مدرسة علوم المعلومات
ⵜⵉⵎⵔⵔⴰⵜ ⵏ ⵜⵓⵙⵙⵏⵉⵡⵉⵏ ⵏ ⵜⵣⵔⴼⵜ
ECOLE DES SCIENCES
DE L'INFORMATION

Elément 02 :

Documents structurés

Pr . J. IDRAIS

Plans de cours

- Introduction

1. Notions préliminaires
2. Principes XML
3. Dialecte XML
4. Règles de syntaxe XML
5. Notions de conformité et de validité

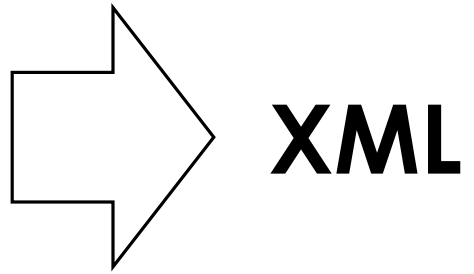
- .....

Introduction

La gestion de données s'est déclinée en deux branches vers 1970:

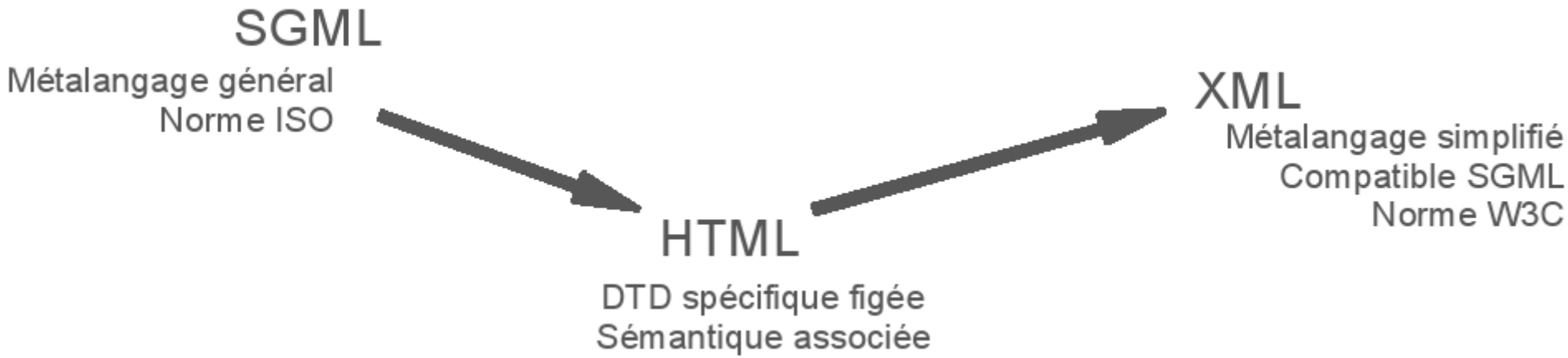
- **BD structurées (Réseau, Relationnel, Objet)**
  - Tables objet-relationnel
  - Langage de requêtes SQL

- **Gestion Electronique de documents (GED)**
  - Documents balisés
  - Recherche d'information par mots-clés
  - Moteurs de recherche (e.g., Google)



