

Rapport XML

Résumé conception :

Nous avons fait comme choix de modélisation de regrouper tout notre document sous un élément racine “agence” afin de permettre si besoin dans le futur de rajouter d’autres agences possédant le même type de prestation.

Nous avons décidé d’inscrire les différents “acteurs” (accompagnateurs, enseignants et clients) au niveau de la racine afin de pouvoir leur faire référence à l’aide d’id, ce qui évite la duplication de code ainsi que de se retrouver avec des personnes n’existant pas dans le fichier.

L’agence possède ainsi des séjours qui contiennent des éléments “sejour” offerts par l’agence. Les éléments “sejour” contiennent des éléments “offre” ainsi que diverses informations telles que le prix, les dates de début et fin ainsi que les clients qui y participent.

Les éléments “offre” contiennent des informations sur différents types d’offres proposées telles que la destination, les langues utilisées, les activités, la description ainsi que le type de public ciblé et le “type” de séjour représenté (colonies de vacances, groupes scolaires, immersion en famille, cours linguistiques).

Les activités sont séparées en 3 types : sportives, culturelles et linguistiques. Chaque activité est supervisée par un ou plusieurs accompagnateurs/enseignants selon le besoin.

Nous avons décidé de donner des dates de début et de fin à chaque activité car nous avons supposé qu’une activité peut ne pas durer pendant la totalité du séjour mais seulement pendant une partie (par exemple si un encadrant n’est plus disponible).

Avantages/faiblesses:

Dans les avantages de notre modélisation, nous avons réussi à éviter au maximum d’avoir de la répétition de données, notamment au niveau des “acteurs” grâce aux références id.

On restreint les choix possibles à l’aide d’énumération, par exemple au niveau des types de séjours, des types d’activités ou encore des niveaux de langues...

Notre idée d’associer des dates à chaque activité apporte un peu de réalisme, car lors d’événements, toutes les activités ne sont pas forcément disponibles constamment, il faut donc s’organiser si on souhaite y assister.

Le fait de tout regrouper sous un élément racine “agence” permet de facilement insérer des nouvelles agences ayant le même format.

Dans les faiblesses de notre modélisation, nous pouvons par exemple avoir des langues proposées dans des cours qui ne sont pas contenues dans l'élément "offre". Cela pourrait être possiblement réglé avec des références, mais il faudrait faire attention aux interférences avec les autres offres.

Nous n'avons pas non plus de vérifications dans le cas où un accompagnateur/enseignant serait attribué à 2 activités différentes qui auraient lieu au même moment.

Scénarios:

Scénario 1 :

Le but de notre premier scénario est de récupérer les informations sur des enseignants et des accompagnateurs d'une agence à partir d'un fichier XML . Les informations sont présentées sous forme d'une liste d'informations de contact pour chacun de ces encadrants (enseignants et accompagnateurs) triée par ordre alphabétique par rapport à leur nom de famille et selon leur poste.

Les informations de contact incluent leur nom, prénom, adresse e-mail, numéro de téléphone, et dans le cas des enseignants, leur spécialité.

Scénario 2 (+ implémentation en python) :

Notre deuxième scénario consiste à résumer les statistiques de l'agence sous forme d'un tableau trié par ordre alphabétique selon le type de séjour.

Afin de remplir notre tableau, nous utilisons une clé nommée "sejourParType" pour regrouper les séjours par type et compter le nombre de clients pour chaque type de séjour. Nous calculons également le nombre total de clients parmi tous les séjours et le nombre moyen de clients par type de séjour.

Nous avons également écrit un code python appelé bonus.py qui s'occupe d'effectuer cette transformation de la même façon que la feuille de style XSLT et de transformer notre fichier xml en le fichier html attendu.

Scénario 3 :

Ce scénario XSLT a pour but d'afficher toutes les activités proposées pour chaque offre de chaque séjour de l'agence de voyage. Afin de faciliter l'implémentation, nous avons décidé d'utiliser la récursion avec les templates.

Le template principal est le match="/" qui correspond à la racine du document XML. Il crée une page HTML et appelle le template "afficherActivites" qui affiche les détails de chaque activité pour chaque offre de chaque séjour.

Le template "afficherActivites" est appelé récursivement pour afficher chaque activité et pour parcourir toutes les activités proposées pour chaque offre de chaque séjour.

Une fois la récursion terminée, nous récupérons une liste de toutes les activités proposées pour chaque offre de chaque séjour, ainsi que le nom de l'offre et le numéro du séjour correspondant.

Scénario 4 :

Le but de ce scénario est de filtrer tous les séjours qui ont une offre disponible pour une langue spécifique dans une période de temps donnée, et de les présenter sous une forme légèrement différente.

Plus précisément, la feuille de style prend en entrée un document XML qui décrit une agence de voyage et ses offres de séjour, et filtre les séjours selon les critères suivants :

- La date de début du séjour est supérieure ou égale à la variable \$dateDebut.
- La date de fin du séjour est inférieure ou égale à la variable \$dateFin.
- L'offre du séjour a une langue disponible qui correspond à la variable \$langue.

Les séjours filtrés sont ensuite présentés sous la forme d'un nouveau document XML, dans lequel chaque séjour est présenté avec son type de séjour, son type de public, le nom de l'offre, sa description, la destination, le prix, la date de début et la date de fin. Le document de sortie inclut également le nom de l'agence de voyage en tant qu'attribut de l'élément racine "agence".

Scénario 5 :

Le but de ce code est de transformer un document XML contenant des informations sur les agences de voyage et les séjours qu'elles proposent en un document texte, en suivant le format JSON.

Le modèle principal de transformation est la balise agence, qui contient des informations sur une agence de voyage. Les séjours proposés par cette agence sont représentés par la balise sejours qui contient les balises dateDebut, dateFin, clients et activites. Les balises activites, activites_sportive, activites_culturelle et activites_linguistique contiennent des informations sur les activités proposées dans chaque séjour. La balise cours contient des informations sur les cours proposés dans les séjours linguistiques. Les balises client, enseignant et accompagnateur contiennent les prénoms des personnes impliquées dans chaque séjour.

Le résultat de la transformation sera un document texte au format JSON qui contient des informations sur les agences de voyage et les séjours qu'elles proposent de façon synthétique.