Les variables en JSP

1. Objectifs

- Etre capable de déclarer, de manipuler des variables en JSP

2. Déclaration des variables

- Une application, un programme, quels qu'ils soient, manipulent des données (valeurs numéri-ques, textes, images...).
- Les données sont enregistrées dans la mémoire de l'ordinateur. Chaque valeur est repérable par un nom et une zone mémoire.
- Le nom est déterminé par le programmeur et la zone mémoire par l'ordinateur. Ces opérations sont effectuées au moment de la déclaration des variables.
- **La syntaxe de déclaration d'une variable est la suivante :**
- **<%! type nomDeLavariable ; %>**
- **Exemple**:

```
<%!int valeur = 3; %> <%! double reel = 5.3, imaginaire; %> <%! String texte = "arbre";
%> <%! java.util.Random hasard = new java.util.Random(); %>
```

  - Dans cet exemple, la variable valeur est définie comme étant de type `int` et les variable `reel` et imaginaire comme étant de type `double`.
  - La variable **`valeur`** est initialisée à 3.
  - La variable **`reel`** est initialisée à 5.3.
  - La variable **`imaginaire`** n'étant pas initialisée directement, le moteur JSP va y placer de lui-même la valeur o.o.

> ***Pour déclarer plusieurs variables de même type, il suffit de placer une virgule entre chaque nom de variable.***

  - Les deux autres variables nommées **`texte et hasard`** sont respectivement de type `String` et java.util.Random.
  - Il ne s'agit pas tout à fait de variables puisque ces espaces mémoire Permettent de stocker des données complexes telles que des suites de caractères ou des valeurs aléatoire calculées directement par JSP. Ces variables sont en réalité appelées objet.

apcpedagogie.com

3. Les noms de variables

- Une déclaration de variable se fait comme ceci :
- **<Type de la variable> <Nom de la variable> ;**
- Cette opération se termine toujours par un point-virgule ("; "). Ensuite, on l'initialise en entrant une valeur.
- Le choix des noms de variables n'est pas limité. Il est toutefois recommandé d'utiliser des noms évocateurs. Par exemple, les noms des variables utilisées dans une application qui gère les codes-barres de produits vendus en magasin sont plus certainement `article`, `prix`, `codebarre` que xyz1, xyz2, xyz3.
- Les premiers, en effet, évoquent mieux l'information stockée que les seconds.
- Les contraintes suivantes sont à respecter dans l'écriture des noms de variables :
  - Le premier caractère d'une variable doit obligatoirement ne pas être un chiffre.
  - Aucun espace ne peut figurer dans un nom.
  - Les majuscules sont différentes des minuscules, et tout nom de variable possédant une majuscule est différent du même nom écrit en minuscules.
  - Les caractères &, -, ", #, ' , {, }, (,) , [,], -, I , - , \, ' 9- -' * -, 0, . , , • , , , .,i, 1 , <, >, L, ainsi que ; et , ne peuvent être utilisés dans l'écriture d'un nom de variable.
  - Tout autre caractère peut être utilisé, y compris les caractères accentués, le caractère de soulignement (_) et les caractères $, € et ù.
  - Un mot-clé du langage (`if`, `String`, `int`...) ne peut être utilisé comme nom de variable.
- Le nombre de lettres composant le nom d'une variable est indéfini. Néanmoins, l'objectif d'un nom de variable étant de renseigner le programmeur sur le contenu de la variable, il n'est pas courant de rencontrer des noms de variables de plus de trente lettres.

4. Les noms réservés

- Les noms de variables ne peuvent pas être les noms suivants (qui sont des noms réservés):
  - abstract
  - boolean, break, byte et byvalue
  - case, cast, catch, char, class, const et continue
  - default, do et double
  - else et extends
  - false, final, finally, float, for et future
  - generic et goto
  - if,implements, import, inner, instanceof, int et interface

apcpedagogie.com

- long
- native, new et null
- operator et outer
- package, private, protected et public
- return, rest et return
- short, static, sctrictfp, super, switch et synchronized
- this, throw, throws, transient, true et try
- var, void et volatile
- while et widefp

5. Affectation d'une donnée à une variable

- L'affectation est le mécanisme qui permet de placer une valeur dans un emplacement mémoire
- Pour stocker une donnée dans une variable que l'on a initialisée, il faut faire une affectation, c'est-à-dire préciser la donnée qui va être stockée à l'emplacement mémoire qui a été réservé lors de l'initialisation.
- `<% variable = valeur; %>`
- Pour cela on utilise l'opérateur d'affectation "=":

```
Nom_de_la_variable = donnée;
```

6. Application

1. **Énoncé**

   - Parmi les, variables suivantes, quelles sont celles dont le nom n'est pas autorisé, et pourquoi ?
     - Compte, pourquoi#pas, Num_2, -plus, Undeux, 2001espace, ©adresse, VALEUR temporaire, ah!ah! , Val$solde, double.

2. **Correction**

   - Les noms de variables suivants ne sont pas autorisés : `pourquoi#pas , -plus, ©adresse,VALEUR_temporaire, ah!ah!` car les caractères #, - , @ et ! sont interdits.
   - `2001espace`: car il n'est pas possible de placer un chiffre en début de variable.
   - `double` car double est un mot-clé du langage.
   - Les noms de variables suivants sont autorisés : Compte, Num_2 (" _") et non pas temporaire, Val$solde. ("-"), Undeux (et non pas un deux), VALEUR_temporaire et Val$solde

Source:

- **Livre:** Initiation à JSP avec 50 exercices coriigés Anne Tasso Sébastien Ermacore

---

Sommaire du cours JSP

Source : https://apcpedagogie.com/les-variables-en-jsp · téléchargé le 06/11/2026 · © APC Pédagogie · apcpedagogie.com

apcpedagogie.com