

XAML - Extensible Application Markup Language

Introduction

XAML (eXtensible Application Markup Language) est un langage de balisage basé sur XML utilisé principalement pour décrire des interfaces utilisateur dans les applications Microsoft, notamment pour .NET MAUI, WPF, UWP et autres frameworks Microsoft.

Structure de base

XAML suit la structure XML :

XML et Balises

Dans XML, "M" signifie Markup (balisage) :

- Une balise s'ouvre et se ferme
 - Ouvrante: <recette>
 - Fermante: </recette>
- Une balise peut avoir :
 - Des attributs
 - Un corps
- Si une balise ne contient pas de "corps", on peut la fermer avec un raccourci :
 - <four option="airChaud" />

Exemple XML 🍪

```
<?xml version="1.0" encoding="utf-8" ?>
<recette nom="brownies">
    <ingredients>
        <ingredient quantite="200" allergene="true">noix de pécan
        </ingredient>
        <ingredient quantite="100">farine
        </ingredient>
        <ingredient quantite="150">sucré</ingredient>
        <!-- TODO: ajouter le beurre et le chocolat -->
    </ingredients>

    <cuisson>
        <four option="airChaud"/>
        <temps unite="minute">15</temps>
    </cuisson>
</recette>
```

Hiérarchie XAML : "les poupées russes" 🎃

XAML organise les éléments de manière hiérarchique, comme des poupées russes imbriquées les unes dans les autres. Par exemple :

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="HelloMaui1.Layout"
    Title="Layout">
    <VerticalStackLayout>
        <FlexLayout>
            <BoxView Color="Green"/>
            <Label Text="RIGHT EYE"/>
        </FlexLayout>
    </VerticalStackLayout>
</ContentPage>
```

Attributs

Les attributs permettent de configurer les propriétés des éléments.

Exemple

```
<ingredients>
    <ingredient quantite="200" allergene="true">noix de pécan
    </ingredient>
    <ingredient quantite="100">farine</ingredient>
    <ingredient quantite="150">sucré</ingredient>
    <!-- TODO: ajouter le beurre et le chocolat -->
</ingredients>
```

Dans cet exemple, `quantite` et `allergene` sont des attributs de l'élément `ingredient`.

Corps des balises

Le corps d'une balise est le contenu entre la balise ouvrante et fermante.

Exemple

Dans l'exemple de la [recette de brownies](#) :

- `<recette>` a un corps composé de 2 éléments : `<ingredients>` et `<cuisson>`
- Pour `<ingredients>`, le corps contient plusieurs éléments `<ingredient>`
- Pour la balise `<temps>`, le corps est "15"

XAML et XML

XAML est du XML avec des balises définies par les namespaces XML.

Namespaces

Les namespaces définissent les ensembles d'éléments disponibles :

```
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="HelloMaui1.Crud"
    xmlns:viewModels="clr-namespace:HelloMaui1.ViewModels"
    Title="Crud">

</ContentPage>
```

Exemples XAML courants

Description	XAML	Rendu
Afficher un bouton "quitter"	<Button Text="quitter" />	[Bouton avec texte "quitter"]
Afficher une grille 2x2	<Grid WidthRequest="100" HeightRequest="100"> <BoxView Grid.Row="0" Grid.Column="0" /> <BoxView Grid.Row="0" Grid.Column="1" Color="Red" /> <BoxView Grid.Row="1" Grid.Column="0" Color="Blue" /> <BoxView Grid.Row="1" Grid.Column="1" /> </Grid>	[Grille 2x2 avec cases colorées]
Afficher un champ texte	<Entry Text="Votre nom" />	[Champ de saisie avec texte par défaut]

Points importants à retenir

1. XAML suit la structure XML avec des balises ouvrantes et fermantes
2. Les balises peuvent avoir des attributs qui définissent leurs propriétés
3. Les éléments XAML s'organisent de manière hiérarchique
4. Les namespaces définissent les ensembles d'éléments disponibles
5. XAML est utilisé principalement pour décrire des interfaces utilisateur dans les applications Microsoft

Exercices pratiques

1. Combien d'attributs y a-t-il dans l'exemple d'ingrédients ?
2. Combien d'éléments contient la balise `<ingredients>` ?
3. Quel est le corps de la balise `<temps>` ?
4. Créer une interface simple avec un bouton et une étiquette.