

## Exercices sur le patron « Méthode template »

### Exercice 1

On veut créer un cadriciel (framework) dans le but d'imprimer des rapports à partir de fichiers XML comme le suivant.

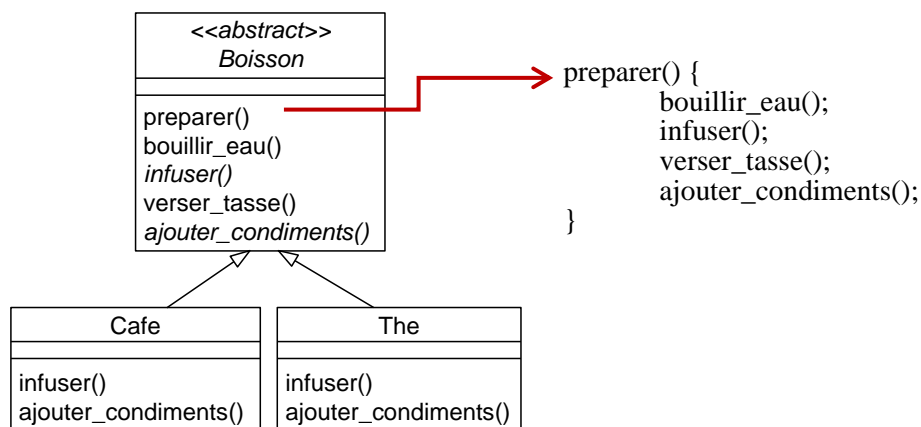
```
<?xml version="1.0" encoding="UTF-8"?>
<document>
  <entete>Rapport de tests</entete>
  <bas_de_page>2017-10-16</bas_de_page>
  <pagination statut="oui" depart="1" position="bas"></pagination>
  <corps>
    Un texte très long peut être écrit ici.
  </corps>
</document>
```

L'entête doit être imprimé au début de chaque page. Le bas de page doit être imprimé à la fin de chaque page. La pagination, lorsqu'elle est demandée, doit être imprimée à chaque page. Le corps du texte, quant à lui est imprimé sur plusieurs pages au besoin. Le cadriciel doit laisser à l'utilisateur la liberté de redéfinir les polices pour chaque élément, de mettre les caractères en gras, en italique, etc.

- 1) Proposez une conception basée sur le patron méthode template. Dessinez le diagramme de classe de votre solution en y incluant une classe concrète qui effectue une impression simple.

### Exercice 2

Dans l'exemple présenté en classe de l'application du patron « Méthode template », j'aimerais rendre optionnelles certaines étapes de mon algorithme de préparation de boisson. Par exemple, j'aimerais rendre l'étape `ajouter_condiments()` optionnelle pour certaines boissons.



Comment faire ça, sachant que je veux garder la méthode `preparer()` comme méthode template?