

PROJET XML

Réaliser par :

Ayoub BOULAHYA (boulahya.ayoub@gmail.com / ayoub.boulahya@etu.univ-amu.fr)

XPATH 2.0 :

Les nouvelles possibilités de **XPATH** avec sa **version 2** quand on a utilisé dans le fichier "**xq.txt**" :

- La boucle for \$variable in expression1 where(condition) return expression2
- La clause order by

Exemple:

for \$intervenant in doc("master.xml")/master/intervenants/intervenant order by \$intervenant/nom

Remarques :

- ▶ Le fichier "**xq.txt**" génère le fichier "**xq.html**" dans le répertoire "**www**" à l'aide de langage **XQuery**.
- ▶ Dans le fichier "**java/Noms_unites_enseignement.java**" on a enregistré l'affichage de console dans un fichier texte "**java/Noms_unites_enseignement_out.txt**" comme on a utilisé le produit "**java/saxon9he.jar**" (version Home Edition).
- ▶ Dossier "**ressources**" qui contient les sous-dossiers : "**css**" et "**images**" avec plusieurs sous fichiers qu'on les faire copier dans le répertoire "**www**" avec la commande "**\$ make -p ...**"
- ▶ pour appliquer le schéma XSD pour le fichier "**master_with_entities.xml**" on passera par la création d'un autre fichier qui sera nommé "**master.xml**" à l'aide de la commande :

"xmllin --valid --noent master_with_entities.xml > master.xml"

- ▶ à la sortie de fichier "**tidy_out.txt**" même s'il n'y a pas d'erreur La commande "**tidy**" génère que des Warnings.
- ▶ **NB. Tous les dossiers et les fichiers se génèrent et s'exécutent sans PROBLEMES !**

Travail réalisé :

- La présentation (Makefile, Documentation) ✓
- La feuille CSS ✓
- La DTD ✓
- La feuille XSLT ✓
- Le schema ✓
- La requête XQuery ✓
- Le code Java ✓

- ▶ Pour exécuter toutes les commandes en une seule ligne:

\$ make -f Makefile