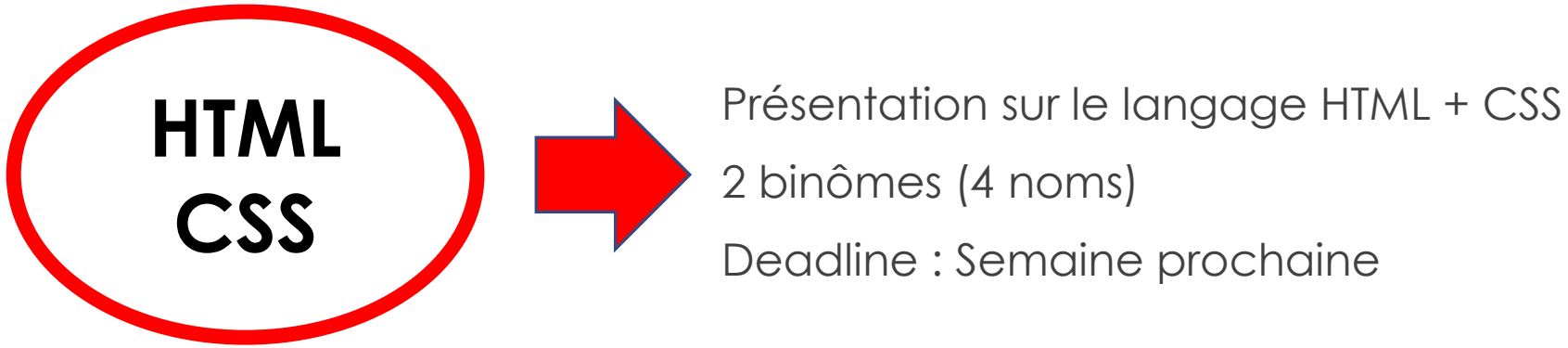
Introduction

- Le langage XML dérive de SGML (**Standard Generalized Markup Language**) et de HTML (**HyperText Markup Language**).
- un langage orienté texte et formé de balises (organisation des données de manière structurée).



Introduction

- **Volontaires** :



Présentation sur le langage HTML + CSS

2 binômes (4 noms)

Deadline : Semaine prochaine

1. Notions préliminaires

Données et metadonnées

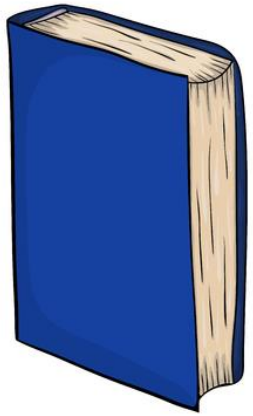
« Une métadonnée est une donnée sur une autre donnée. »

- La manipulation de l'information(oral, écrit papier, écrit, électronique), on est amené à distinguer deux types :
  - une information qui se suffit à elle même peut être désignée par le terme ***donnée***
  - une information qui en décrit ou en commente une autre peut être désignée par le terme ***métadonnée***

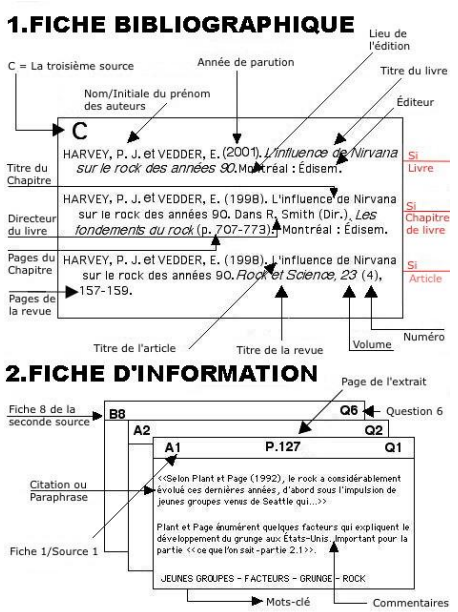
1. Notions préliminaires

Données et metadonnées

« Une métadonnée est une donnée sur une autre donnée. »



Livre



Fiche Bibliographique

1. Notions préliminaires

Données et metadonnées

- la distinction entre ***donnée*** et ***métadonnée*** existe dans tous les domaines de connaissances.
- Une **métadonnée** est une donnée qui en enrichit une autre en lui attachant une description, une propriété, un caractère ou toute sorte d'information utile.

