

Les musées de campagne

Travaux dirigés et pratiques - Programmation par objets 2 (UE ULIN606)

Objets de musée

Un musée de campagne possède des objets variés. On veut pouvoir stocker pour chacun de ces objets un descriptif (texte libre qui peut être vu comme une chaîne de caractères), et sa référence, nombre entier qui l'identifie de façon unique.

Parmi ces objets, certains sont de véritables objets d'art signés ; pour ceux-là, on veut stocker aussi le nom de l'auteur.

Un musée de campagne possède aussi, hélas, des objets légués par d'honorables citoyens et qu'il n'a pas pu refuser : renard empaillé, portrait du généreux donateur, etc.

Pour ces objets regrettables, on veut pouvoir stocker en plus le nom du donateur et l'année du don.

1. Quelles classes imaginez-vous pour représenter tous ces objets ?
2. Écrivez les fichiers `.h` et `.cc` correspondants, en supposant que chacune de ces classes possède :
 - des attributs privés ;
 - un constructeur sans paramètres qui initialise tous les attributs à des valeurs par défaut ;
 - un constructeur avec paramètres qui remplit tous les attributs avec les valeurs données ;
 - des accesseurs de consultation et de modification pour chaque attribut ;
 - une méthode *saisie* qui saisit les valeurs des attributs sur un flot d'entrée ;
 - une méthode *affiche* qui affiche les valeurs des attributs sur un flot de sortie.
3. Écrivez un *main* qui, pour chacune des classes, crée et affiche deux objets : un directement (déclaration de variable), et l'autre à travers un pointeur.

Salles d'exposition

Le musée possède des salles d'exposition d'une certaine capacité (capacité = nombre maximum d'objets qu'on peut mettre en exposition dans la salle).

1. Écrivez une classe qui représente une salle, en respectant les caractéristiques suivantes :
 - une salle est créée vide, avec une certaine capacité (donnée, ou 10 par défaut) ;
 - on la remplit en ajoutant un objet donné à une place donnée ; la place est représentée par un numéro ; les objets peuvent appartenir à n'importe laquelle des classes précédentes, mais on veut une seule méthode *ajoute*, la même quel que soit le type de l'objet ;
 - on peut enlever un objet : on donne le numéro de place et on récupère l'objet ;
 - on veut pouvoir afficher le contenu de la salle ;
 - on veut pouvoir connaître le nombre d'objets présents dans la salle.
 - information technique : on est sûr de ne jamais faire de copie d'une salle, on demande donc que la classe ne contienne pas de constructeur par copie.
2. Faites un *main* pour tester le fonctionnement.