## 

## Cả hai đều là hằng số

## Hằng số

const phải được khởi tạo tại thời điểm khai báo, chỉ có thể được khởi tạo trên hàm tạo (và do đó có giá trị khác nhau tùy thuộc vào hàm tạo được sử dụng).

const: Không thể thay đổi ở bất cứ đâu.

'const là ngầm static

Một const là hằng số thời gian biên dịch trong khi chỉ đọc cho phép một giá trị được tính toán tại thời gian chạy và được đặt trong hàm tạo hoặc trình khởi tạo trường. Vì vậy, một 'const' luôn luôn không đổi nhưng 'chỉ đọc' chỉ đọc khi được gán.

**Hằng :**Bằng cách bất chấp tĩnh. Giá trị không thể thay đổi từ bất cứ đâu (Ctor, Function, runtime v.v.

constgiá trị là giống nhau cho tất cả các đối tượng (và phải được khởi tạo bằng biểu thức bằng chữ),

* Các hằng số là tĩnh theo mặc định
* Chúng phải có giá trị tại thời gian biên dịch (bạn có thể có ví dụ: 3.14 \* 2, nhưng không thể gọi các phương thức)
* Có thể được khai báo trong các hàm
* Được sao chép vào mọi hội đồng sử dụng chúng (mọi hội đồng đều có một bản sao giá trị cục bộ)
* Có thể được sử dụng trong các thuộc tính
* Chúng không thể được khai báo là static(chúng hoàn toàn tĩnh)
* Giá trị của hằng được đánh giá tại thời điểm biên dịch
* hằng số chỉ được khởi tạo lúc khai báo

## Các trường chỉ đọc

* Phải đặt giá trị, bằng cách thoát hàm tạo
* Được đánh giá khi cá thể được tạo
* Chúng có thể là mức cá thể hoặc mức tĩnh
* Giá trị được đánh giá trong thời gian chạy
* chỉ đọc có thể được khởi tạo trong khai báo hoặc bằng mã trong hàm tạo

## Các trường chỉ đọc tĩnh

* Được đánh giá khi thực thi mã chạm tham chiếu lớp (khi phiên bản mới được tạo hoặc phương thức tĩnh được thực thi)
* Phải có giá trị được đánh giá theo thời gian xây dựng tĩnh được thực hiện
* Không nên đặt ThreadStaticAttribution trên các hàm này (các hàm tạo tĩnh sẽ chỉ được thực thi trong một luồng và sẽ đặt giá trị cho luồng của nó; tất cả các luồng khác sẽ có giá trị này chưa được khởi tạo)

Chỉ cần thêm, ReadOnly cho các loại tham chiếu chỉ làm cho tham chiếu chỉ đọc không phải là các giá trị.

readonly: Giá trị này chỉ có thể được thay đổi trong hàm tạo. Không thể thay đổi trong các chức năng bình thường.

**Chỉ đọc** : Giá trị có thể được thay đổi thông qua Ctor khi chạy. Nhưng không thông qua chức năng thành viên

Một readonlythành viên giống như một hằng số ở chỗ nó đại diện cho một giá trị không thay đổi. Sự khác biệt là một readonlythành viên có thể được khởi tạo trong thời gian chạy, trong một hàm tạo, cũng như có thể được khởi tạo khi chúng được khai báo.

readonlycó thể khác nhau cho mỗi lần khởi tạo ...

**Tóm tắt** :

* Giá trị thuộc tính const của bạn được đặt tại thời gian biên dịch và không thể thay đổi khi chạy
* Const không thể được đánh dấu là tĩnh - từ khóa biểu thị chúng là tĩnh, không giống như các trường chỉ đọc có thể.
* Không thể là bất cứ thứ gì ngoại trừ các loại giá trị (nguyên thủy)
* Từ khóa chỉ đọc đánh dấu trường là không thể thay đổi. Tuy nhiên, thuộc tính có thể được thay đổi bên trong hàm tạo của lớp
* Từ khóa chỉ đọc cũng có thể được kết hợp với tĩnh để làm cho nó hoạt động theo cách tương tự như một const (ít nhất trên bề mặt). Có một sự khác biệt rõ rệt khi bạn nhìn vào IL giữa hai
* Các trường const được đánh dấu là "nghĩa đen" trong IL trong khi chỉ đọc là "initonly"