1. Notions préliminaires

Données et metadonnées

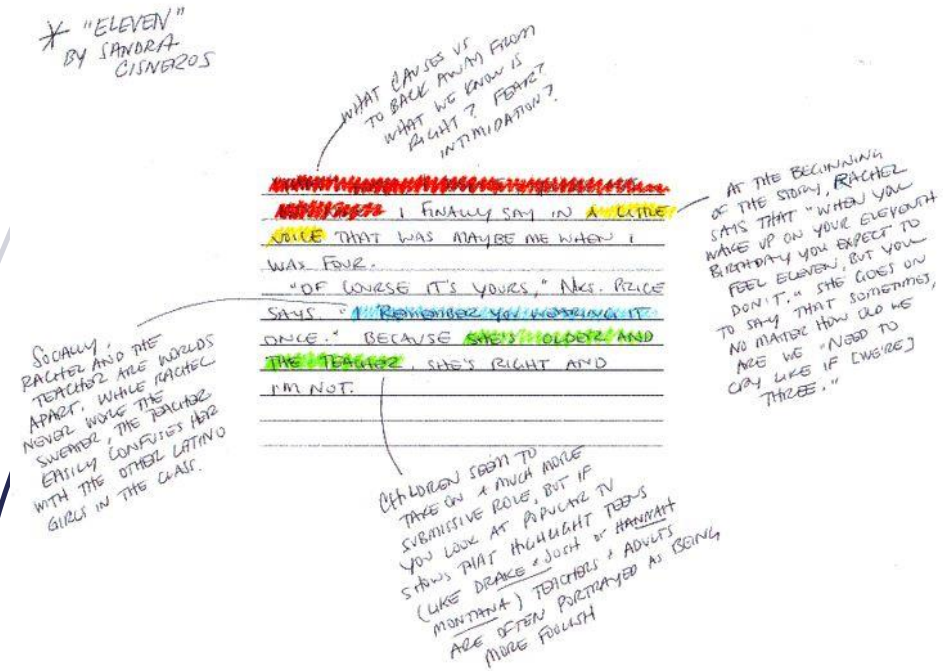
Exemple:

la phrase "***Les ingénieurs développent un logiciel***".

Donnée	**Métadonnée**
Les	Article défini, masculin, pluriel
ingénieurs	Nom commun, masculin, pluriel
développent	Verbe transitif, premier groupe, présent de l'indicatif
un	Article indéfini, féminin, singulier
logiciel	Nom commun, masculin, singulier

1. Notions préliminaires

De l'annotation au balisage



- L'annotation est une activité humaine **visant à enrichir par des remarques**, ou à corriger partiellement, des textes déjà écrits.

- Elle peut suivre **des règles strictes**

(Ex: le cas des éditeurs annotaient un manuscrit pour demander à son auteur de le réviser.)

1. Notions préliminaires

De l'annotation au balisage

- Une transposition vers l'annotation des textes numérisés conduit à un système de balisage où le fragment vedette est délimité par une balise ouvrante et une balise fermante.

Balise ouvrante : <nom_balise>

Balise fermante : </nom_balise>

<nom_balise> texte balisé </nom_balise>

- Les contraintes sur les noms des balises XML seront précisées par la suite.

1. Notions préliminaires

De l'annotation au balisage

<nom_balise>

Les balises constituent donc un **outil pour annoter et structurer des textes**, tout en **conservant leur portabilité**.

En effet le texte annoté reste du "**texte brut**", lisible sur n'importe quelle plate-forme et par n'importe quel logiciel.

</nom_balise>

1. Notions préliminaires

Noms XML

Un ***nom XML*** est une chaîne de caractères qui respecte certaines contraintes de forme et qui, de ce fait, peut être utilisée pour jouer un rôle particulier dans un document XML.

Les contraintes que doit respecter un *nom XML* sont les suivantes :

- Les *noms XML* peuvent contenir les caractères alphanumériques (lettres ou chiffres), y compris les lettres accentuées et les lettres d'alphabets non latins.
- Ils peuvent également contenir le caractère souligné "_", le tiret "-", ou le point "."
- Ils ne doivent contenir **aucun autre signe de ponctuation** que ceux mentionnés ci-dessus.
- Ils ne doivent contenir **aucune sorte d'espace, ni de saut de ligne**.

