# Bài 7 – Welcome to WPF

## 1. eXtensible Application Markup Language

Ngôn ngữ đánh dấu ứng dụng.

Ví dụ: <StackPanel> <Button Content="Click Me"/> </StackPanel>

## 2. Thuộc tính

* Cú pháp thuộc tính: đặt tên thuộc tính đối tượng đang được đặt, theo sau là toán tử gán (=).
* Giá trị của một thuộc tính luôn được chỉ định là một chuỗi được chứa trong dấu ngoặc kép.

Ví dụ: <Button Background="Blue" Foreground="Red" Content="This is a button"/>

## 3. Property Element Syntax

Đối với vài thuộc tính của Object Element, khai báo cúpháp thuộc tính là không thể bởi vì giá trị của thuộc tính không nằm trong phạm vi "chuỗi giá trị“; thay vào đó ta phải khai báo Property Element Syntax

## 4. Text Content

Một số ít các element có khả năng chứa text như là content của nó , để có được khả năng này cho các element ta phải khai báo đúng ,Class của Object Element tương ứng phải có khai báo content property có kiểu string hoặc object

Ví dụ: <Button>Hello</Button>

## 5. Events (Attribute Syntax)

* Events được dùng cho các thành phần như là một sự kiện. Tên của Events là tên của sự kiện.
* Trong WPF, hiện thực một sự kiện của XAML: tên của sự kiện là tên của phương thức xử lý.

## 6. Root Element và Namespace

* Mỗi tập tin XAML chỉ có 1 root element như : Window , Application , Page , Resource Dictionary
* Root Element chứa các thuộc tính xmlns & xmlns:x ,các thuộc tính này chỉ định bộ xử lý XAML, XAML namespace chứa các định nghĩa kiểu cho việc sao lưu các kiểu đánh dấu như Element.
* Thuộc tính xmlns chỉ định namespace XAML mặc định, trong các namespace XAML mặc định các object element có thể được xác định mà không cần đến 1 tiền tố.

*Note:* các thuộc tính xmlns chỉ cần khai báo trên các root element của mỗi tập tin XAML.

## 7. Các thành phần của WPF

### Panel

* Sắp đặt các thành phần khác nhau trên giao diện.
* Mỗi panel có thể chứa các thành phần con (control panel,…).
* Panel khác nhau cho phép sắp xếp thành phần con theo những cách khác nhau.

### Các Panel thông dụng

#### StackPanel

* Bố trí các thành phần con nằm trong nó bằng cách sắp xếp chúng theo thứ tự trước sau
* Các phần tử sẽ xuất hiện theo thứ tự mà chúng được khai báo trong file XAML theo chiều dọc (ngầm định) hoặc theo chiều ngang (Orientation = "Horizontal”)

#### WarpPanel

Cho phép sắp xếp các phần tử từ trái sang phải. Khi một dòng phần tử đã điền đầy khoảng không gian cho phép theo chiều ngang, WrapPanel sẽ cuốn phần tử bị tràn khởi giới hạn kích quy định của nó xuống đầu dòng tiếp theo (tương tự như việc cuốn text).

#### DockPane

Cho phép các phần tử bám lên các cạnh của panel DockPanel bao chứa chúng. Nếu như có nhiều phần tử cùng bám về một cạnh, chúng sẽ tuân theo thứ tự mà chúng được khai báo trong file XAML

#### Canvas

* Sử dụng phương thức sắp xếp các phần tử UI theo vị trí tuyệt đối bằng cách đặt thuộc tính Top (đỉnh) và Left (bên trái) của chúng.
* Thêm vào đó, thay vì đặt thuộc tính Top, Left, ta có thể đặt thuộc tính Bottom (đáy), Right (bên phải). Nếu đặt đồng thời thuộc tính Left và Right, thuộc tính Right sẽ bị bỏ qua. Tương tự thuộc tính Top sẽ được ưu tiên hơn thuộc tính Bottom.
* Phần tử UI sẽ không thay đổi kích thước để thỏa mãn 2 thuộc tính trên cùng một lúc. Các phần tử được khai báo sớm hơn trong file XAML sẽ có thể bị che khuất phía dưới các phần tử được khai báo muộn hơn nếu vị trí của chúng xếp chồng lên nhau.

#### Grid

* Cho phép sắp đặt các thành phần con lên một lưới toạ độ
* Grid sẽ tự động chia đều các dòng và cột (dựa trên kích thước của phần nội dung)
* Sử dụng dấu sao (\*) để phân định kích thước theo tỉ lệ hoặc phân định giá trị tuyệt đối về chiều cao hoặc chiều rộng cho hàng và cột.

### Control:

* Label: Nhãn
* TextBox: Hộp soạn thảo
* Button: Nút bấm
* CheckBox: Hộp chọn
* RadioButton: Hộp chọn radio (chỉ được phép chọn 1 mục trong mỗi nhóm)
* ListBox: Hộp danh sách
* ComboBox: Hộp danh sách thả xuống