

Introduction

Vous avez sans doute remarqué que nous vous avons fourni par erreur les classes implémentant la classe abstraite Figure (Circle, Ellipse, Rectangle, RoundedRectangle et Polygon) ainsi que les listeners servant à construire ces figures en utilisant des événements souris (classes filles de AbstractCreationListener). Pour compenser cette erreur, nous allons quelque peu modifier le travail à réaliser.

Travail à réaliser

- Un certain nombre de classes restent à créer ou à compléter
 - `utils.FlyweightFactory`
 - `figures.Drawing`
 - `widgets.DrawingPanel`
 - `widgets.EditorFrame` : Créez l'interface graphique en éditant ce fichier avec WindowBuilder intégré à eclipse : Menu contextuel → Open With → WindowBuilder Editor
 - Complétez les différents Listeners déclarés en tant que classes internes et agissant principalement sur l'attribut `drawingModel` (instance de la classe `Drawing`).
 - `widgets.InfoPanel`
 - Vous ajouterez une fonctionnalité de déplacement des figures dans l'application. Pour ce faire il faudra :
 - Ajouter un attribut `moveMode` à la classe `Drawing` qui permettra de switcher entre le mode création de nouvelles figures et le mode déplacement des figures existantes.
 - Ce mode pourra être commandé depuis la classe `EditorFrame` avec un `JCheckBox`, un `JToggleButton` ou encore avec un `JCheckBoxMenuItem` en échangeant les listeners associés à la classe `DrawingPanel`.
 - Il faudra donc ajouter un listener particulier (`MoveShapeListener`) parmi les classes filles de `AbstractCreationListener` permettant non plus de créer une nouvelle figure mais de déplacer une figure située sous le curseur.
 - Ajouter un attribut de type `AffineTransform` à la classe `Figure` afin que celle ci puisse être appliquée au contexte graphique (`Graphics2D`) lors du dessin de la figure et modifiée lorsque l'on déplacera une figure.
- Par ailleurs il vous faudra créer les classes suivantes
 - Dans le package `figures` (héritières de `AbstractFigure`)
 - ~~Circle~~
 - ~~Ellipse~~
 - ~~Rectangle~~
 - ~~RoundedRectangle~~
 - ~~Polygon~~
 - Polygone régulier (à n côtés pour faire des Triangle, Carré, Pentagone, ...)
 - Etoile (à n branches)
 - Il faudra mettre à jour l'enum `enums.FigureType` pour y ajouter vos nouvelles figures.

- Dans le package figures.creationListeners (héritières de AbstractCreationListener)
 - ~~Tout les listeners dont vous aurez besoin pour créer graphiquement les figures ci-dessus.~~
 - Les listeners dont vous aurez besoin pour créer les polygones réguliers et les étoiles (vous pourrez vous inspirer des listeners existants) ainsi qu'un listener particulier vous permettant de déplacer les figures plutôt que de les créer.