1. Notions préliminaires

Noms XML

Un ***nom XML*** est une chaîne de caractères qui respecte certaines contraintes de forme et qui, de ce fait, peut être utilisée pour jouer un rôle particulier dans un document XML.

Les contraintes que doit respecter un *nom XML* sont les suivantes :

- Le premier caractère **doit être une lettre ou le caractère "_"** , mais ni un chiffre, ni un tiret, ni un point.
- Les *noms XML* peuvent contenir une occurrence du caractère ":", mais seulement dans des situations particulières (espaces de noms).
- Les *noms XML* sont sensibles à la casse (une minuscule et la majuscule correspondante sont considérées comme des lettres différentes).
- Les *noms XML* ne sont pas limités en longueur.
- Le prefixe "xml" relève d'un usage normalisé.

2. Principes de XML

1. Séparation de la forme et du contenu

- XML est **conçu pour structurer du contenu** sans se préoccuper, a priori, d'une quelconque **visualisation**.

- Toute sorte de **structuration est concevable** selon **la nature des données** et l'objectif suivi.

2. Principes de XML

1. Séparation de la forme et du contenu

Exemples de données structurées sous forme d'un document XML :


DANS LES COULISSES DU NET

Qui pourrait faire sauter Internet ?

Terrorisme, piratage, pannes, engorgement : de lourdes menaces pèsent sur la Toile. Comment est-elle protégée ? Quels sont les risques ?

Les gardiens de la Toile

- une revue : ensemble d'articles structurés en résumé, sections, sous-sections, paragraphes, figures, tableaux, bibliographie, etc.

- un catalogue de fournitures : ensemble d'articles structurés en désignation, descriptif, prix, taille, etc.

2. Principes de XML

1. Séparation de la forme et du contenu


DANS LES COULISSES DU NET

Qui pourrait faire sauter Internet ?

Terrorisme, piratage, pannes, engorgement : de lourdes menaces pèsent sur la Toile. Comment est-elle protégée ? Quels sont les risques ?

Les gardiens de la Toile

- Le premier exemple va plutôt **concerner l'édition électronique**,

- le second exemple est orienté vers une **exploitation des données** sous forme de requêtes.

2. Principes de XML

2. Portabilité

- XML permet de mémoriser des *données structurées* de tous domaines sous forme de *fichiers textes*.
- XML est caractérisé par une **portabilité universelle** (n'importe quel plate-forme ou système est capable de lire des fichiers textes).

3. Visualisation indépendante

- XML **n'est pas** un *langage de présentation* comme HTML, mais la visualisation des documents est possible, de manière indépendante, en leur attachant une **feuille de style** qui spécifie comment chaque élément doit être traité graphiquement.

4. Programmation

- XML **n'est pas** un langage de programmation, mais des *instructions de traitement* peuvent être intégrées aux documents XML et fournir des informations aux applications auxquelles sont destinées le document.

2. Principes de XML

5. Encodage des caractères

- XML utilise par défaut le jeu de caractères ***Unicode*** et le format d'encodage ***UTF-8***,

  (d'autres systèmes d'encodage peuvent être utilisés à condition d'être spécifiés dans les documents.

6. Modèles de documents

- La structure des documents XML peut être contrôlée par des modèles (DTD, XMLSchéma), ce qui permet à des communautés d'intérêt de travailler avec des outils efficaces (édition électronique, organisation des données) spécialisés sur leur domaine.

7. Meta langage

- XML permet de structurer toutes sortes de connaissances et est utilisé dans un grand nombre de domaines. Lorsqu'une communauté utilise XML pour structurer ses propres données elle peut être amenée à définir un jeu de balises, nommé **dialecte XML**, qui lui est spécifique.

