

Design Patterns - TP3

Code initial pour le TP3



Rappel du cours : ☑☑☑ <http://bit.ly/jmb-cpoa>

Informations générales

NOM

BRUEL

Prénom

Jean-Michel

Groupe #

☒ Enseignants

☐ 1

☐ 2

☐ 3

☐ 4

☐ Innopolis

Pré-requis

Il vous faut :

☒ Un compte [GitHub](#)

☐ Un terminal de type [Git Bash](#) (si vous utilisez Window\$)



Essayez la commande suivante dans votre terminal pour vérifier votre environnement **git** :

```
git config --global -l
```

Tâche initiale

☒ Cliquez sur le lien Github Classroom fourni par votre enseignant (en fait c'est déjà fait si vous lisez ces lignes).

☐ Clonez sur votre machine le projet Github généré pour vous par Github Classroom.

☐ Modifiez le **README** pour modifier Nom, Prénom et Groupe.

☐ Commit & push:

 `commit/push`

fix #0 Initial task done



Dans la suite de ce document, à chaque fois que vous trouverez un énoncé commençant par **fix #...** vous devez vérifier que vos scripts/fichiers modifiés sont bien dans votre dépôt local en vue de committer et de pusher les modifications sur votre dépôt distant en utilisant comme message de commit cet énoncé.



- Si vous voulez vérifier que vous êtes prêt pour le **fix #0**, utilisez la commande : **make check**.
- Si vous voulez avoir la liste des Todos de ce TP/TP, exécutez **make todos**.

Les exercices de ce TD sont tirés de l'excellent livre "Tête la première : Design Pattern". Bert Bates, Eric Freeman, Elisabeth Freeman, Kathy Sierra. Editions O'Reilly. 2005.



Les Pizzerias

QUESTION

- Implémentez entièrement l'application de Pizzeria vue en TD3 de manière à ce :
 - qu'elle mette en œuvre le patron Fabrique Abstraite
 - qu'elle mette en œuvre le patron Singleton (pour les fabriques)
 - que le programme de main (attention, ce n'est pas un test JUnit) livré dans votre dépôt produise le résultat ci-dessous.



Commencez par le programme fourni et servez-vous des *QuickFix* pour "générer" un maximum de code.

Rendus attendus

ToDo

- ☐ un **pom.xml** fonctionnel qui lance les tests maven de votre application
- ☐ un **build.gradle** fonctionnel qui lance les tests gradle de votre application
- ☐ le diagramme de classe de votre application, dans un fichier **TP3.plantuml**, placé dans le répertoire **doc** de votre dépôt.

On utilisera le modèle suivant pour les pizzas :

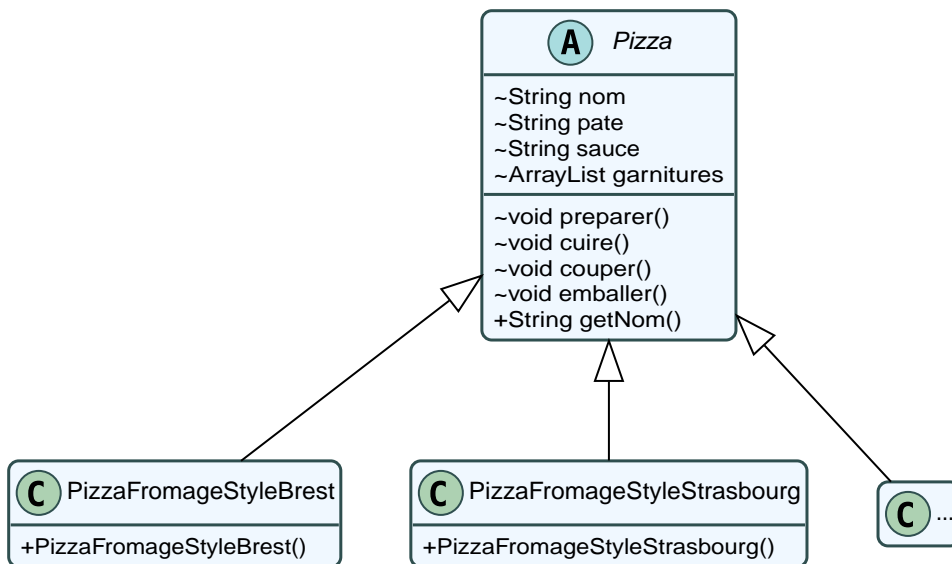


Diagram generated using <http://plantuml.sourceforge.net>.

Figure 1. Diagramme de classe des Pizzas

```
public class PizzaTestDrive {
    public static void main(String[] args) {
        Pizzeria boutiqueBrest = PizzeriaFactory.getInstance().creer("Brest");
        Pizzeria boutiqueStrasbourg = PizzeriaFactory.getInstance().creer(
            "Strasbourg");

        Pizza pizza = boutiqueBrest.commanderPizza("fromage");
        System.out.println("JMB a commandé une " + pizza.getNom() + "\n");

        pizza = boutiqueStrasbourg.commanderPizza("fromage");
        System.out.println("JMI a commandé une " + pizza.getNom() + "\n");
    }
}
```

Résultat d'exécution

```
$ java -jar target/pizzeria.jar
Préparation de Pizza sauce style brest et fromage
Étalage de la pâte...
Ajout de la sauce...
Ajout des garnitures:
  Parmigiano reggiano râpé
Cuisson 25 minutes à 180°
Découpage en parts triangulaires
Emballage dans une boîte officielle
JMB a commandé une Pizza sauce style brest et fromage

Préparation de Pizza pâte style Strasbourg et fromage
Étalage de la pâte...
Ajout de la sauce...
Ajout des garnitures:
  Mozzarella en lamelles
Cuisson 25 minutes à 180°
Découpage en parts carrées
Emballage dans une boîte officielle
JMI a commandé une Pizza pâte style Strasbourg et fromage
```



Attention, ce TP est évalué dans le cadre du contrôle continu. L'autograding de classroom lancera les tests via `gradle test` et `maven test`, ainsi que `test0` et le test du modèle. Ceci constituera 80% de la note. Les 20% suivants seront évalués par votre encadrant de TP sur la base de vos tests (qualité et nombres).

 `commit/push`

```
fix #All: Completed all duties
```

Contributeurs

- [Jean-Michel Bruel](#)

À propos...

Baked with [AsciiDoctor](#) (version [2.0.12](#)) from 'Dan Allen', based on [AsciiDoc](#). 'Licence Creative Commons'.  [licence Creative Commons Paternité - Partage à l'Identique 3.0 non transposé](#).