

HERITAGE, POLYMORPHISME

1 Objectifs du TD

- 1) Comprendre la différence entre une fonction virtuelle, virtuelle pure ou non-virtuelle.
- 2) Comprendre la notion d'interface en C++.
- 3) Comprendre l'intérêt de l'utilisation du polymorphisme.

EXERCICE 1 – Ecriture/Utilisation d'une interface en C++

- 1) Déclarez dans un fichier `DocumentBuilder.h` une interface `DocumentBuilder` avec les méthodes suivantes :
 - `void beginDocument()`
 - `void endDocument()`
 - `void beginParagraph()`
 - `void endParagraph()`
 - `void addText(const string &)`
 - `void newLine();`
- 2) Ecrivez une fonction (dans le fichier `main.cpp`) ayant la signature suivante : `void createDocument(DocumentBuilder *)` qui fabrique un document composé de deux paragraphes chacun d'eux composés de deux textes espacés par un saut de ligne.
- 3) Compiler et linker votre programme en écrivant une fonction `main` vide. Expliquez pourquoi il n'y a aucun problème de compilation.
- 4) Essayez maintenant d'utiliser la fonction `createDocument` dans votre fonction `main` en essayant d'instancier la classe `DocumentBuilder`. Commentez les erreurs données par le compilateur.

EXERCICE 2 – Implémentation de la classe `TxtBuilder`

- 1) Déclarez et définissez dans les fichiers `TxtBuilder.h` et `TxtBuilder.cpp` une classe `TxtBuilder` qui implémente l'interface `DocumentBuilder`.
- 2) Ajoutez la méthode `getDocument` à la classe `TxtBuilder` qui permet de récupérer la chaîne de caractères correspondant au document produit.
- 3) Modifier votre `main` pour instancier un objet de type `TxtBuilder` et pour le passer à la fonction `createDocument`. Faites afficher le résultat à l'écran.

EXERCICE 3 – Implémentation de la classe `HtmlBuilder`

- 1) Déclarez et définissez dans les fichiers `HtmlBuilder.h` et `HtmlBuilder.cpp` une classe `HtmlBuilder` qui implémente l'interface `DocumentBuilder`.
- 2) Ajoutez la méthode `getDocument` à la classe `HtmlBuilder` qui permet de récupérer la chaîne de caractères correspondant au document produit.
- 3) Modifier votre `main` pour instancier un objet de type `HtmlBuilder` et pour le passer à la fonction `createDocument`. Faites afficher le résultat à l'écran.

EXERCICE 4 – Utilisation du polymorphisme

- 1) Modifiez votre programme `main` pour afficher le texte en version `Html` ou en `Txt` en fonction du paramètre passé à votre programme.
- 2) Que doit-on modifier dans notre programme si nous voulons ajouter une nouvelle sortie (par exemple `word-xml`) ?