

Les diagrammes d'objets

Table des matières

I. Théorie	3
II. Exercice : Quiz	9
III. Pratique	10
IV. Exercice : Quiz	14
V. Essentiel	15
VI. Auto-évaluation	15
A. Exercice	15
B. Test	16
Solutions des exercices	16

I. Théorie

Durée : 1 h

Environnement de travail : un PC connecté à internet

Contexte

Dans le contexte UML, il existe différents types de diagrammes qui sont regroupés en plusieurs catégories.

On retrouve d'abord les diagrammes statiques, que l'on appelle diagramme de structure. Il y a également les diagrammes de comportement et, pour finir, les diagrammes dynamiques, que l'on appelle aussi les diagrammes d'interaction.

Nous allons voir ensemble l'ensemble de ces diagrammes qui viennent compléter le diagramme de classe. Nous verrons également dans quel but les utiliser et comment les construire.

Nous allons commencer dans ce cours par le diagramme d'objet, qui fait partie de la catégorie des diagrammes statiques et qui vient apporter des informations supplémentaires au diagramme de classe.

Définition Qu'est-ce qu'un diagramme d'objets ?

Un diagramme d'objets, c'est un diagramme qui va représenter une classe, mais en y ajoutant les spécificités de celle-ci. Il fait partie de la catégorie des diagrammes statiques.

En d'autres termes, c'est une instance d'un diagramme de classe représentant les objets et les liens qui les unissent, mais pas que. Sachez qu'il est possible de construire un diagramme d'objets à partir d'autres diagrammes UML qui existent.

En développant un diagramme d'objets, on va se concentrer sur tous les attributs d'un ensemble d'objets, ainsi que sur les liens qu'ils ont les uns avec les autres à un instant donné, sans porter aucun dynamisme au sein du diagramme, bien que l'on puisse écrire des états.

En bref, il sert à montrer un contexte précis à un instant précis : par exemple lorsqu'on souhaite montrer un environnement avant ou après une interaction.

Il faut toutefois noter que, quand on parle d'un diagramme d'objet, il va avant tout s'agir d'un outil de documentation et de communication, rien de plus. Il facilite la communication entre les différents acteurs d'un projet.

Exemple

Un objet représente une instance d'une classe. Si nous définissons une classe « *Personne* », alors nous pouvons créer un objet en définissant alors une instance de notre classe « *Personne* ».

Durant le processus de développement d'un diagramme d'objets, il faut que le diagramme puisse respecter plusieurs contraintes. Tout d'abord, il faut veiller au bon respect des cardinalités que nous avons mises en place dans notre diagramme de classe. Ensuite, chaque attribut d'une classe doit avoir une valeur affectée dans chaque instance de classe.

En d'autres termes, nous devons, pour une instance de classe, donner une valeur à chaque attribut. Imaginons que dans notre classe, nous ayons renseigné que les attributs seraient un nom, un prénom et un âge. Pour l'exemple de notre classe *Personne*, nous pouvons créer une instance de notre classe *Personne*, et nous lui assignons notamment un nom, un prénom et un âge qui correspondent aux attributs de la classe *Personne*, définie un peu plus tôt dans notre diagramme de classe.

Ce diagramme d'objets est très utile pour obtenir une vue d'ensemble du système que l'on va développer.

On va se servir du diagramme d'objets afin de donner des exemples ou différents cas de figure qui permettront, par la suite, d'affiner le diagramme de classe afin de mieux le comprendre.