3. Dialectes XML

Sigle	Nom	Domaine d'utilisation
CML	Chemical Markup Language	Chimie
	DocBook	Édition de livres, articles etc.
MathML	Mathematical Markup Language	Mathématiques
SMIL	Synchronized Multimedia Integration Language	Multimedia
SVG	Scalable Vector Graphics	Graphiques vectoriels
XHTML	eXtensible HyperText Markup Language	Visualisation de pages Web
	XML Schema	Modèles de documents
XSL-FO	eXtensible Stylesheet Language - Formatting Objects	Mise en page de documents
XSLT	eXtensible Stylesheet Language Transformations	Transformation de documents XML
XUL	Xml-Based User interface Language	Interfaces graphiques

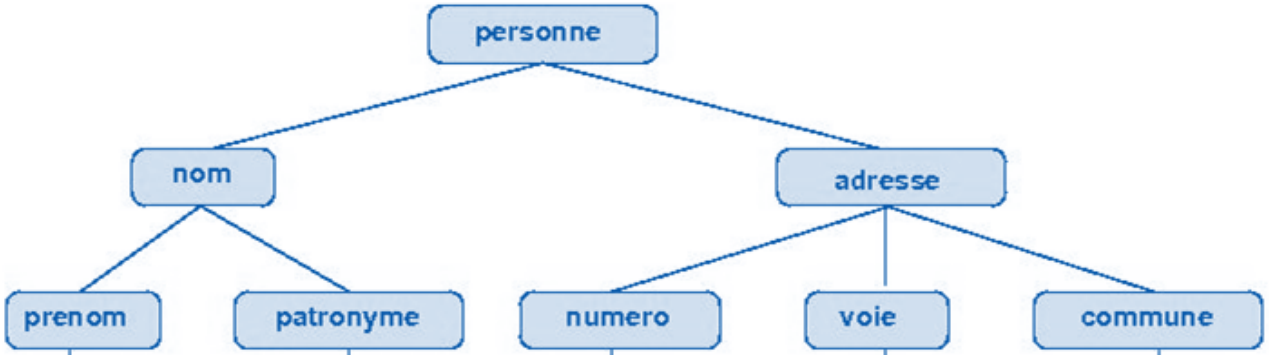
4. Règles de syntaxe XML

Structure générale d'un document XML

La structure fondamentale d'un document XML est un **arbre d'éléments** qui reflète la **structure logique** des informations qu'il mémorise.

Cet arbre est réalisé par un jeu de balises ouvrantes et fermantes, correctement parenthèses, qui représentent les **éléments**.

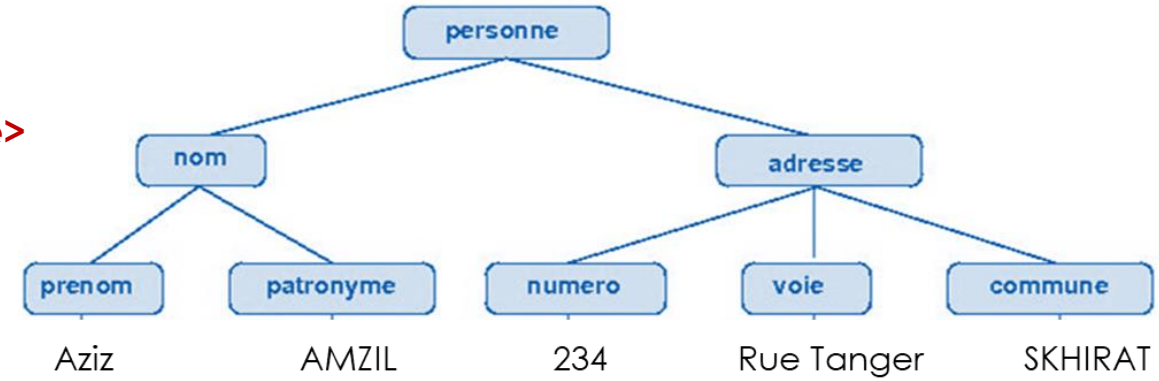
Il y a un seul **élément racine**.



4. Règles de syntaxe XML

Structure générale d'un document XML

```
<personne>
    <nom>
        <prenom>Aziz</prenom>
        <patronyme>AMZIL</patronyme>
    </nom>
    <adresse>
        <numero>234</numero>
        <voie>rue Tanger</voie>
        <commune>SKHIRAT</commune>
    </adresse>
</personne>
```



4. Règles de syntaxe XML

Éléments et attributs

Éléments

Un élément se compose d'un contenu entouré par **une balise ouvrante** et la **balise fermante** correspondante.

