

1. Lập luận Stream cơ bản

Lập Stream là một lập luận, có mặt trong không gian System.IO. Nó là lập luận cha của tất cả các lập luận con khác. Nói cách khác, tất cả các lập luận con có sẵn hoặc là lập luận cha Stream, hoặc là lập luận con có sẵn phát triển từ nó nên các hiệu ứng riêng. Hình T3.1 cho thấy toàn cảnh.

Các thành viên của luồng stream

Thuộc tính

Bảng sau liệt kê các luồng con của Stream bao gồm 5 luồng con và ba thuộc tính chung:

Thuộc tính	Mô tả
CanRead	Các dòng dữ liệu mà bạn kiểm tra luồng hiện hành có thể đọc nội dung của nó hay không. Nhìn chung, thuộc tính này sẽ trả về true khi thực hiện các hành động đọc luồng. Giá trị trả về: true hoặc false. Nếu giá trị trả về là false thì còn kèm thêm một thông điệp NotSupportedException.
CanSeek	Mức độ luồng có thể truy cập vào một vị trí nào đó trong luồng. Thuộc tính CanSeek kiểm tra luồng hiện hành có thể thực hiện dò tìm vị trí con truy cập luồng hay không. Giá trị trả về: true hoặc false. Nếu giá trị trả về là false thì còn kèm thêm một thông điệp NotSupportedException.
CanWrite	Thuộc tính này thường kiểm tra trước khi thực hiện các hoạt động ghi vào luồng hiện hành. Giá trị trả về: true hoặc false. Nếu giá trị trả về là false thì còn kèm thêm một thông điệp NotSupportedException.

Ngoài ra, còn hai thuộc tính khác:

Thuộc tính	Mô tả
Length	Trả về một giá trị kiểu long, cho biết chiều dài của luồng, tính theo byte. Thuộc tính này thường dùng để xác định chiều dài luồng, hoặc vị trí cuối của một mảng byte, hay một biến mảng một chiều lưu trữ luồng.
Position	Dùng để tìm kiếm vị trí con truy cập của luồng. Dùng thuộc tính này, luồng phải thực hiện dò tìm (thuộc tính CanSeek phải trả về giá trị true)

Phương thức

Phương thức Seek(). Phương thức dùng để tìm kiếm vị trí con trỏ trong luồng theo đường truy cập ngẫu nhiên. Seek() cần cung cấp hai tham số. Thứ nhất, một giá trị long cho ra vị trí tính từ “mức gì”. Thứ hai, giá trị SeekOrigin chính là “mức gì” mà tham số thứ nhất tham chiếu. SeekOrigin có ba giá trị gồm SeekOrigin.Begin, SeekOrigin.Current và SeekOrigin.End. (Tham khảo bài ví dụ).

Các phương thức đọc và ghi luồng được chia thành hai tập riêng dành cho truy xuất luồng theo đường ngẫu nhiên hay bất kỳ.

Phương thức đọc, ghi ngẫu nhiên

Phương thức	Mô tả
Read() và ReadByte()	Đọc số lượng n byte dữ liệu từ luồng theo đường ngẫu nhiên. Phương thức Read() cần truyền số byte cần đọc. Trong số byte này, nó lấy con trỏ hiện tại và vị trí byte cần đọc tiếp theo. ReadByte() chỉ đọc một byte, xong nó lấy con trỏ lên một vị trí. Chú ý, phương thức Read() trả về 0 nếu nó đã hết dữ liệu, trong khi ReadByte() sẽ trả về -1.
Write() và WriteByte()	Viết số lượng n byte dữ liệu vào luồng theo đường ngẫu nhiên. Phương thức Write() ghi một chuỗi byte vào luồng và lấy con trỏ hiện tại và vị trí byte cần ghi tiếp theo. WriteByte() chỉ ghi vào luồng một byte, ghi xong nó lấy con trỏ lên một vị trí.



Phương thức đọc, ghi bất kỳ

Phương thức	Mô tả
BeginRead() và BeginWrite()	Đọc số lượng n byte dữ liệu từ luồng theo đường bất kỳ. Có hai phương thức sử dụng 5 tham số. Thứ nhất, một byte cần đọc/ghi. Thứ hai, một số nguyên chỉ vị trí bắt đầu đọc/ghi. Thứ ba, một số nguyên chỉ số lượng byte cần đọc/ghi. Thứ tư, là một hàm tùy chọn, delegate ¹ AsyncCallback, nó gọi khi hoạt động đọc/ghi hoàn thành. Thứ năm, là một tham số do người dùng cung cấp như một phân biệt hành động khác nhau của các yêu cầu khác nhau. Có hai phương thức sử dụng giao tiếp IAsyncResult để hiển thị tình trạng của hoạt động bất kỳ.

¹ Delegate: thường được dịch là “y nhai hàm”, nhưng chuyên môn về nó là delegate. Nó tương tự như con trỏ trong C++. Mục đích sử dụng nó là phù hợp hóa giá trị truy cập giữa các ngôn ngữ và phương thức.

EndRead() và EndWrite()	c s d ng n u mu n ghi n i dung vào m t lu ng theo d ng ng b . Ph ng th c Write() ghi m t chùm byte vào lu ng và y con tr n v trí byte ã ghi c sau cùng. WriteByte() ch ghi vào lu ng t ng byte, ghi xong nó c ng y con tr lên m t v trí.
-------------------------------	--

Các ph ng th c qu n lý lu ng

Ph ng th c	Mô t
Flush() 	Xóa t t c n i dung b m sau khi ã chuy n toàn b n i dung t i ích c a i t ng lu ng Stream.
Close() 	Gi i phóng tài nguyên, óng các t p tin và socket liên quan t i lu ng. N u nó c g i thì nó t ng thi hành Flush() tr c
SetLength	Thi t l p chi u dài c n thi t c a lu ng hi n hành