Les différents éléments du diagramme d'objets

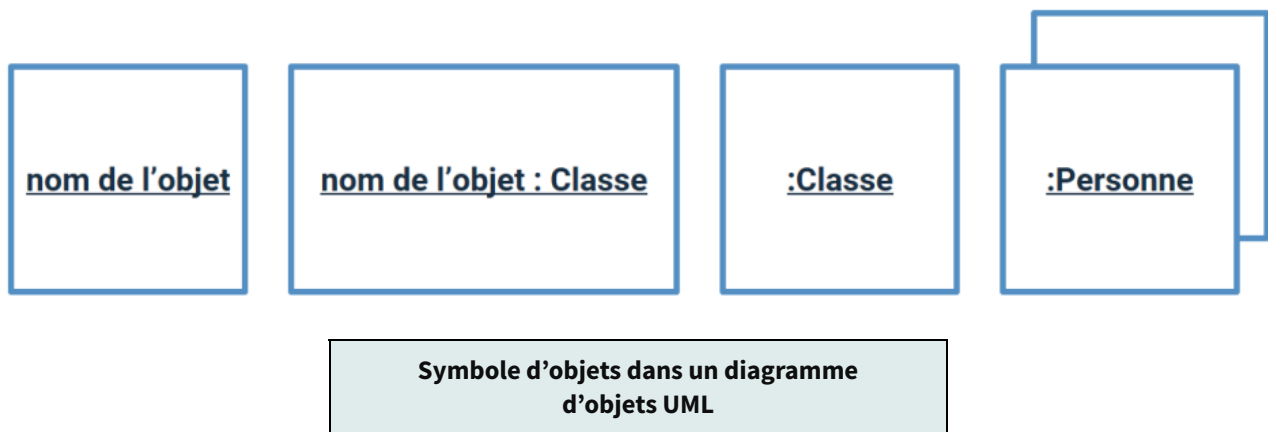
Un diagramme d'objets est composé d'objets qui représentent les instances de nos classes, mais il est également composé de liens qui représentent les instances d'associations. En effet, tout comme notre diagramme de classe, un diagramme d'objets va avoir des associations entre les objets. Voyons-les plus en détail :

Un objet

Un objet renvoie à une instance d'une classe. Il figure l'état d'une classe à un instant précis.

Il est possible de représenter un objet de diverses façons.

Chaque objet va être représenté dans un rectangle dans lequel va figurer le nom souligné de celui-ci, et éventuellement la ou les valeurs d'un ou plusieurs attributs.

