



Module : Programmation Orientée Object (POO)

Enseignant : B. Khaldi

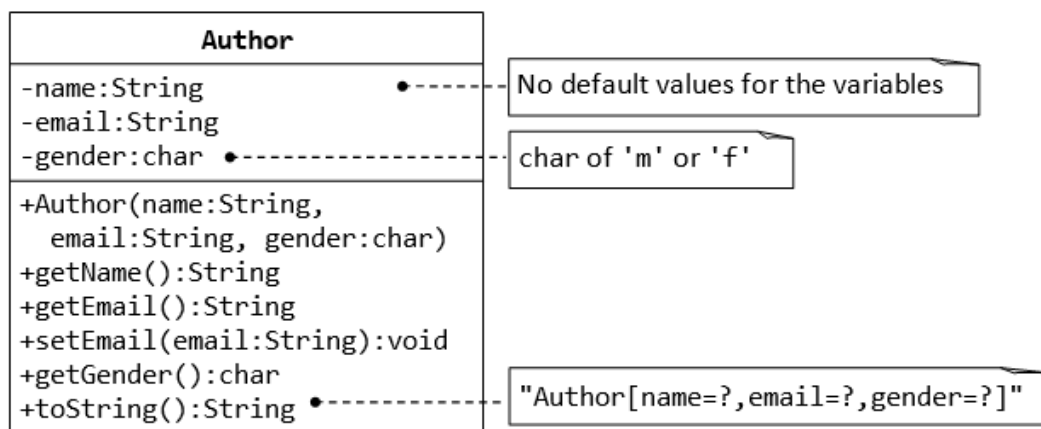
Année universitaire : 2018/2019

Date : / /2018.

## TD n° : 2

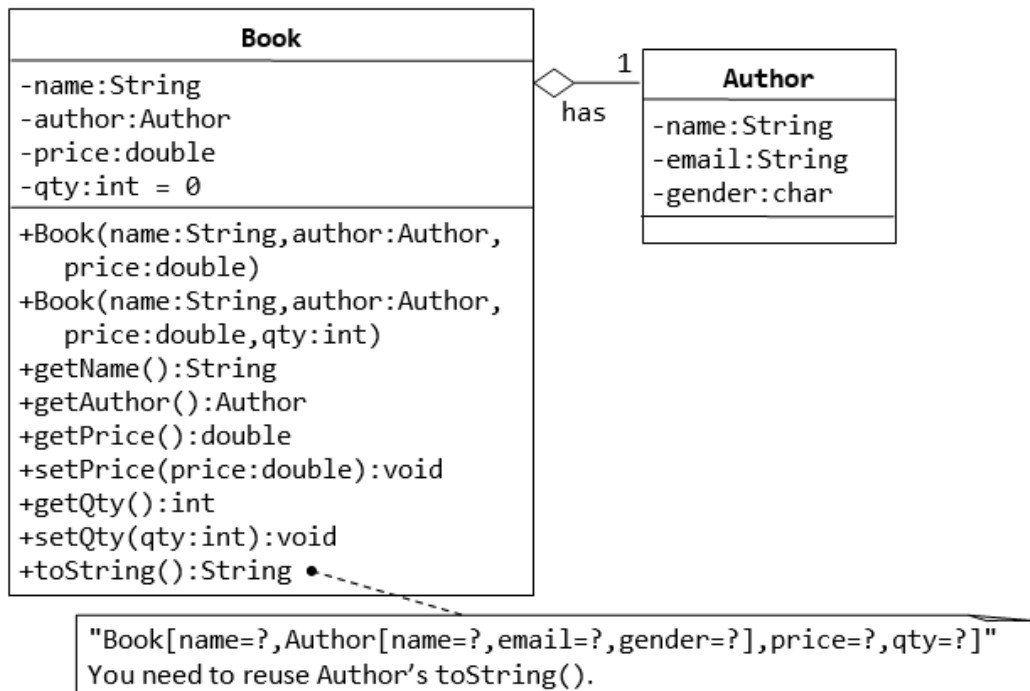
Cette série vise à mettre en œuvre le concept de composition en POO.

**Exercice 1.** Une classe appelée **Author** (illustrée dans le diagramme suivant) est conçue pour modéliser l'auteur d'un livre. Ecrivez la classe **Author**.



1. Écrivez également une pilote de test appelée **TestAuthor** pour tester toutes les méthodes publiques de la class **Author**.

**Exercice 2.** Une classe appelée **Book** est conçue (comme indiqué dans le diagramme suivant) pour modéliser un livre écrit par un auteur. Ecrivez la classe **Book**.



1. Ecrivez un pilote de test appelé **TestBook** pour tester toutes les méthodes publiques de la classe **Book**. Prenez en considération que vous devez construire une instance de **Author** avant de pouvoir construire une instance de **Book**.
2. Essayer :
  - a. Impression du nom et de l'e-mail de l'auteur à partir d'une instance du livre.
  - b. Introduction de nouvelles méthodes appelées **getAuthorName ()**, **getAuthorEmail ()**, **getAuthorGender ()** dans la classe **Book** pour renvoyer le nom, l'e-mail et le genre de l'auteur du livre.

**Exercice 3.** Nous avons considéré qu'un livre est écrit par un et un seul auteur. En réalité, un livre peut être écrit par un ou plusieurs auteurs.

1. Modifiez la classe **Book** pour prendre en charge un ou plusieurs auteurs.
2. Ecrivez un pilote de test (appelé **TestBook**) pour tester la nouvelle classe **Book**.