

XPath

XML Path

Objectifs

- ▶ • Comprendre le rôle de XPath
- ▶
 - Apprendre à naviguer dans un document XML en utilisant les expressions XPath

Présentation de XPath

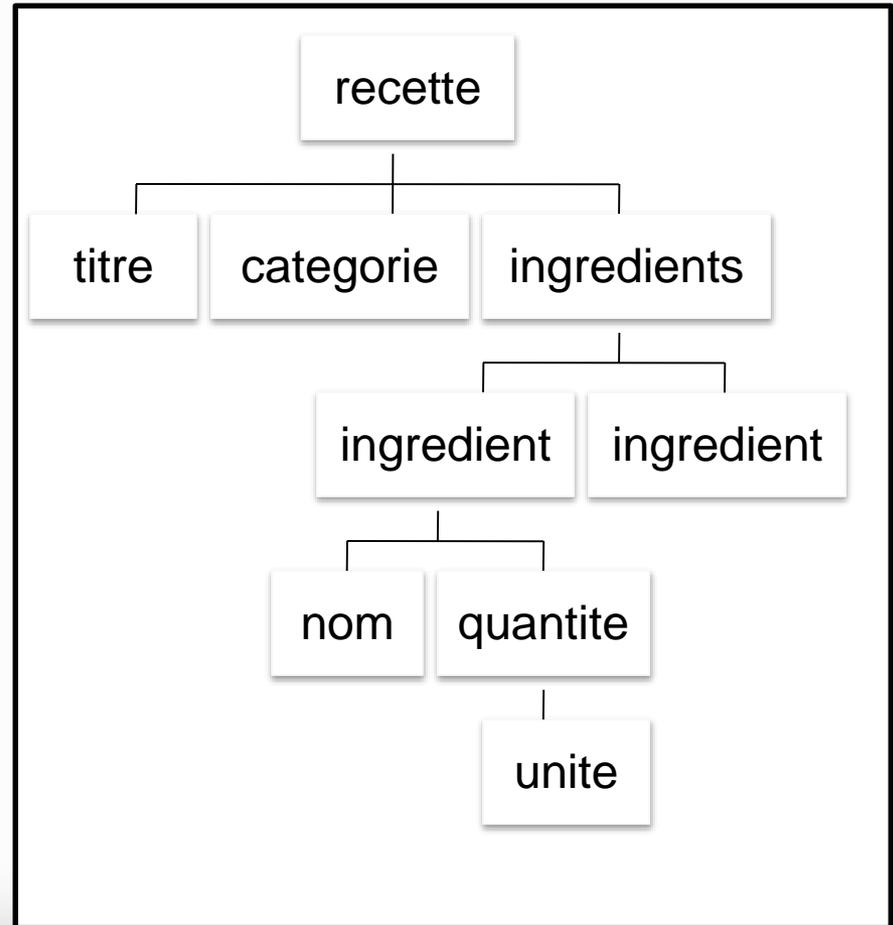
- XPath est un langage d'expression qui sert à identifier les nœuds d'un document XML.
- Recommendation W3C
XPath version 2.0

Les chemins de localisation

Document XML

```
<?xml version="1.0"?>
<recette>
  <titre>Crème au beurre</titre>
  <categorie>Dessert</categorie>
  <ingredients>
    <ingredient>
      <nom>beurre</nom>
      <quantite unite="grammes">
        100
      </quantite>
    </ingredient>
    ...
  </ingredients>
</recette>
```

Arborescence XML



Les chemins de localisation

- Un chemin de localisation sert à désigner une partie de l'arborescence XML. Il est composé de:
 - Un Axe: direction dans l'arbre (les fils, les frères, les ancêtre...)
 - Un type de nœud à localiser: les éléments, les textes...
 - Des prédicats: des conditions à respecter (expressions booléennes à vérifier sur chaque nœud trouvé).

Les chemins de localisation

- Un chemin de localisation sert à désigner une partie de l'arborescence XML. Il est composé de:
 - Un Axe: direction dans l'arbre (les fils, les frères, les ancêtre...)
 - Un type de nœud à localiser: les éléments, les textes...
 - Des prédicats: des conditions à respecter (expressions booléennes à vérifier sur chaque nœud trouvé).

Les chemins de localisation

Les éléments

Il existe deux types d'expressions Xpath:

- Absolue

Si le point de départ est la racine du document

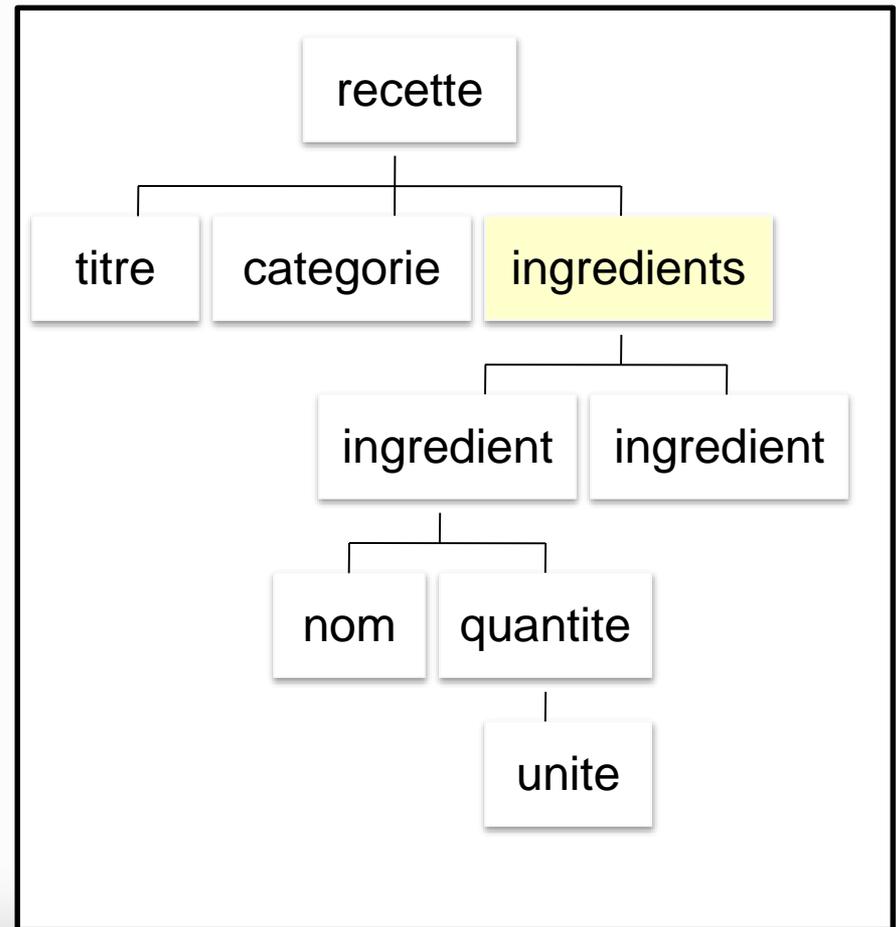
```
/recette/ingredients/ingredient/nom
```

- Relative

Si le point de départ est un autre nœud différent de la racine du document

```
//ingredient/nom
```

Arborescence XML



Les chemins de localisation

Les éléments

Nœud parent ..

