

Technologie *Web*

1. Données bioinformatiques en XML

- Utilisez la page Web

`http://forge.info.univ-angers.fr/~gh/Webrd/m1info2012.php`

pour déterminer, à partir de votre date de naissance l'identifiant de la protéine à utiliser. Vous indiquerez votre date de naissance et ce numéro sur votre copie. Allez ensuite sur le site d'Uniprot et cherchez cette protéine dans la *Protein Knowledgebase (UniProtKB)*.

- A l'aide de ce numéro, rapatriez en local le fichier XML associé et modifiez la ligne 2 pour qu'elle ne contienne que `<uniprot>`. Combien a-t-il d'éléments ? et d'éléments distincts ? Combien d'attributs distincts sont utilisés ? Vous détaillerez comment vous avez trouvé ces informations (quel logiciel, outil ou commande et avec quelle option, commande ou manipulation).
- Donnez au sens de XPATH en mode AFP le chemin absolu complet de la première personne listée en tant qu'auteur pour la première référence bibliographique. Quelle est l'expression XPATH minimale qui fournit la même information ?
- Ecrire une transformation XSL 1.0 qui à partir de votre fichier XML produit un fichier XML correspondant à la structure suivante

```
<protein id="votre_numéro">
  <premierAuteur>
    leNomTrouvé
  </premierAuteur>
</protein>
```

2. Question de culture et d'expression

XML fonctionne avec une structure d'arbre et un arbre est un graphe particulier. *Mais peut-on "bien représenter" un "vrai" graphe avec XML ?*

Vous essaieriez de répondre à cette question en fournissant une argumentation raisonnée, en au moins dix lignes, avec au moins trois mots de trois syllabes ou plus. Vous distinguerez les graphes orientés de ceux non orientés et, si vous avez plus de connaissances, vous pourrez discuter de multigraphes, de graphes valués, pondérés, probabilistes, conceptuels...

3. Gestion de joueurs : pseudos, ages et roles

Dans le cadre d'une restructuration des fichiers d'une centrale de jeux, on décide de modéliser des joueurs à l'aide d'une structure d'arbre exprimée en XML.

L'élément-racine sera nommé `jrs` si vous êtes né(e) avant le premier juillet et `plrs` sinon. Chaque joueur (élément nommé respectivement `jr` ou `plr`) est doté obligatoirement d'un attribut `id` et d'un attribut `pseudo`. Il comprend dans cet ordre un élément `nom`, un élément `prénom`, un élément `sexe` facultatif, un élément `date de naissance (ddn)` au format `jj/mm/aaaa` et enfin un élément `role` facultatif.

1. Donnez le fichier XML associé aux trois joueurs suivants :
 - Moi, `id "j1"` et `pseudo "moi"`, avec le sexe codé en 1/2 façon sécurité sociale, `role "modérateur"` ;
 - Mickey, `id "j1928_1"`, `pseudo "mic"`, sans sexe, `ddn 18/11/1928`, sans `role` (prénom vide mais présent) ;
 - Minnie, `id "j1928_2"`, `pseudo "min"`, sans sexe, `role "femme"` (prénom vide mais présent).

Pour la `ddn` de Minnie, vous utiliserez votre jour et votre mois de naissance mais avec l'année 1928.

2. Donner une grammaire DTD minimale qui utilise tous les éléments et attributs imposés avec respect des cardinalités.
3. Donner en XML Schéma la définition du type `ddnType` qui impose à une chaîne de caractères de correspondre au format `jj/mm/aaaa`. On pourra admettre que pour l'examen `jj` varie de 01 à 31 pour tous les mois (mais si vous êtes capable de fournir une expression exacte en fonction du nombre de jours par mois, elle est bienvenue).

4. Donner une grammaire XSD minimale qui utilise tous les éléments et attributs imposés avec respect des cardinalités. On utilisera bien sûr le type `ddnType` que l'on vient de définir.
5. Donner une transformation XSL qui calcule grossièrement l'âge et l'ajoute en tant qu'attribut sachant qu'on utilise 2012 pour comme année de référence (l'âge est donc juste la différence entre 2012 et l'année `aaaa` de la `ddn`).
A défaut d'ajouter l'âge, vous pouvez au moins écrire une transformation XSL qui la calcule et l'affiche en mode texte.
6. Donner une transformation XSL qui liste les personnes par année décroissante et par ordre alphabétique croissant pour une même année avec indication du pseudo entre parenthèses. On produira juste un fragment valide pour XHTML 1.0 strict – pas une page Web complète – défini par les éléments `dl`, `dt` et `dd`. Ainsi le texte produit comportera juste, si vous êtes né(e) en 1957 :

```
<dl>
  <dt>1957</dt>
  <dd>
    Moi (moi).
  </dd>
  <dt>1928</dt>
  <dd>
    Mickey (mic) ; Minnie (min).
  </dd>
</dl>
```

On essaiera de respecter l'indentation et le formatage proposés.