

Mémo Diagramme de classe

Description des entités (classe ou interface)

- Une **entité** est représentée sous la forme d'une boîte avec le nom de cette entité
- Pour rendre les choses lisibles sur copie, une classe **abstraite** contient l'annotation <<abstract>> à coté de son nom et une **interface** contient l'annotation <<interface>>
- Une boîte décrivant une entité contient les **attributs non objets** ainsi que ses **méthodes**, leur **signature** (paramètre, type de retour) et leur **visibilité** (public + , privé -)

NomEntité	<ul style="list-style-type: none">• Annotation <<interface>> si c'est une interface• Annotation <<abstract>> si c'est une classe abstraite• En italique si classe abstraite
attributs	<ul style="list-style-type: none">• Attribut → « nom : type »• Nom d'un attribut commence avec une minuscule• Visibilité : + public, - private, # protected
méthodes	<ul style="list-style-type: none">• Méthode → « nom (param) : type retour »• Nom de méthode est verbe infinitif + commence minuscule• Nom en italique si méthode abstraite• Nom surligné si méthode statique• Visibilité : + public, - private, # protected

Description des relations

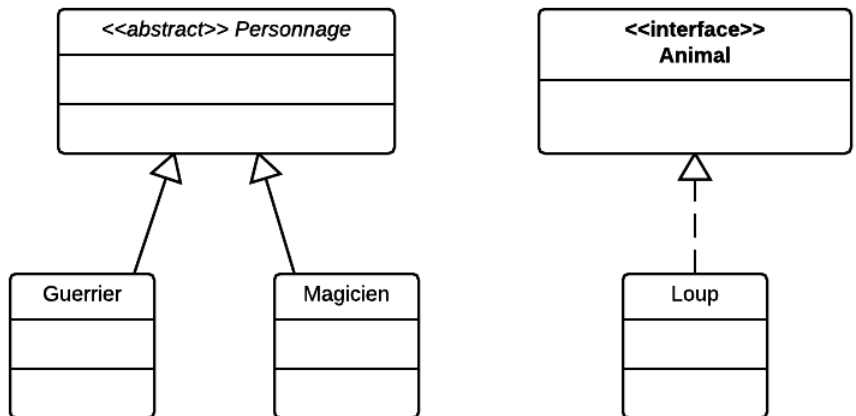
- La **relation de sous-typage** représente le fait qu'une classe hérite d'une autre classe ou implémente une interface. La relation d'héritage est représentée par une ligne avec une flèche creuse (la ligne est en pointillé s'il s'agit d'une relation d'implémentation)

Héritage ————>

Implémentation - - - ->

On peut lire «hérite de» ou «implémente» en suivant le sens de la flèche.

Guerrier hérite de Personnage



- La **relation d'association** représente le fait qu'une classe possède en attribut une autre classe. La relation d'association est représentée par une ligne pleine avec une flèche simple **de la classe contenante vers la classe utilisée**. Une relation d'association est définie par les classes reliées, un **nom**, une **visibilité** (public, privé) et des **cardinalités**.

Association $\xrightarrow[\text{nom}]{\text{card}}$

On peut lire «possède en attribut» en suivant le sens de la flèche.

UnRectangle possède en attribut un coin de type UnPoint