Nœud contextuel .

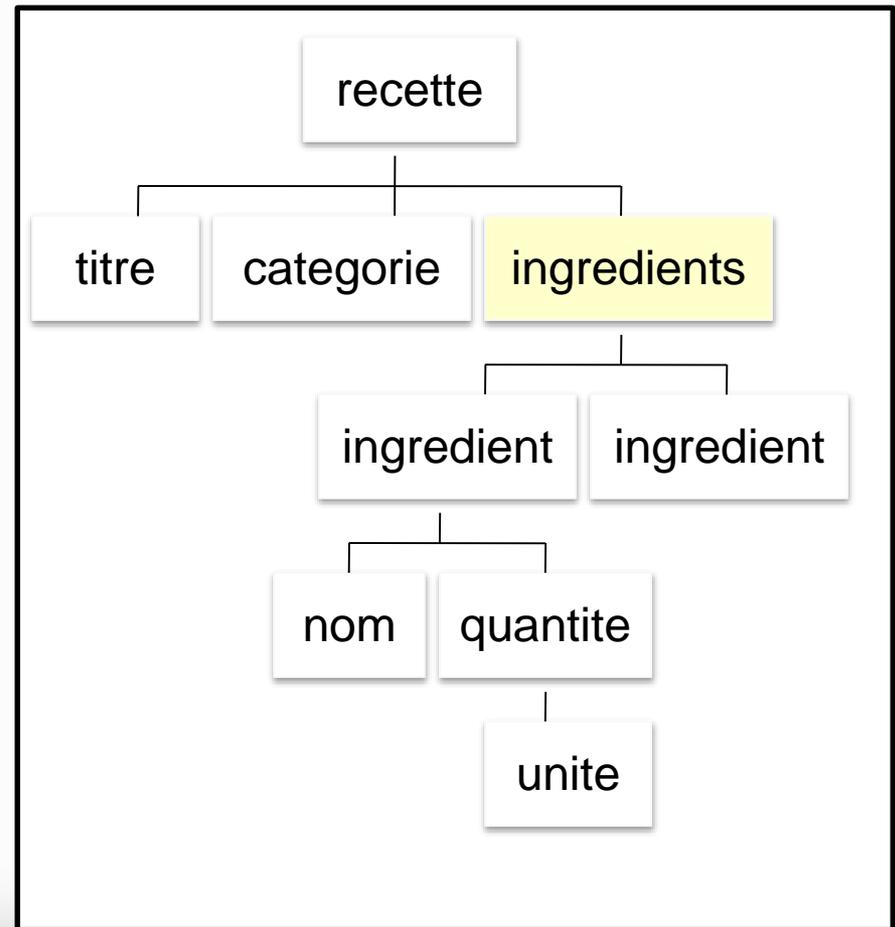
//ingredient

Tous les éléments *ingredient* dans n'importe quel chemin

/recette/ingredients/ingredient[1]

Le premier sous-élément *ingredient* de l'élément contextuel

Arborescence XML



Les chemins de localisation

Les éléments

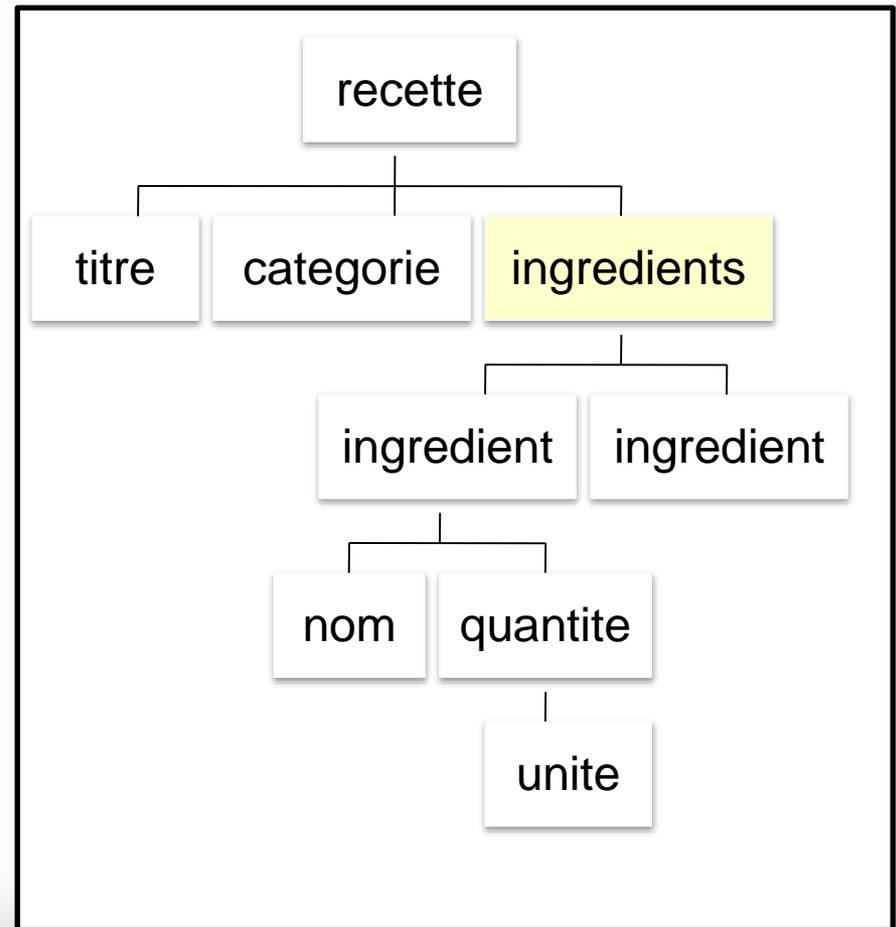
`//ingredient | //titre ..`

Composition de requêtes: tous les éléments **ingredient** avec tous les éléments **titre**