- Le nom de l'élément qui se trouve entre les caractères < et > doit être un **nom XML**.
- Il est laissé au choix du créateur du document ou du modèle, mais une bonne pratique veut qu'il caractérise le contenu.

```
<nom_élément>contenu</nom_élément>
```

Le contenu d'un élément peut prendre différentes formes :

- contenu vide : `<vide></vide>` qui s'écrit aussi : `<vide/>`

4. Règles de syntaxe XML

Éléments et attributs

Attributs

- On peut attacher des informations supplémentaires à tout élément en lui assignant un ou plusieurs **attributs**.
- Un **attribut** est un couple *nom=valeur* qui se trouve à l'intérieur de la balise ouvrante de l'élément.
- Le nom d'un attribut doit être un *nom XML*.

```
<nom_élément nom_attribut="valeur_attribut" >
        Contenu
</nom_élément>
```

- Un même élément peut avoir plusieurs attributs, dont les noms doivent être tous différents.
- L'ordre d'écriture des attributs à l'intérieur de la balise ouvrante d'un élément n'est pas significatif.
- La valeur d'un attribut est obligatoirement entourée de séparateurs (guillemets simples ou doubles).

4. Règles de syntaxe XML

Éléments et attributs

Attributs

```
<personne  id=''55433''  sex=''M''>
        <nom>
                <prenom>Aziz</prenom>
                <patronyme>AMZIL</patronyme>
        </nom>
        <adresse>
                <numero>234</numero>
                <voie>rue Tanger</voie>
                <commune>SKHIRAT</commune>
        </adresse>
</personne>
```

4. Règles de syntaxe XML

Prologue

Prologue

Le prologue, ou déclaration XML est recommandée dans les documents XML.

il constitue **impérativement la première ligne** du document et ne doit être précédé d'aucune autre ligne, ni de blancs, ni de commentaires.

Le prologue minimum informe sur la version du langage et s'écrit sous la forme suivante :

```
<?xml version="1.0" ?>
```

On peut y ajouter **le système d'encodage** (UTF-8 pris par défaut).

```
<?xml version="1.0" encoding = "UTF-8" ?>
<?xml version="1.0" encoding = "ISO-8859-1" ?>
```

4. Règles de syntaxe XML

Prologue

Prologue

`standalone` : indique si le document nécessite (yes) ou non (no) la présence d'autres documents (modèles, des feuilles de styles ou d'autres documents XML à inclure).

La valeur "no" est utilisée par défaut.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
```

4. Règles de syntaxe XML

Sections CDATA

- **Sections CDATA**

Les sections CDATA XML contiennent du texte brut qui doit être inclus, mais pas analysé, avec le code XML qui le contient.

Une section CDATA XML peut contenir n'importe quel texte. Cela inclut des caractères XML réservés. La section CDATA XML se termine par la séquence «]]> ». Cela implique les points suivants :

- Vous ne pouvez pas utiliser une expression incorporée dans un littéral CDATA XML, car les délimiteurs d'expression incorporées sont du contenu CDATA XML valide.
- Les sections CDATA XML ne peuvent pas être imbriquées, car content ne peuvent pas contenir la valeur «]]> ».

<![CDATA[*contenu de la section*]]>

4. Règles de syntaxe XML

Sections CDATA

- **Sections CDATA**

Vous pouvez affecter un littéral CDATA XML à une variable ou l'inclure dans un littéral d'élément XML.

<![CDATA[*contenu de la section*]]>

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
`CDATASection`	✓ 1	✓ 12	✓ 1	✓ 12.1	✓ 3	✓ 18	✓ 4	✓ 12.1	✓ 1	✓ 1.0	✓ 37

4. Règles de syntaxe XML
Sections CDATA

Sections CDATA

- Une section CDATA ne peut pas contenir la chaîne de caractères "]]>".
- Les sections CDATA imbriquées ne sont pas autorisées.
- Les "]]>" qui marque la fin de la section CDATA ne peut pas contenir d'espaces ou de sauts de ligne.
- CDATA fait partie du document alors que le commentaire ne fait pas partie du document.

4. Règles de syntaxe XML

Instruction de traitement (Processing Instruction)

- Un instruction de traitement est spécifiée dans la partie prologue. Elle permet de passer des instructions à une application externe au document XML.
- Exemple : lien vers des feuilles de style

