

Préparé par : Ahlam Chakrane

Le choix des Design Pattern :

1- le pattern stratégie :

Puisque nous voudrions que la méthode « traiter » puisse traiter le contenu du dessin en utilisant une famille d'algorithmes qui sont interchangeables dynamiquement, et puisque chaque algorithme peut évoluer indépendamment de la classe qui les utilise.

2- le pattern observer :

Vu que nous voudrions qu'à chaque fois que l'état de l'objet paramétré change,

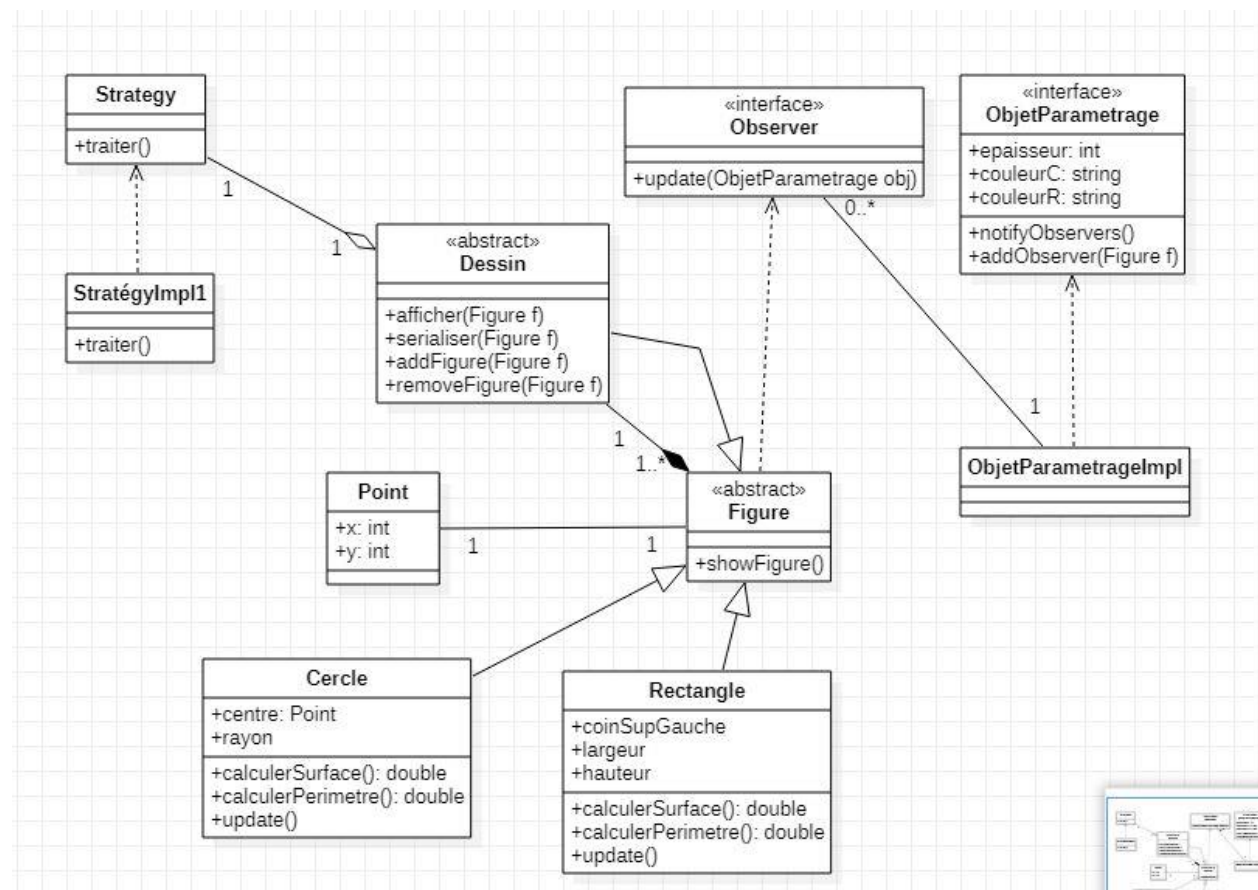
toutes les figures doivent être notifiées pour se mettre à jour.

3- le pattern Composite :

Car nous voudrions Donner la possibilité de Créer un groupe de figures et dans Chaque groupe

on peut trouver d'autres groupes.

Le diagramme de classe :



Test de l'application :

```
"C:\Program Files\Java\jdk-17.0.4.1\bin\java.exe" ...  
Cercle  
Couleur du contour: Red  
Couleur du remplissage: Yellow  
Epaisseur du contour: 1.2  
Perimetre: 18.84  
Surface: 28.26  
  
Rectangle  
Couleur du contour: Orange  
Couleur du remplissage: Dark  
Epaisseur du contour: 1.5  
Perimetre: 20.0  
Surface: 24.0  
Traitement 1 de strategy  
  
Surface des figures contenues dans le dessin: 28.26  
Perimetre des figures contenues dans le dessin: 18.84
```

Affichage des figures de dessins

Couleur du contour: Red

Couleur du remplissage: Yellow

Epaisseur du contour: 1.2

Perimetre: 18.84

Surface: 28.26

Couleur du contour: Orange

Couleur du remplissage: Dark

Epaisseur du contour: 1.5

Perimetre: 20.0

Surface: 24.0

Supprimer une figure du dessin

Traitement 1 de strategy

Notifier observers des changements On modifie juste un seul attr

Couleur du contour: Pink

Couleur du remplissage: Yellow

Supprimer une figure du dessin

Traitement 1 de strategy

Notifier observers des changements On modifie juste un seul attr

Couleur du contour: Pink

Couleur du remplissage: Yellow

Epaisseur du contour: 1.2

Perimetre: 18.84

Surface: 28.26

Process finished with exit code 0

|