`//ingredient/*`

Tous les éléments fils de tous les éléments **ingredient**

Arborescence XML



Les chemins de localisation: recherche par contexte

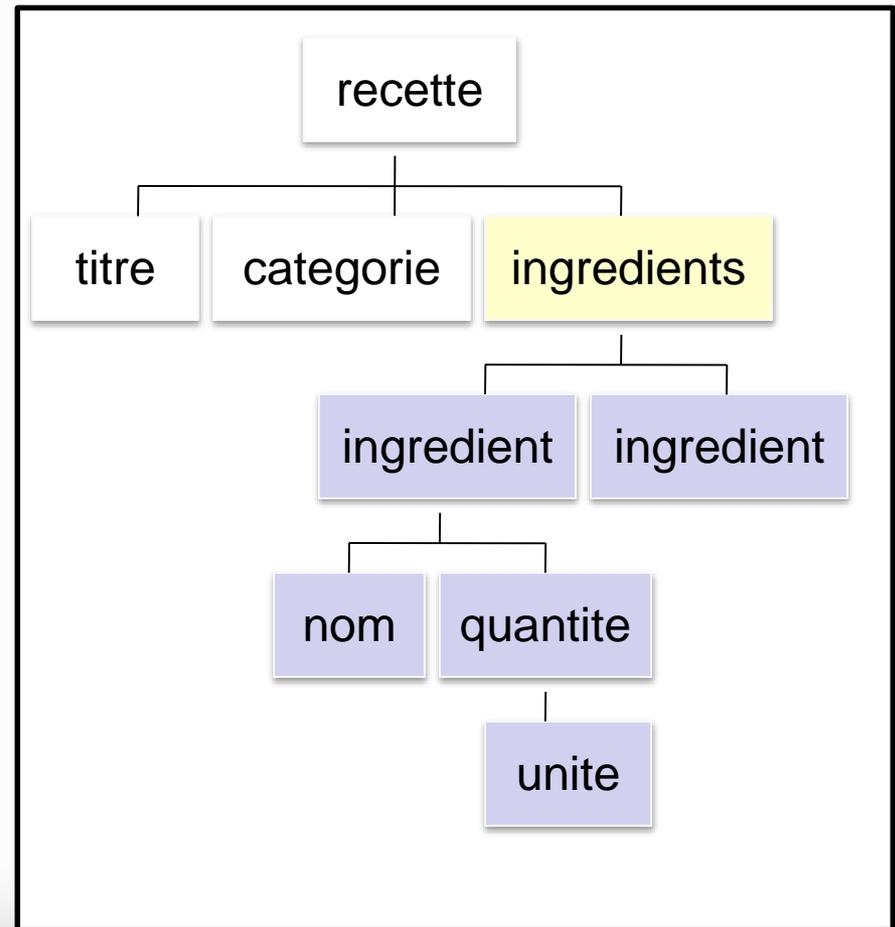
descendant

```
/recette/ingredients/descendant::*
```

Les fils de *ingredients* et leurs descendants

```
<ingredient>  
  <nom/>  
  <quantite unite="gramme"/>  
</ingredient>  
<ingredient/>
```

Arborescence XML



Les chemins de localisation: recherche par contexte

ancestor

```
/recette/ingredients/ingredient/ancestor::*
```

Le père d'un nœud et ses ancêtres

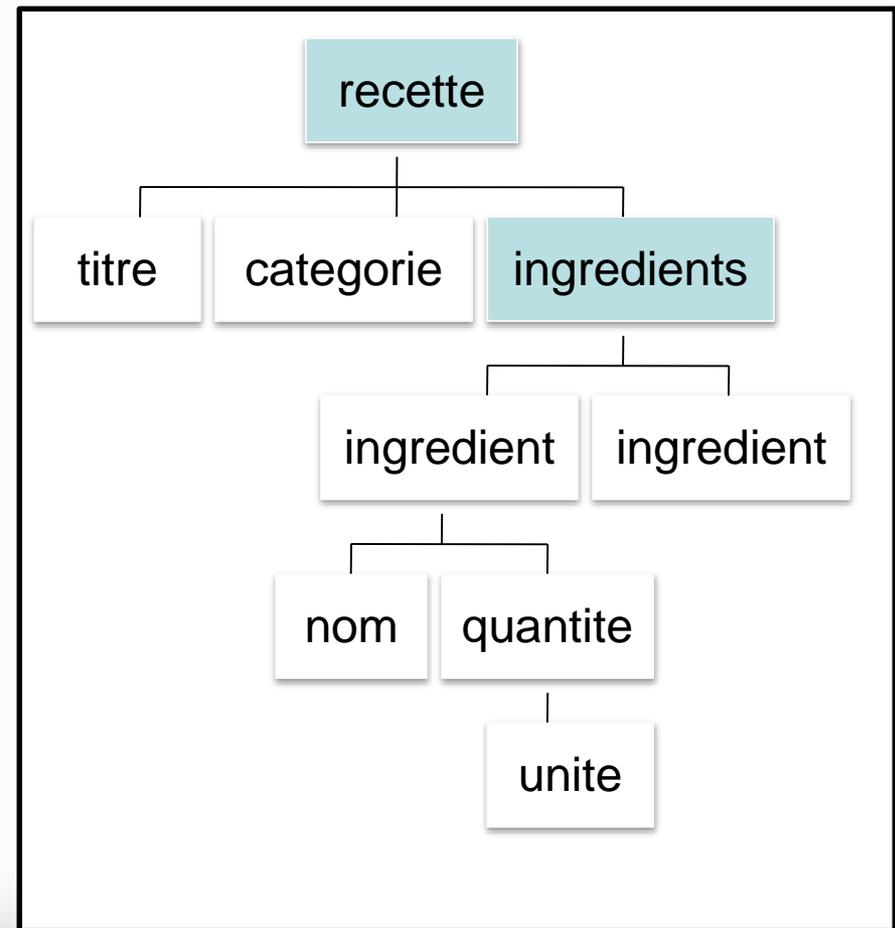
child

```
/recette/ingredients/ingredient/child::*
```

Les fils d'un nœud

➔ **`<nom>beurre</nom>`**
`<quantite unite="gramme"/>`

Arborescence XML



Les chemins de localisation: recherche par contexte

Frère gauche

```
/recette/ingredients//preceding_sibling::*
```

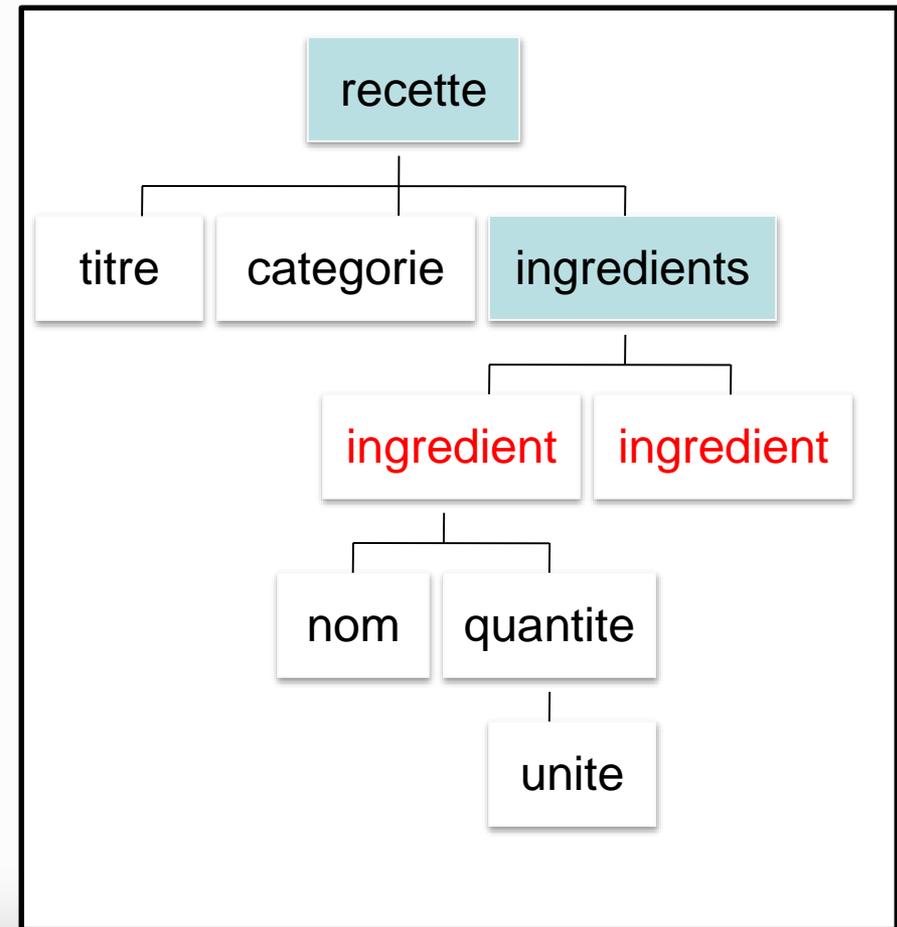
Le frère gauche du nœud
ingredients → categorie

