3장. Content

1. Content 프로퍼티?

* 내용물
* 문자열이나 비트맵이미지 ,그림 , 버튼 ,스크롤 바 등을 지정할 수 있고  
  WPF에서 지원하는 50여개의 컨트롤을 이용할수 있다.
* 클라이언트 영역에 필요한 객체를 할당하여 사용할 수 있는 Window클래스의

프로퍼티

* Object 형으로 Window를 제외한 어떠한 객체라도 할당할 수 있다. (Window객체는 반드시 트리의 Root가 되어야함. 자식이 될 수 없음)
* 하나의 Content 프로퍼티에는 오직 하나의 객체만 할당 가능..
* Content 프로퍼티 동작에 관한 모든 객체는 UIElement를 상속 받은 것 : OnRender로 출력..

그렇지 않은 것 : ToString으로 출력..

* UIElement를 직접 상속받은 유일한 클래스는 FrameworkElement
* 모든 Element는 FrameworkElement를 상속받는다.

1. 텍스트 문자열 할당  
   -Content 프로퍼티에 string객체를 대입함으로써 클라이언트 영역에 손쉽게 문자를 출력할 수 있다..  
   - Font 와 관련된 각종 프로퍼티를 이용해 글자의 모양을 다양하게 바꿀수 있다.  
   - 모든 객체는 ToString 매서드를 통해 문자열을 반환한다. Element를 제외한 객체는 Content에 대입을 하면 그에 대한 ToString 값을 출력해 주지만. 예외적으로 array(예: new int[5]; )같은 경우에는 Int32[] Array 의 형태로 출력되는걸 볼 수 있다.

Run : 서식 있는 텍스트 또는 서식 없는 텍스트의 개별 섹션을 나타냅니다.

Run 및 [LineBreak](http://msdn.microsoft.com/ko-kr/library/system.windows.documents.linebreak(v=vs.95).aspx) 개체는 텍스트 블록 콘텐츠의 기본 단위입니다. 텍스트 블록의

텍스트에 서식을 지정하려면 다양한 Run 개체에 서식을 적용할 수 있습니다.

**※**ContentElement 와 FrameworkContentElement = UIElement와 FrameworkElement 차이

ContentElement 와 FrameworkContentElement = UIElement와 FrameworkElement와 유사

그러나 OnRender 메소드가 없어 단독적으로 그려지지 않는다.

다른 Element에 일부분이 된다...

ContentControl 은 Content 프로퍼티가 있어 독자적으로 그려짐..

**※Element?**

WPF의 최소 작업단위로 표현 할 수 있다.

WPF에서 프로그램 가능한 작업의 단위는 Element 우리말로 직역하자면 원소, 요소(선, 도형, 점...)

기존의 사용하던 컨트롤(버튼,리스트 박스 등..)과 비슷한 개념이지만 WPF 에서는 이보다 더 큰 범위의 기능을 제공한다.

Element 는 shape + control 이라고 표현 할 수 있다.

WPF의 Element들은 자체적으로 Hit Test를 지원하기 때문에 마우스의 모든 이벤트를 사용 좀더 정확히 구현할수 있다.

윈도우 상에서 그려진 이미지가 자동으로 상태관리가 된다.

하나의 Element는 그 모양에 관한 (벡터로 이루어진 모든 기하학도형..) 정보들과 이벤트 작업속성 등을 담을 수 있게 구성되어 있다 . 한마디로 말해 컨트롤을 원하는 모양으로 꾸밀 수 있는 기능이 추가 되었다고 보면 이해가 쉬울 것**. )**

1. 이미지 할당.

- 이미지를 출력하기위한 순서: Uri 객체 생성 🡪 위치지정 🡪 BitmapImage객체생성🡪

Image객체에 등록 🡪 Content에 대입. 순으로 진행

* 디렉토리의 이미지 로딩하기 : Uri객체 생성자에 FullPath를 인자로 넘긴다.
* Image 객체의 각종 프로퍼티를 이용해 이미지조정
* Foreground 프로퍼티는 이미지사용시에는 아무런 효과가 없다.

1. 텍스트에 각각 다른 속성 주기.  
   - TextBlock 객체의 Inlines.Add 매서드를 사용해 문장과 스타일을 각각 추가한다
2. OnRender를 이용한 그리기.

* OnRender 는 UiElement를 상속받은 모든 Element에 존재하며 .  
  그리기에 관한 정보를 담고있다.
* OnRender 를 오버라이딩 하면 DrawingContext 객체를 매개변수로 받는다  
  이 객체에 미리 정의된 매서드를 이용해 각종 그리기에 대한 속성을 저장한다.

Memdc와 비슷한 역할…

* 최종적으로 Content에 할당시에 OnRender를 호출하여 저장된 그림들을 그려 주 게 된다..