

Les bibliothèques de base contiennent les actions de base d'une application web et donc les plus utilisées. Elles sont divisées en quatre catégories: la gestion des variables de scope, les actions conditionnelles, les itérations et les URLs.

a. Gestion des variables de scope:

Ces balises permettent d'afficher, créer, modifier et supprimer des variables en plus de gérer les exceptions.

- `<c:out/>`: Permet d'afficher une expression après l'avoir évalué tout en protégeant nos variables contre la faille XSS en protégeant les caractères spéciaux HTML.
- `<c:set/>`: Permet de définir une variable dans une portée particulière (page, requête, session ou application). Elle permet aussi de modifier la valeur d'une propriété d'un bean.
- `<c:remove/>`: Permet de supprimer une variable de scope indiquée.
- `<c:catch/>`: Permet d'intercepter les exceptions qui peuvent être générées par son corps.

b. Actions conditionnelles:

Ces balises permettent d'effectuer des tests conditionnels de la même manière que les mots-clés `if` ou `switch` du langage Java.

- `<c:if/>`: Permet d'effectuer un traitement conditionnel de la même manière que `if` dans Java.
- `<c:choose/>`: Permet un traitement conditionnel de la même manière que `switch` dans Java. C'est à dire plusieurs possibilités dont une seule sera réalisée et évaluée. Elle exécute le corps du premier tag `<c:when/>` dont la condition est vérifiée.
- `<c:when/>`: Permet de définir une option de l'action `<c:choose/>`. Elle doit impérativement avoir la balise `<c:choose/>` comme parent.
- `<c:otherwise/>`: Permet de définir le traitement à exécuter si aucun tag `<c:when/>` n'a été évalué. Elle doit impérativement avoir la balise `<c:choose/>` comme parent et doit être placée après la dernière balise `<c:when/>`. Elle est équivalente au mot-clé `default` dans `switch` dans Java.

c. Itérations:

Ces balises permettent d'effectuer des boucles de la même manière que les mots-clés `for` ou `while` en Java.

Les balises d'itération peuvent avoir 5 attributs communs: `var`, `varStatus`, `begin`, `end` et `step`.

- `<c:forEach/>`: Permet d'effectuer des itérations sur un ensemble de collections de données.

- `<c:forTokens/>`: Permet de découper une chaîne de caractères en plusieurs segments selon un certain délimiteur et ensuite, d'itérer sur ces segments.

d. URLs:

Ces balises permettent de gérer les URLs.

- `<c:param/>`: Permet d'ajouter un paramètre à notre URL. Elle doit avoir comme balise parent (direct ou indirect) une balise `<c:url/>`, `<c:import/>` ou `<c:redirect/>`.
- `<c:url/>`: Permet de créer des URLs absolues, relatives au contexte, ou relatives à un autre contexte.
- `<c:redirect/>`: Permet de rediriger le client vers d'autres pages en envoyant une commande de redirection HTTP
- `<c:import/>`: Permet d'importer une ressource selon son url. Cette ressource peut appartenir à un autre contexte ou même à un autre serveur.

4. Librairies XML:

Cette librairie permet d'utiliser et traiter des fichiers XML dans une page JSP.

a. XPath:

XPath est un langage de formulation d'expression par rapport à un fichier XML et donc il est utilisé pour accéder aux données des pages XML. C'est une recommandation de W3C qui fournit sa documentation.

b. Actions XML de Base:

L'analyse de fichiers XML et l'accès à ses variables en utilisant des expressions du langage XPath.

- `<x:parse/>`: Permet d'analyser un document XML.
- `<x:out/>`: Permet d'évaluer et d'afficher une expression XPath.
- `<x:set/>`: Permet de définir une variable XPath dans une portée particulière (page, requête, session ou application).

c. Actions de contrôle XML:

Ces balises permettent d'effectuer des actions de contrôle de la même manière que ceux de la librairie standards, sauf qu'ils appliquent une action XPath sur un document XML.

d. Transformation XSLT:

XSLT (eXtensible Stylesheet Language Transformations) est un langage de transformation XML, défini par W3C. Il permet de transformer un document XML en un autre format (PDF, HTML, ...). Il utilise XPath pour désigner une partie d'un arbre XML.

- `<x:transform/>`: Permet d'appliquer une transformation XSLT sur un document XML.
- `<x:param/>`: Permet d'ajouter un paramètre à une transformation XSLT et doit avoir comme balise parente `<x:transform/>`