Frère droite

```
/recette/categorie/following_sibling::*
```

Le frère droite du nœud categorie
→ ingredients

Arborescence XML



Les chemins de localisation

Les attributs

`/recette/ingredients/ingredient/@unite`

L'attribut *unite* dans l'élément *quantite*

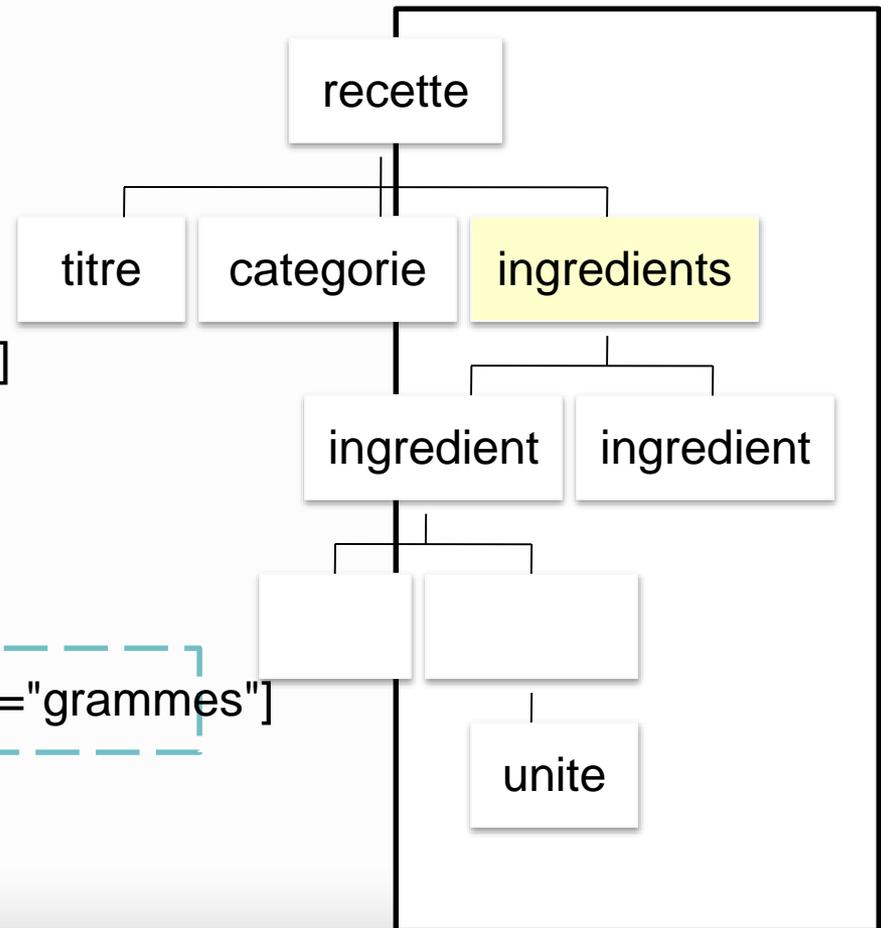
`/recette/ingredients/ingredient/quantite[@unite]`

Les éléments *quantite* ayant un attribut *unite*

`/recette/ingredients/ingredient/quantite[@unite="grammes"]`

Les éléments *quantite* ayant un attribut *unite* possédant la valeur "grammes"

Arborescence XML



Les fonctions XPath

Les fonctions de noeuds

count(//ingredient)

Détermine le nombre des nœuds *ingredient*

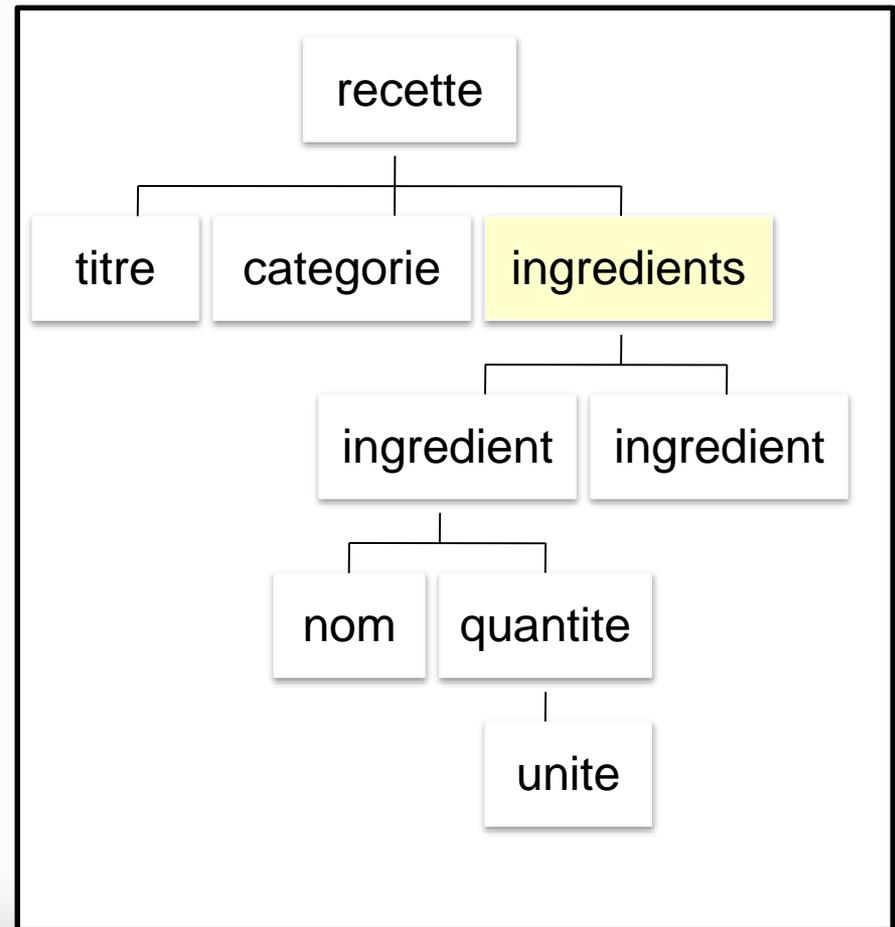
//ingredient[**last()**]

Sélectionne le dernier élément *ingredient* contenu dans l'élément contextuel

//ingredient[**position()**=2]