```
<?xml-stylesheet type="text/xsl" href="biblio.xsl"?>

<?xml-stylesheet href="biblio.css" type="text/css"?>
```

5. Notions de conformité et de validité

Document XML bien formé

Un document XML doit contenir :

- Une déclaration XML,
- Un ou plusieurs éléments,
- Un élément racine encapsulant tous les autres éléments et leurs attributs,

Eléments :

- Les éléments non vides ont une balise de début et de fin,
- Sont correctement imbriqués <P> <EM> ... </EM> </P>
- Les éléments vides ont un / à la fin de la balise,
- Les noms des balises ouvrantes et fermantes correspondent.

Tout document XML qui respecte ces règles est dit **document bien formé**

Attributs :

- Un nom d'attribut n'apparaît que dans la balise ouvrante et une seule fois,
- Les valeurs des attributs sont entre guillemets ou apostrophes,
- La valeur des attributs n'appelle pas d'entités externes

5. Notions de conformité et de validité

Document XML bien formé

Document XML valide

Un document XML est valide s'il est :

- **Bien formé** (well formed document) c-à-d il vérifie les règles XML,
- il est conforme à une DTD (**Document Type Definition**) ou à un schéma XML, sorte de grammaires définissant la structure syntaxique d'un document XML.

5. Notions de conformité et de validité

Exercice :

1) Le document ci-dessous n'est pas bien formé.

Noter la position et la nature de chaque erreur de syntaxe.

```
01 : <?xml version="1.0" ?>
02 : <!-- Annuaire d'illustration -->
03 : <annuaire>
04 :   <personne <!-- une lere personne --> >
05 :     <nom>BEN AHMED</nom>
06 :     <prénom>YASSINE <!- ou Polo --> </prenom>
07 :     <date format="ISO" format="fr-fr'>2022-12-08
08 :     <telephone/>
09 :   </personne>
10 :   <personne> <!-- une 2nd personne -->
11 :     <nom>AIT AHMED</nom>
12 :     <prenom courant='Vrai'>SALMA</prenom>
13 :     <date format=ISO>2021-06-24</date>
14 :     <telephone>
15 :       <indicatif tel> 212 </indicatif tel>
16 :       <num#tel> 61 40 44 02 </num#tel>
17 :     </personne>
18 :   </telephone>
19 : </annuaire>
20 : <annuaire>
21 :   <personne>
22 :     <nom>AMZIL</nom>
23 :     <prenom>Abdelaaziz</prenom>
24 :     <telephone/>
25 :   </personne>
26 : </annuaire>
```

Le langage XML
(eXtended Markup Language)

- Un format général de documents orienté texte.
- Un standard incontournable de l'informatique.
- utilisé pour le stockage de documents, la transmission de données entre applications.
  - Simple,
  - Flexible
  - extensible

les objectifs de conception

- XML doit pouvoir être utilisé sans difficulté sur Internet
- XML doit soutenir une grande variété d'applications
- XML doit être compatible avec SGML et HTML
- Il doit être facile d'écrire des programmes traitant les documents XML
- Le nombre d'options dans XML doit être réduit au minimum, idéalement à aucune

les objectifs de conception

- Les documents XML doivent être lisibles par l'homme et raisonnablement clairs
- La spécification de XML doit être disponible rapidement
- La conception de XML doit être formelle et concise
- Il doit être facile de créer des documents XML
- La concision dans le balisage de XML est peu importante