

Le fichier Help :

La phase 1 :

Cette phase consiste à créer différentes formes qui héritent de la classe Forme en implémentant toutes ces méthodes, les classes sont Triangle , Chapeau , Quadrilatere.

Pour créer n'importe quel forme on doit appeler la classe avec comme paramètre (nom de la forme , le poin1 , le point2 , ...)

Pour les classes Triangle et Chapeau faut les appeler avec 3 points , par contre pour la classe Quadrilatere faut l'appeler que avec 2 points en paramètre..

Pour ajouter les formes à l'ardoise : ardoise.ajouterForme(..).

On peut tester cette phase avec le fichier exécutable (TestPhase1.java).

A propos du déplacement des oiseaux :

J'ai déplacé les oiseaux un par un (je trouve pas un moyen de les déplacer en une seule commande sans avoir recours à la phase 2 -Les Formes Composées-)

La phase 2 :

Cette phase consiste à créer une autre classe que j'ai appelé FormeComposee, cette classe hérite de la classe Forme et implémente toutes ces méthodes.

Pour Tester cette classe j'ai créer des formes composées Maison qui contient (le toit , le corps , la porte) ; et aussi L'étoile qui contient quatre chapeaux.

Dans cette phase j'ai pu déplacer les oiseaux (ce qui a été demandé dans la phase 1 😊), et ceci avec cette commande :

```
ardoise.deplacer("C",10,20);
```

Ce qui signifie déplacer tous les chapeaux (tous les oiseaux) ce qui est possible après avoir mis tous les autres chapeaux en forme complexes (le toit de la maison et aussi les branches de l'étoile).

On peut tester cette phase avec le fichier exécutable (TestPhase2.java).

La phase 3 :

Cette phase consiste à tester toutes les classes qu'on vient de créer et avoir un coverage le plus près de 100%.

TestPhase3 (14 juin 2023 16:56:00)

Element	Coverage	Covered Instru...	Missed Instruct...	Total Instructio...
▼ 📁 Classes	100,0 %	472	0	472
▼ 📁 (default package)	100,0 %	472	0	472
> 📄 Chapeau.java	100,0 %	116	0	116
> 📄 FormeComposee.java	100,0 %	94	0	94
> 📄 Quadrilatere.java	100,0 %	134	0	134
> 📄 Triangle.java	100,0 %	128	0	128

Pour tous les cas possibles :

1^{er} Cas :

Le cas où on crée un segment on n'importe quel forme avec deux points similaires, dans ce cas on peut rajouter à la classe Forme et Segment :

```
if (segment1.getPointDepart().equals(segment2.getPointArrivee())) { throw new  
IllegalArgumentException("Non valide , veuillez modifier l'un des points passées en paramètre");
```

2eme Cas :

Si l'on veut déplacer une forme en passant en paramètres négatives, dans ce cas on ajoute :

```
if (deplacementX < 0 || deplacementY < 0) { throw new IllegalArgumentException("Les valeurs de  
déplacement passées en paramètre doivent être positives"); }
```

3eme Cas :

Si l'on veut ajouter une forme qui existe déjà dans l'ardoise avec les mêmes points et le même nom, dans ce cas on ajoute :

```
if (formes.contains(forme)) { throw new IllegalArgumentException("Cette forme existe déjà dans  
l'ardoise , veuillez changer le nom de la forme ou les coordonnées de la forme"); }
```

Cette phase est exécutable avec le fichier (TestPhase3.java).