Sélectionne le 2^{ème} élément *ingredient* contenu dans l'élément contextuel

Arborescence XML



Les fonctions XPath

Les fonctions de noeuds

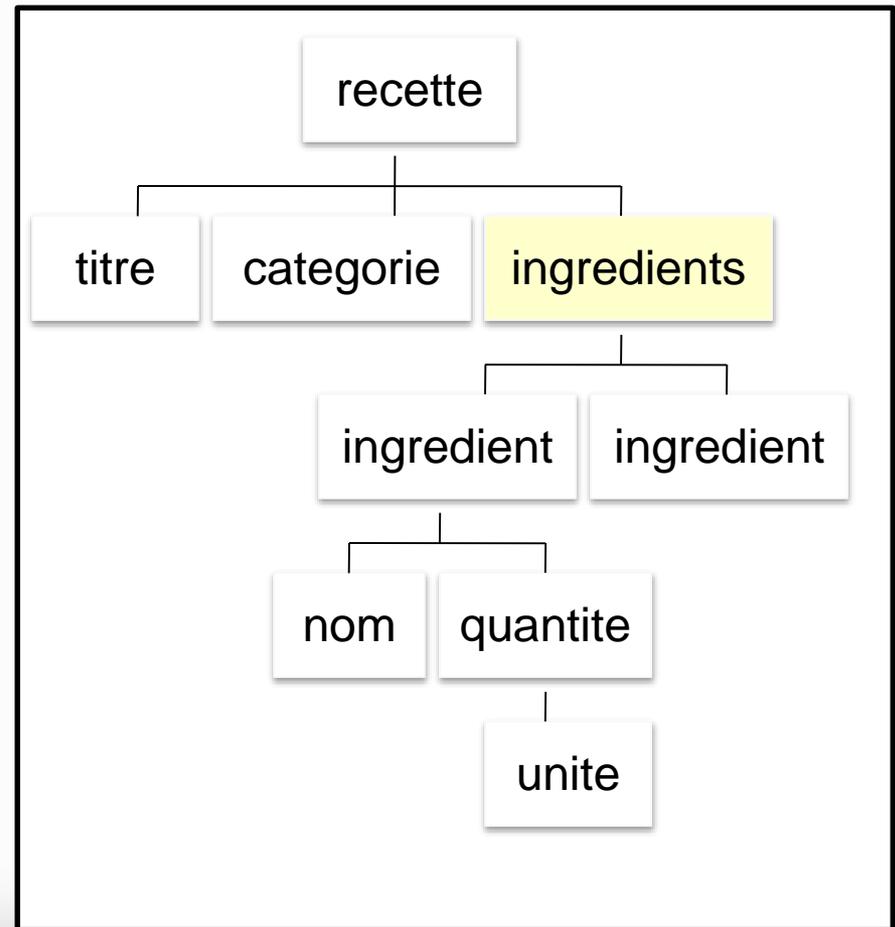
`/recette/categorie/text()`

Tous les nœuds categorie de type textuels

`name()`

Renvoie le nom du nœud contextuel

Arborescence XML



Les fonctions XPath

Les fonctions sur les chaînes de caractère

concat()

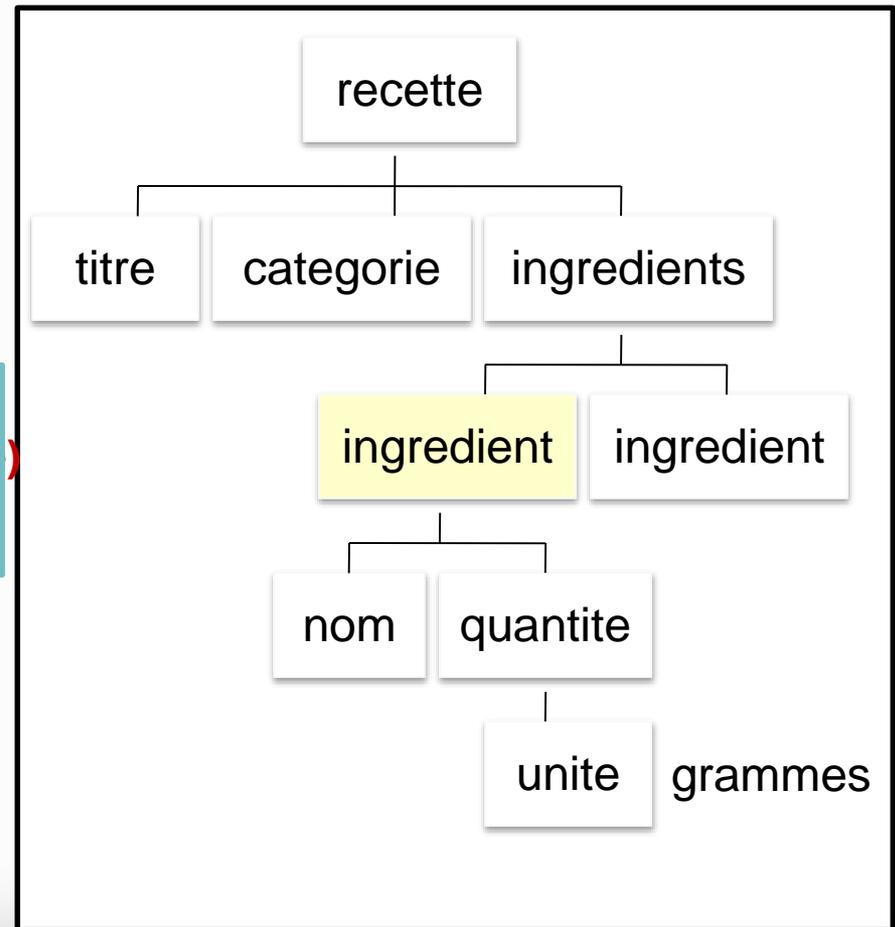
Permet de concaténer plusieurs chaînes de caractère.

Exemple:

concat(« le titre de la recette est » , / recette/titre)

On peut concat plusieurs chaîne de caract .
une chaîne de caract peut être
Ou pas une expression xpath

