

Classes, instances, associations et collections

Une école primaire souhaite développer un logiciel permettant la mise en commun des questions de lecture et des fiches de lecture créées par les professeurs des écoles. Ce logiciel permet de rechercher des questions ou des fiches de lecture pour un livre donné, de saisir de nouvelles questions pour un livre, et de composer des fiches de lecture, qui sont sauvegardées dans le système, et qui peuvent être exportées en format pdf.

1 Modélisation

Des éléments de modélisation vous sont fournis sous forme du diagramme de classes partiel de la figure 1. Une question est forcément contenue dans au moins une fiche de lecture.

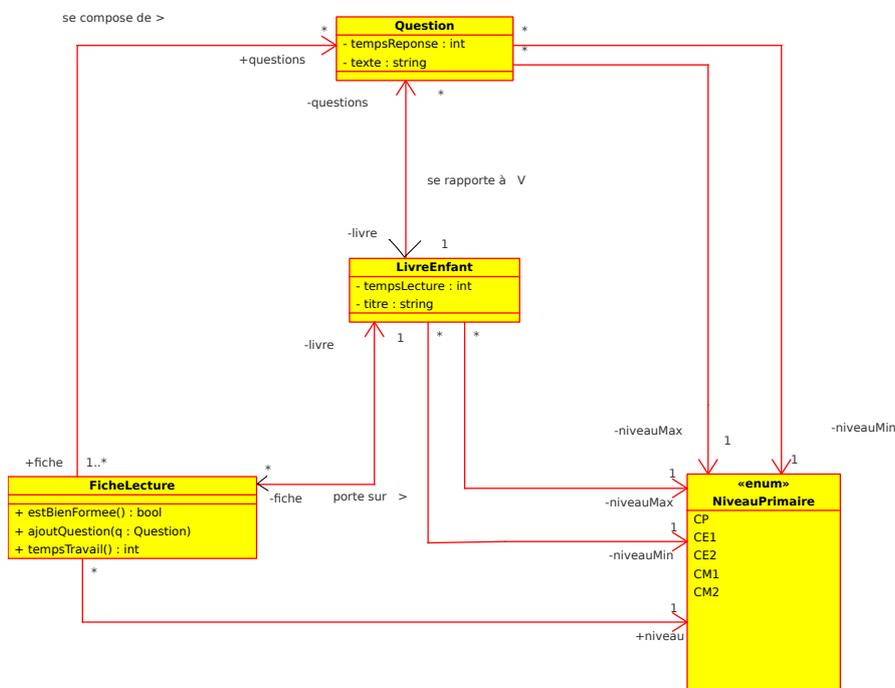


FIGURE 1 – Diagramme de classes partiel

Question 1. Complétez le diagramme de classes de la figure 1 de manière à représenter le fait que chaque question a une forme, qui est soit un QCM, soit une réponse à texte libre, soit un texte à compléter.

Question 2. Complétez le diagramme de classes de la figure 1 de manière à représenter le système de mise en commun des fiches de lecture de l'école. Ce système de fiches gère des questions et des fiches de lecture pré-définies. Il permet la création d'une nouvelle fiche de lecture pour un livre donné, de lister les questions disponibles pour un livre donné, et de lister les fiches de lecture disponibles.

Question 3. Complétez le diagramme de classes de la figure 1 en ajoutant à la classe `LivreEnfant` et à la classe `Question` les constructeurs qui vous paraîtront pertinents (un constructeur par classe).

Question 4. Représentez sous forme de diagramme d'objets une fiche de lecture de niveau CP, pour le livre "Lulu Vroumette", de niveau CP-CE1, et de temps de lecture 10 minutes. La fiche a deux questions de type réponse à texte libre, et de temps de réponse une minute, de niveau CP, et de texte respectivement "Quel est le nom de l'auteur?" et "Quel est le nom de l'illustrateur?".

2 Programmation en Java

Dans la suite de l'énoncé, on suppose disposer sans les écrire au préalable de tous les accesseurs (en lecture et en écriture) pour les attributs et les rôles navigables des associations, et on utilise le langage Java.

Question 5. On choisit de stocker les questions d'une fiche de lecture dans un `Vector`. Donnez la déclaration d'un tel `Vector` dans la classe `FicheLecture`.

Question 6. Donnez l'implémentation de la méthode `ajoutQuestion` de la classe `FicheLecture` qui prend en paramètre une question, et l'ajoute à la fiche de lecture à condition que la question se rapporte au même livre que la fiche de lecture et que la même question n'existe pas déjà (au sens de l'égalité de valeur) dans la fiche de lecture. Si cette condition n'est pas vérifiée, cette méthode ne fait qu'afficher un message d'erreur. On pourra utiliser la méthode `contains` de la classe `Vector` qui retourne vrai si et seulement si le `Vector` contient l'élément passé en paramètre (au sens de l'égalité de valeur).

Question 7. Donnez l'implémentation de la méthode `tempsTravail` de la classe `FicheLecture` qui calcule le temps que mettra un élève à réaliser la fiche de lecture. Ce temps est la somme du temps de lecture du livre et des réponses aux questions de la fiche, majorée de 10% pour le temps de lecture des questions et de relecture des réponses. La majoration sera arrondie à la minute supérieure. On pourra utiliser la méthode de la classe `Math` : `static double ceil(double a)` qui est définie comme suit : *Returns the smallest (closest to negative infinity) double value that is greater than or equal to the argument and is equal to a mathematical integer.* On notera que l'on peut comparer les littéraux d'énumération avec les opérateurs `<` et `>`.

Question 8. Donnez l'implémentation de la méthode `estBienFormée` de la classe `FicheLecture` qui renvoie vrai si et seulement si elle correspond à un temps de travail par l'élève inférieur à 45 minutes, et que son niveau scolaire est inclus dans la plage de niveaux scolaires de toutes ses questions.

Question 9. Donnez l'implémentation d'une méthode `main` qui crée la fiche de lecture décrite à la question 5. On pourra utiliser sans les définir les constructeurs introduits à la question 4. On suppose disposer d'un constructeur pour `FicheLecture` qui prend en paramètre un livre.

Question 10. Terminez l'implémentation en Java et testez-la.