



Interface – Refactoring – Sensibilisation à la séparation des responsabilités

❶ Rappel : Utilisation de Git

Récupération du dépôt distant

A chaque fois que vous commencez à travailler chez vous ou à l'université, dans votre explorateur de fichier cliquer droit sous votre dossier 'LesGaulois' puis ouvrir 'Git Bash Here'.

En ligne de commande tapez la commande 'git pull'.

Historisation périodique de votre projet

Rappel : lorsque vous avez fini une question du sujet de TP (ce qui vous a amené à créer ou modifier plusieurs classes), vous devez procéder à l'historisation de votre dépôt local et la mise à jour du dépôt distant.

Dans votre explorateur de fichier cliquer droit sous votre dossier 'LesGaulois' puis ouvrir 'Git Bash Here' . Il faut réaliser trois commandes successives :

- Enregistrement des modifications à historiser dans l'index au moyen de la commande 'git add .' (n'oubliez pas le point)
- Enregistrement dans le dépôt local des fichiers mis à l'index avec la commande 'git commit -m <Intitulé des modifications>'
Exemple : `git commit -m "TP1 fin de l'exercice 1"`
De manière plus professionnelle il faudrait donner la nature des modifications effectuées.
- Mise à jour du dépôt distant avec la commande 'git push'¹

¹ A la première utilisation de la commande, une authentification par le navigateur peut être demandée. Complétez cette identification dans votre navigateur et revenez sur votre terminal continuer votre processus Git.



② Un conteur pour nos histoires

Un conteur est défini par son nom. Lorsque l'on veut qu'il nous conte une histoire on lui transmet une bataille. Cette bataille sera toujours décrite de cette manière :

- Décrire le contexte
- Choisir les combattants
- Préparer le combat,
- Décrire le combat,
- Donner le résultat.

Les batailles pourront être très différentes, il peut s'agir du forgeron et du poissonnier d'un village gaulois qui se frappe au sujet de la fraîcheur du poisson vendu, comme d'un groupe de gaulois affrontant un bataillon de soldat ou encore une embuscade préparer par des soldats désireux d'être promus...

1. Faire le diagramme de classes, et le faire valider à l'enseignant.
2. Implémenter votre conception dans un nouveau package batailles.
3. Historiser votre code.

③ Des soldats sournois : l'embuscade

Contexte : la classe embuscade sera construite comme une bataille et se déroulera au cœur de la forêt. Deux soldats tendront une embuscade à trois gaulois. Le nombre de protagonistes sera fixe mais pas les individus. Les soldats se seront préparés en portant tous leurs équipements. Les gaulois faisant une simple promenade n'auront pas anticipé la bataille et n'auront pas bu de potion magique.

Mises en garde : au fur et à mesure que nous allons développer les cinq méthodes de description des batailles nous allons nous rendre compte que d'avoir fait des affichages directement dans les différentes méthodes des classes comme Personnage, Gaulois ou Romain n'étaient pas un choix très judicieux.

Il nous faudra donc modifier des méthodes déjà écrites afin qu'elles retournent la chaîne à affichée plutôt que d'afficher directement.

La méthode dans la classe Personnage que vous avez dû écrire afin de calculer la force du coup et redéfini dans la classe Soldat afin de prendre en compte son équipement est particulièrement embêtante car elle fait deux actions.

La première est de retourner le coup de la force une fois cette force absorbée par l'équipement du romain.

La deuxième est d'afficher ce qui se passe lorsqu'un gaulois frappe un romain.

Le refactoring sera donc un peu plus complexe car il faudra couper cette méthode en deux.

Cette expérience sur une évolution de code non prévue (ce qui est le plus souvent le cas en développement logiciel) nous permettra d'introduire en cours le besoin de structurer le code en séparant les responsabilités.



1. Dans le package batailles, créer une nouvelle classe. Dans la fenêtre de création de la classe :
 - Noter son nom : Embuscade,
 - Ajouter l'interface créée dans la partie 2 « bouton Add... »,
 - Décocher « Constructors from superclass »

Une fois la classe créée, au-lieu de retourner null dans chaque méthode, retourner le nom de la méthode.

Dans le package test_fonctionnel créer la classe TestBataille qui possède un main (cochez l'option « public static void main(String[] args) » dans la fenêtre de création de la classe).

Dans cette classe créer les gaulois et les soldats suivants (faire un copier/coller) :

```
Gaulois abraracourcix = new Gaulois("Abraracourcix", 5);
Village village = new Village(abraracourcix);

Gaulois agecanonix = new Gaulois("Agecanonix", 1);
Gaulois assurancetourix = new Gaulois("Assurancetourix", 2);
Gaulois asterix = new Gaulois("Astérix", 5);
Gaulois obelix = new Gaulois("Obelix", 15);
Gaulois cetautomatix = new Gaulois("Cétautomatix", 8);
Gaulois ordralfabetix = new Gaulois("Ordralfabetix", 8);

village.ajouterVillageois(agecanonix);
village.ajouterVillageois(assurancetourix);
village.ajouterVillageois(asterix);
village.ajouterVillageois(obelix);
village.ajouterVillageois(cetautomatix);
village.ajouterVillageois(ordralfabetix);

Soldat chorus = new Soldat("Chorus", 4, Grade.CENTURION);
Camp camp = new Camp(chorus);

Soldat minus = new Soldat("Minus", 2, Grade.SOLDAT);
Soldat brutus = new Soldat("Brutus", 5, Grade.CENTURION);
Soldat milexcus = new Soldat("Milexcus", 2, Grade.SOLDAT);
Soldat tullius = new Soldat("Tullius Octopus", 2, Grade.TESSERARIUS);
Soldat ballondebaudrus = new Soldat("Ballondebaudrus", 3, Grade.OPTIO);
Soldat quintilius = new Soldat("Quintilius", 2, Grade.SOLDAT);

camp.ajouterSoldat(brutus);
camp.ajouterSoldat(milexcus);
camp.ajouterSoldat(tullius);
camp.ajouterSoldat(ballondebaudrus);
camp.ajouterSoldat(minus);
camp.ajouterSoldat(quintilius);
```



Créer un objet de type :

- Conteur, par exemple le conteur "Goscinny",
- Embuscade,
- Faire conter l'histoire de l'embuscade par le compteur.

Affichage :

```
Je suis Goscinny. Je vais vous conter une histoire qui se déroule en 50 avant Jésus-Christ,
du temps où la Gaule est occupée par les Romains.
```

```
decrireContexte
choisirCombattants
preparerCombat
decrireCombat
donnerResultat
```

2. Décrire le contexte : retourne simplement le speech de l'embuscade.

Dans une sombre forêt dans un coin reculé de la Gaule, quatre comparses se promènent.

3. Choisir les combattants :

Pour écrire cette méthode ne pas oublier qu'une méthode ne doit pas dépasser 20 lignes. Pour qu'elle soit lisible la décomposer en méthodes privées simples, par exemple une méthode permettant de sélectionner les 4 promeneurs dans le village gaulois et une méthode permettant de sélectionner les 2 soldats qui ont pour grade SOLDAT participant à l'embuscade. L'affichage des promeneurs séparés de virgules, avec un « et » entre les deux derniers et terminant par un point est aussi compliqué. Il serait là aussi pertinent d'utiliser une méthode privée.

Affichage de cette partie :

```
Il s'agit de Ordralfabétix, Cétautomatix, Obelix et Astérix.
Mais cachés derrière des bosquets se cachent Quintilius et Minus.
```

A suivre au prochain TP...