|  |
| --- |
| C:\Users\Thomas\EPSI\Etude Systémique\Logo_EPSI.png |
| **Normes de codage** |
| Marianne et les 5 Fantastiques |
|  |
| Mlle. DEMOULIN Marianne Mlle. STALTER Marianne M. MEURILLON Grégoire M. JAGOURY Matthieu M. DESZCZ Sébastien M. MAURICE Rémy M. DEBAS Thomas |
| **Années 2013-2014**  **B3 Initial** |

|  |
| --- |
|  |

***Normes de codage***

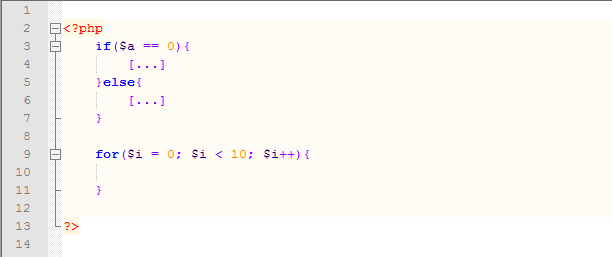
**I – Organisation du code**

1. ***Accolades***

Les accolades doivent être situées à la fin d’une ligne commençant par « if », « while », « for »…

En revanche, les accolades de fin doivent être sur une ligne spécifique. Dans le cas d’un « if », s’il y a un « else », l’accolade de fin de « if » se trouve juste avant le « else ».

Exemple :



1. ***Indentation***

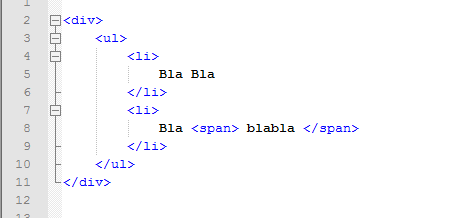
L’indentation doit être parfaitement respectée pour que le code soit agréable à lire.

Une indentation correspond à un Tab, et non pas à des espaces, qui prennent plus de place en mémoire.

Après l’accolade de chaque « if », « while », « for », « do…while », et après toutes les balises HTML ayant un contenu, « <div> », « <a> », « <ul> », il faut obligatoirement passer une ligne, et décaler l’indentation de 1 vers la droite.

Seuls quelques exceptions autorisent la non-indentation : « <span> », qui est une balise in-line, et les titres « <h1> », « <h2> », … qui ne prennent pas de place… Dans ce cas, le texte est placé directement entre les deux balises, sur la même ligne.

Exemple :



1. ***Commentaires***

Chaque fonction doit être commentée. De même, chaque grande partie de code doit être commentée.

On privilégiera la forme : « /\* …. \*/ », qui peut s’étaler sur plusieurs lignes.

En revanche, il est tout de même possible de faire un commentaire de type « // » en fin de ligne, notamment pour expliquer ce à quoi correspond une variable.

**II – Normes de codage**

Tous les noms de classes, variables, méthodes, … doivent être écrits en ANGLAIS.

Aucun nom ne doit contenir de caractères spéciaux.

1. ***Classes***

Le nom de la classe doit toujours commencer par une majuscule.

S’il y a plusieurs mots dans le nom de la classe, chaque première lettre des mots doit être écrite en majuscule.

Exemples :

maClasse MaClasse maclasse Maclasse

En règle générale, le nom de la classe doit aussi correspondre au nom du fichier. Elle doit aussi être le plus explicite possible.

1. ***Variables***

Le nom d’une variable doit toujours commencer par une minuscule.

S’il y a plusieurs mots dans le nom de cette variable, chaque première lettre des mots doit être écrite en majuscule.

Exemples :

maVariable MaVariable

Le nom de la variable doit être explicite, on doit savoir tout de suite à quoi elle sert.

L’utilisation d’une variable à un caractère (exemple : a, i, …) est exclusivement réservée aux boucles.

1. ***Constantes***

Les noms de constantes doivent toujours être écrits intégralement en majuscule.

Les mots d’une constante doivent être séparés par un UNDERSCORE (\_).

Exemple : MA\_CONSTANTE

1. ***Bases de données***

Tous les noms de table doivent commencer par « m5f\_ ».

Exemple :

La table utilisateur doit se nommer :

« mf5\_utilisateurs ».