کند.  
بهبود کارایی: کلاس‌های sealed می‌توانند بهینه‌سازی‌هایی توسط کامپایلر داشته باشند که منجر به بهبود عملکرد و کاهش سربار محاسباتی می‌شود.

## نکات مهم در استفاده از کلاس Sealed

کلاس‌های sealed نمی‌توانند به‌عنوان پایه برای سایر کلاس‌ها مورد استفاده قرار گیرند. یعنی امکان ارث‌بری از آن‌ها وجود ندارد.  
کلاس‌های sealed همچنان می‌توانند ویژگی‌ها و متدهای خود را داشته باشند و از تمام قابلیت‌های دیگر کلاس‌ها بهره ببرند.  
کلاس‌های sealed نمی‌توانند از کلاس‌های دیگر ارث‌بری کنند؛ یعنی همواره باید به‌صورت مستقل عمل کنند.

## سؤالات متداول

آیا می‌توان از کلاس sealed در هر پروژه‌ای استفاده کرد؟  
بله، استفاده از کلاس sealed در هر پروژه‌ای مجاز است. اما معمولاً در مواردی کاربرد دارد که نیازی به ارث‌بری از یک کلاس نداشته باشیم و بخواهیم ساختار کلاس‌ها ثابت باقی بماند.

آیا کلاسی که sealed است، می‌تواند از یک کلاس دیگر ارث‌بری کند؟  
خیر، کلاس‌های sealed نمی‌توانند از کلاس‌های دیگر ارث‌بری کنند. این کلاس‌ها به‌عنوان کلاس‌های نهایی در نظر گرفته می‌شوند.

آیا استفاده از کلاس‌های sealed برای همه کلاس‌ها توصیه می‌شود؟  
خیر، استفاده از sealed برای همه کلاس‌ها توصیه نمی‌شود. معمولاً زمانی از sealed استفاده می‌شود که از نیاز به ارث‌بری جلوگیری شود و قصد داشته باشیم کنترل بیشتری روی کلاس‌ها داشته باشیم.

## نتیجه‌گیری

کلاس‌های sealed در سی‌شارپ به برنامه‌نویسان این امکان را می‌دهند که جلوی ارث‌بری ناخواسته را بگیرند و کنترل بیشتری بر ساختار و رفتار کدهای خود داشته باشند. همچنین این کلاس‌ها با بهینه‌سازی‌هایی که انجام می‌دهند، می‌توانند به بهبود کارایی برنامه کمک کنند. با درک صحیح و استفاده مناسب از sealed می‌توان برنامه‌های خود را بهتر و بهینه‌تر طراحی کرد.

برای یادگیری عمیق‌تر و استفاده حرفه‌ای از مفاهیم سی‌شارپ، می‌توانید در دوره‌های آموزشی ما شرکت کنید و با مثال‌های کاربردی بیشتر، این مفاهیم را در پروژه‌های خود پیاده‌سازی کنید.

در این مقاله، به بررسی کلاس sealed در #C پرداختیم. کلاس sealed به ما امکان می‌دهد تا از ارث‌بری ناخواسته جلوگیری کنیم و کنترل بیشتری روی رفتار کلاس‌هایی که از آنها ایجاد می‌شوند، داشته باشیم. همچنین، استفاده از کلاس‌های sealed می‌تواند به بهبود کارایی برنامه کمک کند.