

# Guide du tuteur – Projet QCM - Fiche S5B

## 1 Introduction

Afin de faciliter l'implémentation de votre projet QCM, nous mettons à votre disposition une série de classes Java qui vous permettront d'effectuer facilement des opérations pour la manipulation d'un QCM, comme par exemple la lecture d'un fichier texte contenant la description des questions et choix possibles. Vous disposez sur icampus de la librairie qcm.jar (à ajouter dans le classpath du projet) et la javadoc de celle-ci. Il est important de consulter la javadoc car vous n'avez pas les sources de ces classes utilitaires.

Nous fournissons également le code source de `AssessmentFormConsole.java` qui est une classe application (avec une méthode `main`) illustrant la manipulation des concepts et outils dont vous aurez besoin. Vous pouvez exécuter cette classe, c'est le point d'entrée de l'application.

## 2 Description des classes

Un objet `AssessmentForm` contient tout ce qu'il faut pour permettre la présentation d'un QCM à l'utilisateur. Schématiquement, il consiste en un tableau d'objets `Question`. Les questions elles-mêmes contiennent un tableau d'objets `Choice` (choix). Ceci est représenté à la Figure 1.

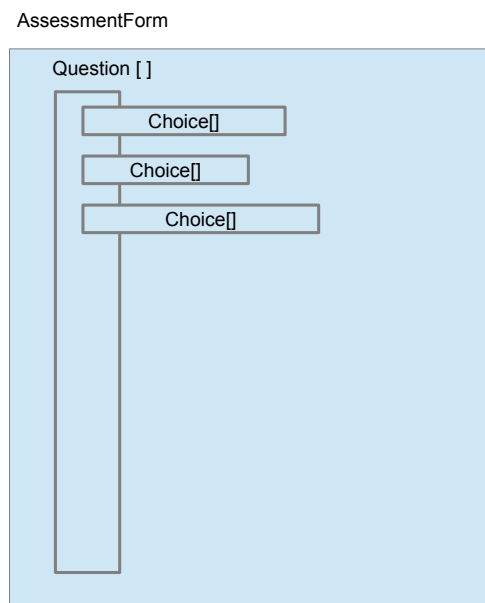


FIGURE 1 – Vue schématisée d'un objet `AssessmentForm`

### 2.1 mcq.AssessmentForm

Ceux-ci sont construits au départ d'un fichier texte respectant une structure bien particulière (voir l'exemple de fichier `QCM.txt` fournit). Il est important de noter que la méthode `getQuestions` permet

de récupérer le tableau de questions.

## **2.2 mcq.Question**

Cette classe représente une question. Il est possible d'obtenir l'énoncé de la question avec `getWording()` ainsi que le tableau des choix possibles avec `getChoices()`.

## **2.3 mcq.Choice**

Cette classe représente un choix à une question. Un `String` qui peut être obtenu avec `getValue()` représente la description du choix. On peut aussi tester la véracité d'un choix avec `isCorrect()`.