Arborescence XML



Les fonctions XPath

Les fonctions sur les chaînes de caractère

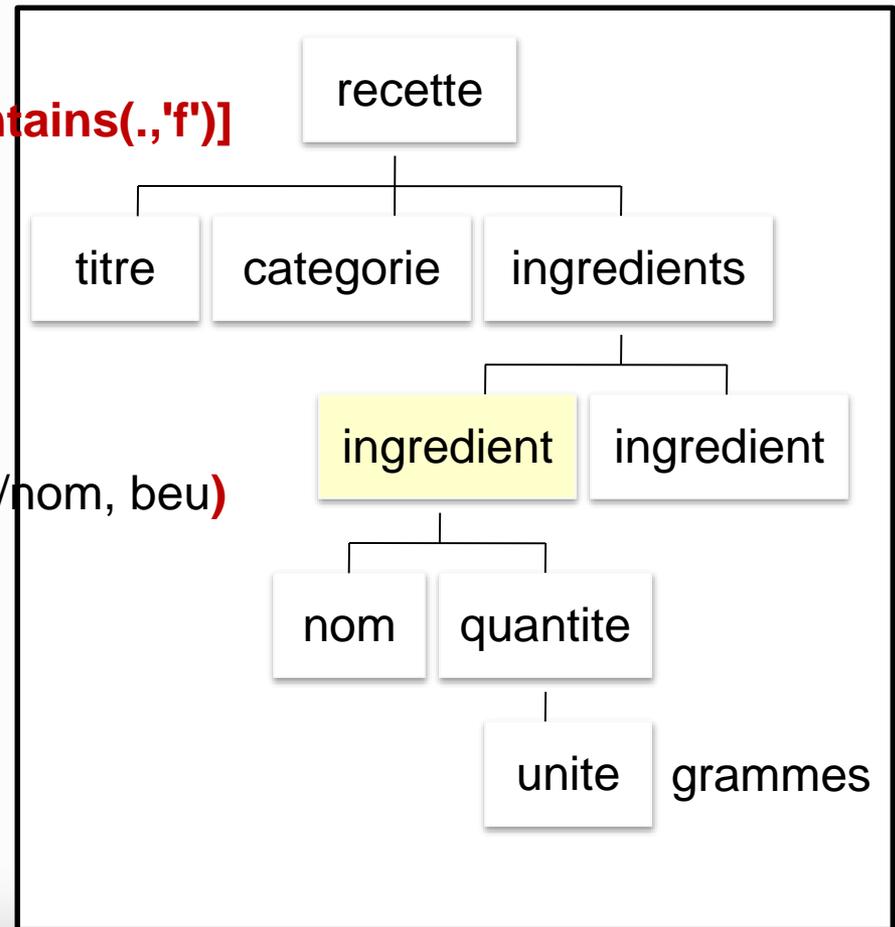
`/recette/ingredients/ingredient/nom[contains(.,'f')]`

Retourne les noms contenant la lettre f.

`starts-with(/recette/ingredients/ingredient/nom, 'beu')`

Retourne true si la première chaîne de caractère passée en paramètre commence par la deuxième chaîne, si non elle retourne false.

Arborescence XML



Les fonctions XPath

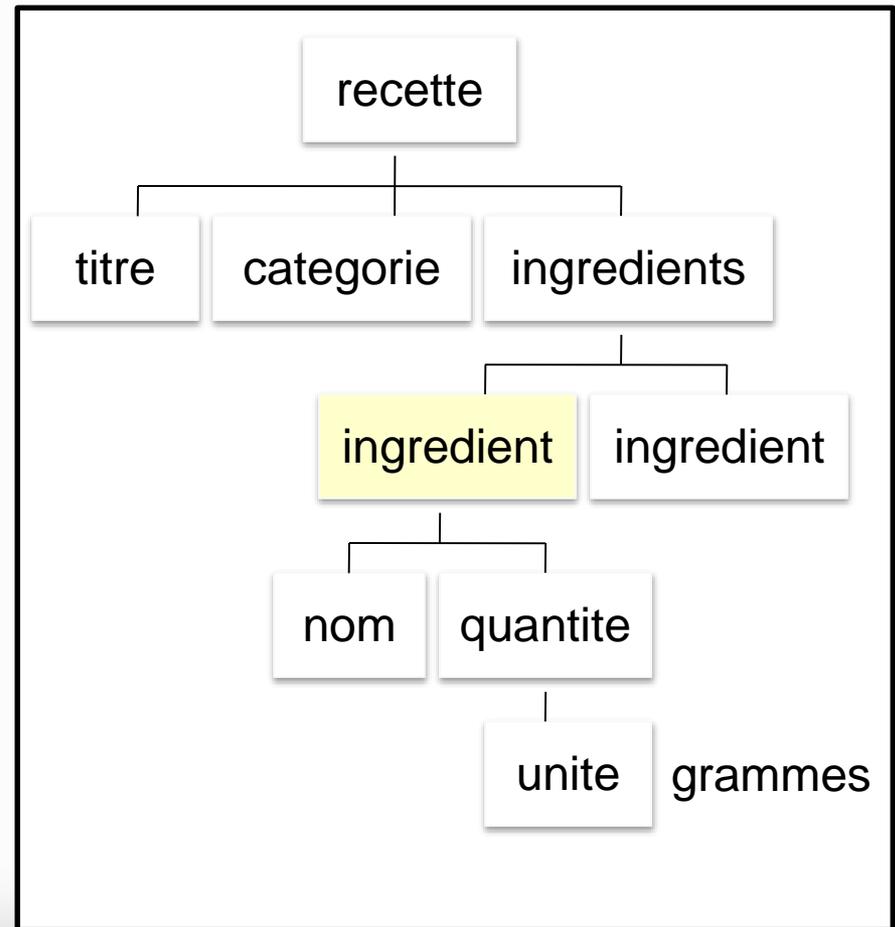
Les fonctions sur les chaînes de caractère

substring(*chaîne1*, *pos*, *long*)

Retourne une sous-chaîne de **chaîne1** à partir de la position **pos** dont la longueur est égale à **long**.

Exemple: **substring**(/recette/titre, 2,5)

Arborescence XML



Les fonctions XPath

Les fonctions sur les chaînes de caractère

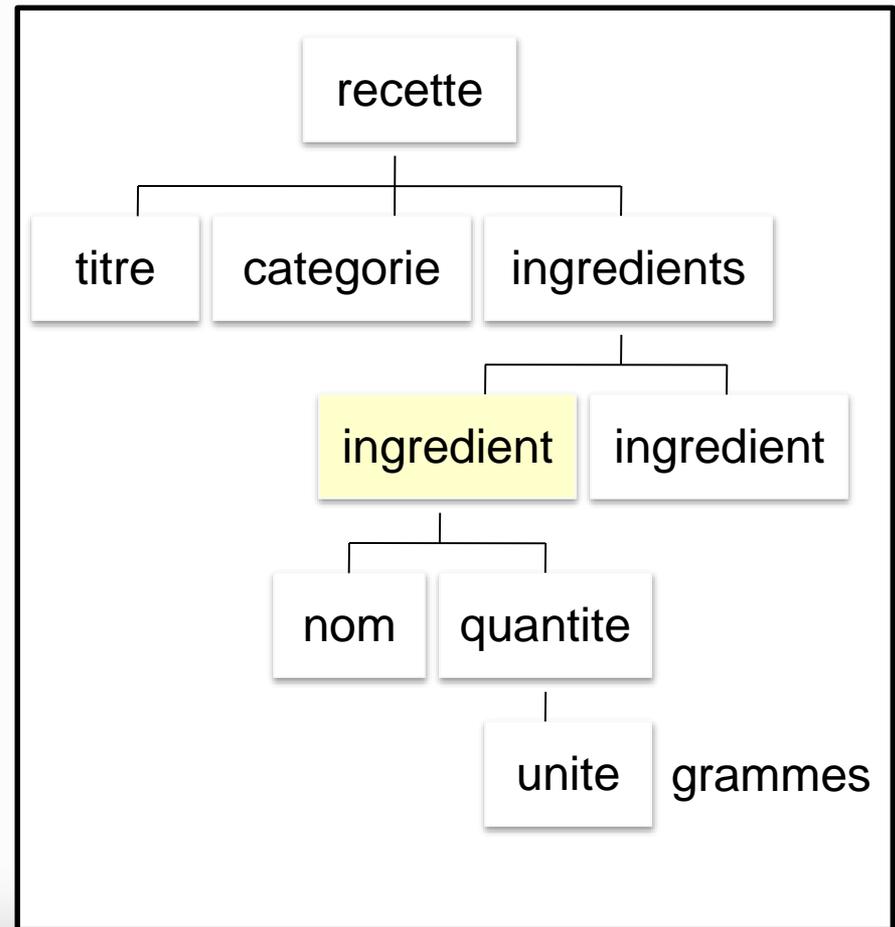
substring-after(*chaîne1*, *chaîne2*)

Retourne une sous-chaîne de **chaîne1** qui suit la première occurrence de **chaîne2**.

