

Convention de codage

Les utilisations de bibliothèques externes

L'utilisation de bibliothèques externes doit être limitée au maximum. Le but étant de réduire au mieux la dépendance du projet à des projets externes dont la pérennité n'est pas assurée. De plus, utiliser une bibliothèque revient souvent à utiliser bêtement des méthodes dont on ignore le fonctionnement, on s'éloigne donc du cadre pédagogique d'une formation.

Les imports

Les imports au début des classes ne doivent pas contenir de * (exemple, pas de « import java.awt.* ; »), nous n'importons que ce dont on a besoin. NetBeans simplifie grandement ce comportement.

Les déclarations de méthode

L'ordre des mots clés: visibilité → qualificatif → type de retour → nom de la méthode. Exemple:

```
public synchronized void setTruc(int machin) { }
```

Conventions de nommage des variables

Une dérivée de la convention de nommage hongroise consiste à précéder le nom des variables de classe par un underscore (_).

Ceci permet deux choses:

- Dans un constructeur, le paramètre appelé, par exemple cpt pourra initialiser le membre de classe _cpt, plus besoin d'utiliser le this ici. (exemple this.cpt = cpt devient _cpt = cpt;)
- Dans le code, il permet de distinguer très facilement les variables de la méthode et celles de classe et éviter de se tromper à cause d'un oubli du mot clé this

De plus, pour les noms de variable composés, au lieu d'utiliser le underscore, on utilisera des majuscules au début de chaque mot, exemple:

nombre_de_poney devient nombreDePoney

Les variables static ou constant sont à écrire en majuscule, précédées elles-aussi par un underscore.

Le constructeur de classe

Le constructeur de classe ne doit qu'uniquement construire un objet de la classe. Aucun appel de méthode

d'exécution de code, celle-ci devra être faite via l'objet: `objet.calculTruc()`;

De plus, le code du constructeur doit être aussi peu dense que possible: par l'utilisation de méthodes d'initialisations des composants/données.