Exemple: **substring-after**("012345", "2")

Renvoie la chaîne une sous-chaîne "345"

Arborescence XML



Les fonctions XPath

Les fonctions sur les chaînes de caractère

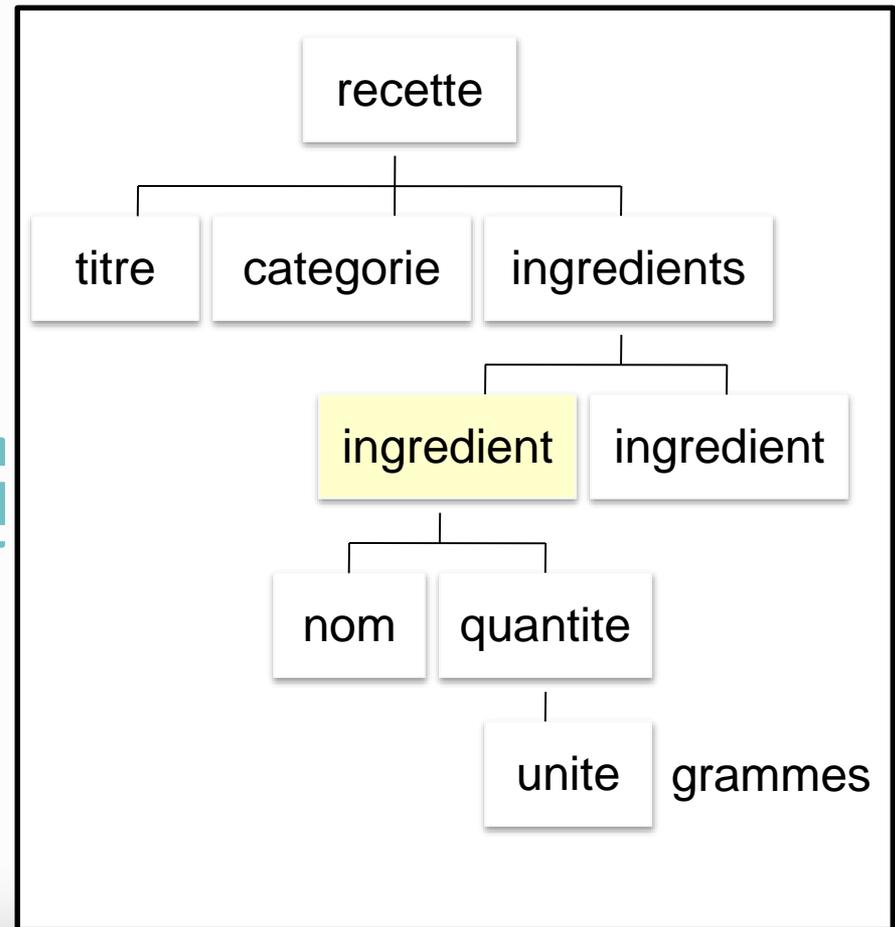
Substring-before(*chaine1*, *chaine2*)

Retourne une sous-chaîne de **chaîne1** qui précède la première occurrence de **chaîne2**.

Exemple: **substring-before**("012345", "2")

Renvoie la sous-chaîne "01"

Arborescence XML



Les fonctions XPath

Les fonctions sur les chaînes de caractère

translate(*chaîne1*, *chaîne2*, *chaîne3*)

Transforme la **chaîne1** en remplaçant chaque occurrence des caractères de **chaîne2** par le caractère correspondant (c'est-à-dire occupant la même position) de **chaîne3**.

Exemple: *pour transformer chaque lettre « a » minuscule en majuscule:*

translate("chaîne", "a", "A")

translate(/monde/nom, " ", "A")

Les fonctions XPath

Les fonctions sur les chaînes de caractère

normalize-space(*chaîne*)

Retourne la chaîne de caractère qu'elle reçoit en argument après en avoir **supprimé les espaces** situés au **début** et à la **fin**, et y avoir remplacé chaque séquence d'espaces consécutifs par un espace unique.

string-length(*chaîne*)

Retourne la longueur de la chaîne de caractère qu'elle reçoit en argument.

Les fonctions XPath

Les fonctions numériques

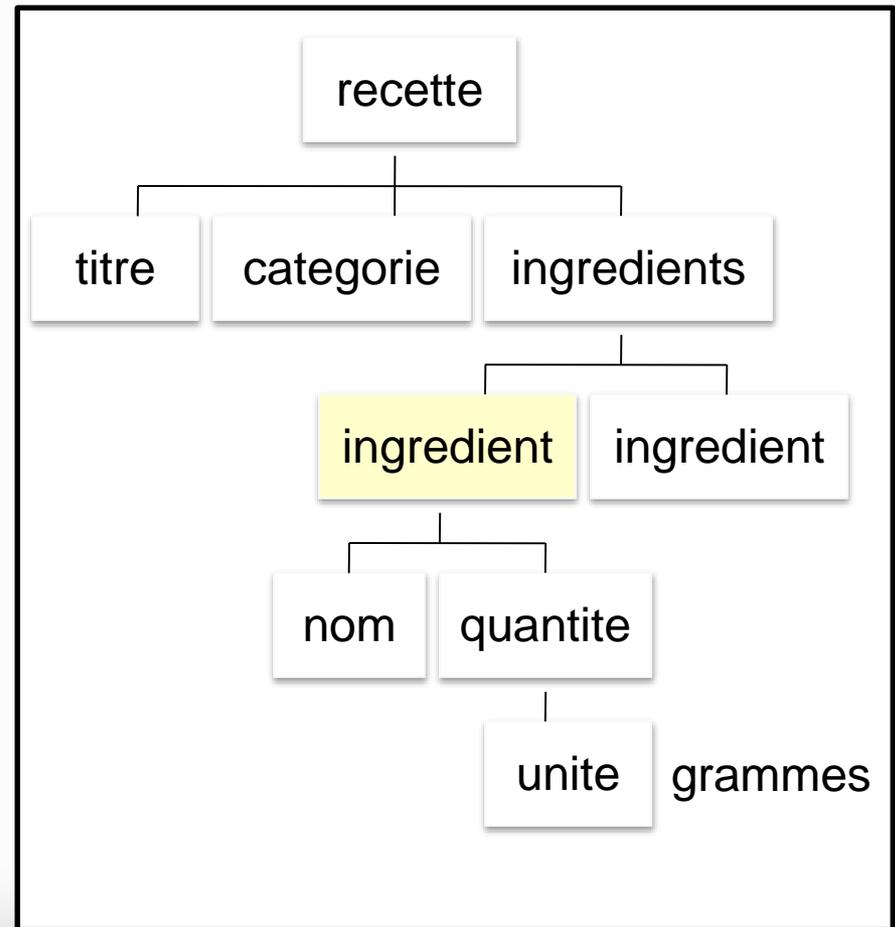
`number(1254)`

Convertir un texte en nombre

`sum(/recette/ingredients/ingredient/quantite)`

Additionner les valeurs de l'ensemble de noeuds

Arborescence XML



Les fonctions XPath

Les fonctions numériques

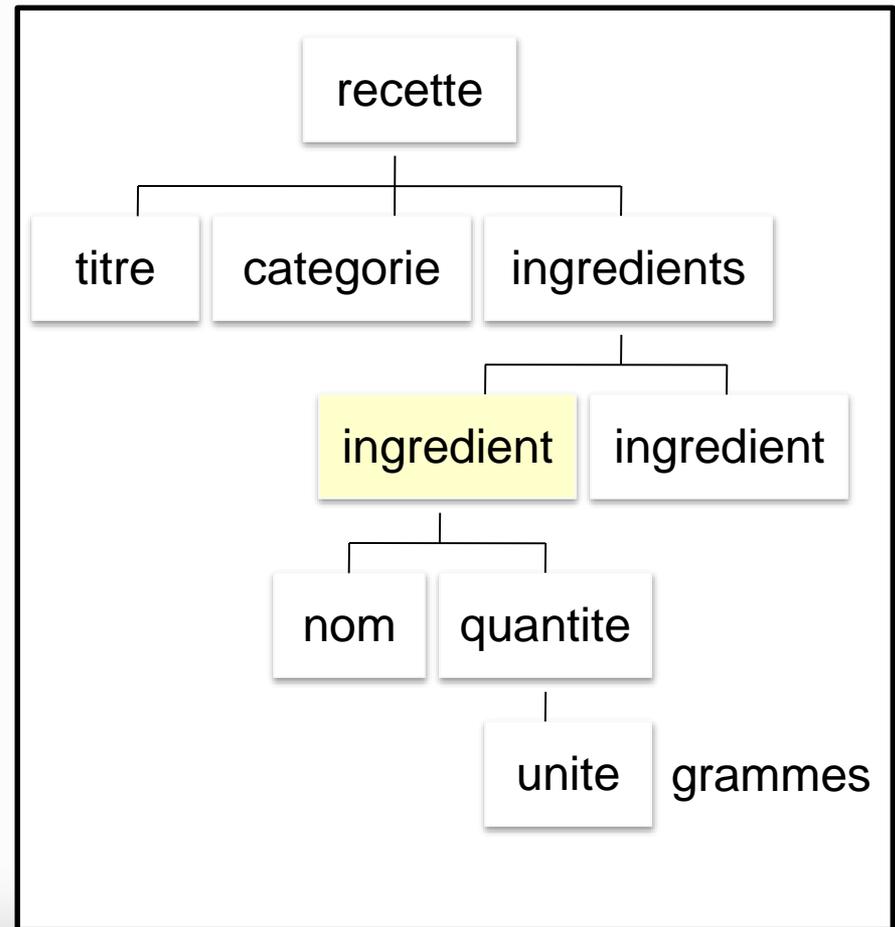
ceiling(50.65)

évalue un nombre décimal et retourne **le plus petit** nombre entier supérieur ou égal au nombre évalué.
→ 51

floor(50.65)

Retourne **le plus grand** entier égal ou inférieur au nombre qu'elle reçoit en argument. → 50

Arborescence XML



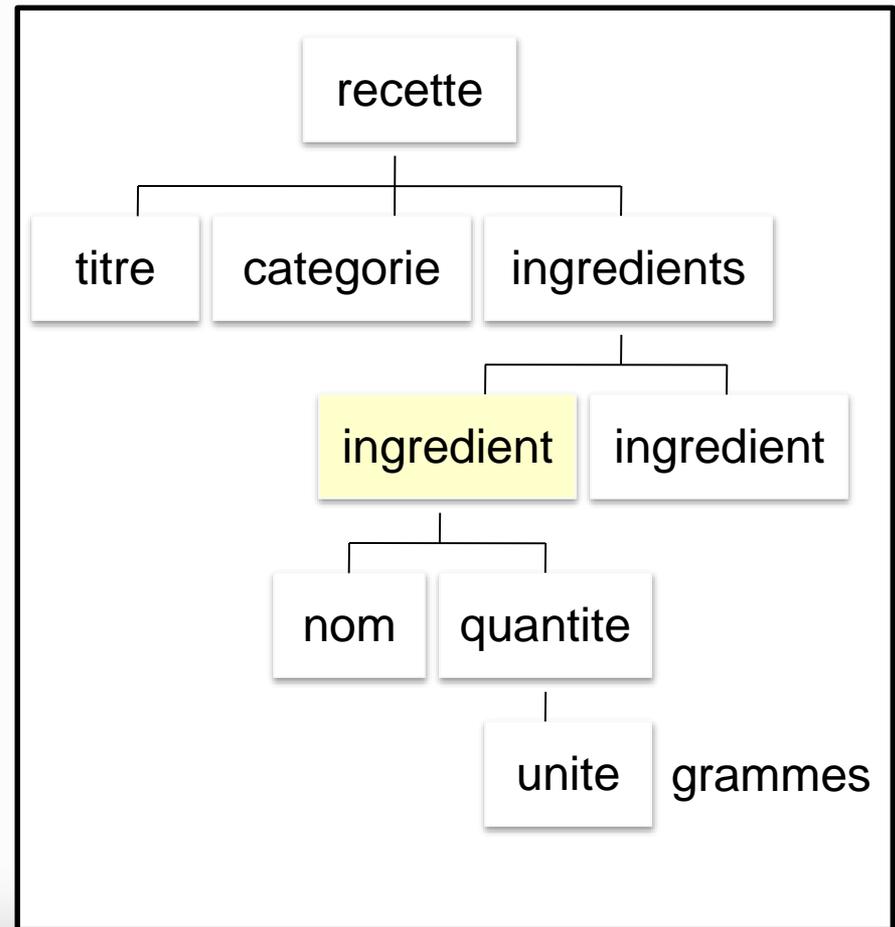
Les fonctions XPath

Les fonctions numériques

`round(50.5)`

Arrondit le nombre qu'elle reçoit à l'entier **le plus proche**. → 51

Arborescence XML



Les fonctions XPath

Les fonctions booléennes

- Il est parfois nécessaire de créer une constante booléenne initialisée à une valeur true ou false.

Cette opération est réalisable par un appel respectivement aux fonctions **true()** et **false()**

- La fonction **not(expression)** : retourne l'inverse de la variable logique de son argument.

- La fonction **boolean (expression)** : retourne **true** si le résultat de l'expression n'est pas null, si non elle retourne **false**.