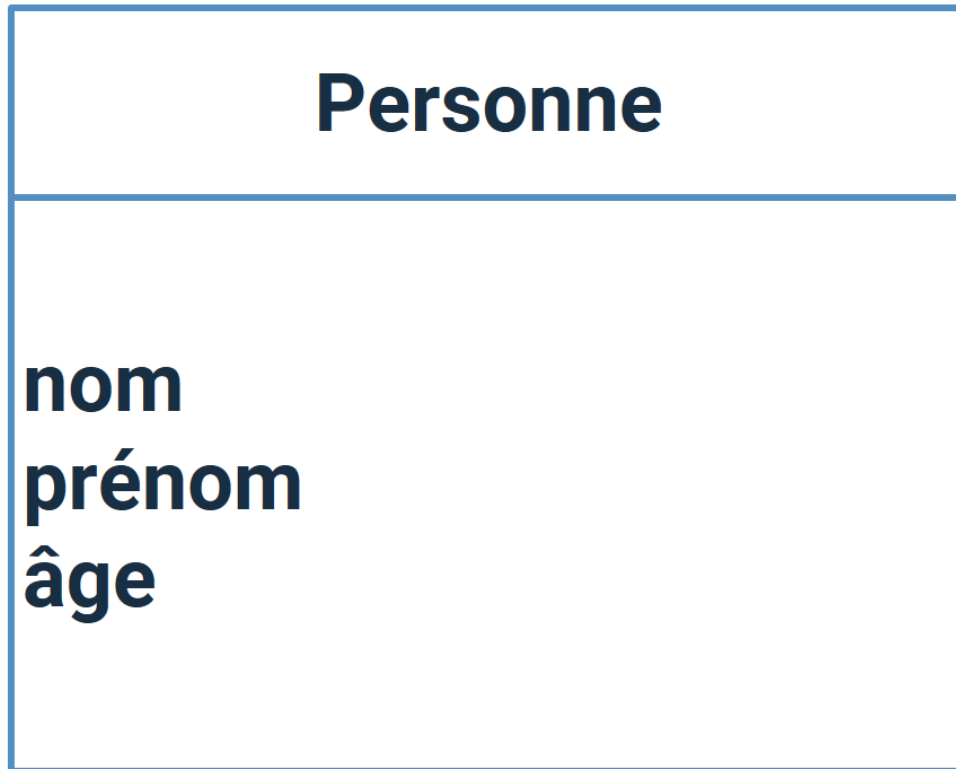


De gauche à droite :

- Représente une instance nommée d'une classe anonyme (exemple - voitureDeLouna),
- Représente une instance nommée d'une classe (exemple - voitureDeLouna : voiture),
- Représente une instance anonyme d'une classe (exemple - : voiture),
- Représente une collection d'instances anonymes d'une classe. Un message envoyé vers un groupe est reçu par tous les objets du groupe.

Les valeurs des attributs d'objets

Une représentation d'un objet peut contenir des attributs significatifs comme dans notre exemple :



Classe personne dans un diagramme de classe



Léo, instance de classe Personne avec des attributs

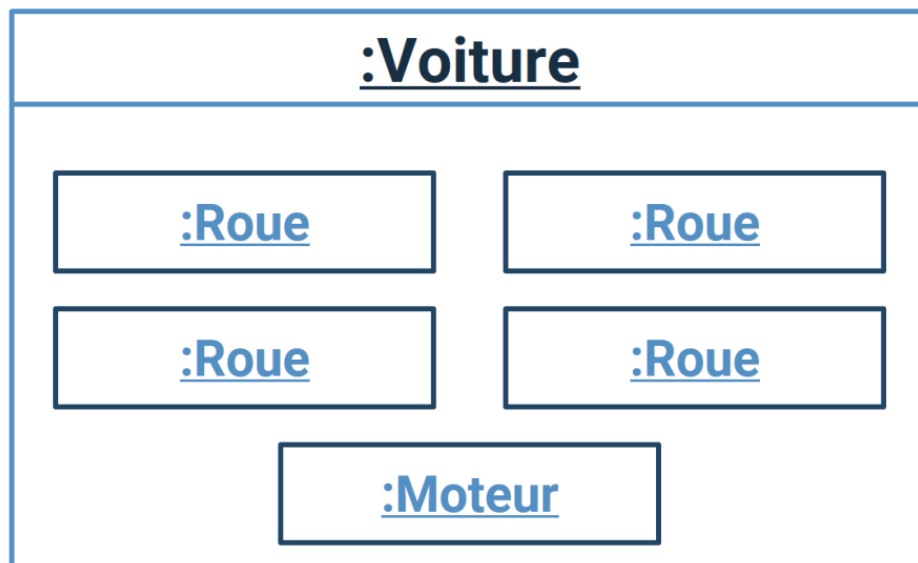
La différence majeure entre les attributs d'une classe au sein d'un diagramme de classe et les attributs d'un objet représentant une instance de classe réside dans le fait que, dans notre diagramme de classe, seuls les attributs sont listés avec la nature de ceux-ci. Par exemple, le nom de la classe Personne peut être une chaîne de caractères de maximum 255 caractères et non nullable. Pour ce qui est de l'âge, toujours dans notre diagramme de classe, ce serait donc un nombre entier, un INT, non nullable.

Dans notre diagramme d'objets, nous allons prendre en compte ces attributs et nous allons leur assigner des valeurs. Comme vous pouvez le constater ci-dessus, pour notre instance de la classe Personne que nous avons nommé Léo, nous avons assigné la valeur « Mattei » à l'attribut du nom, la valeur « Léo » à l'attribut du prénom et la valeur « 54 » à l'attribut de l'âge.

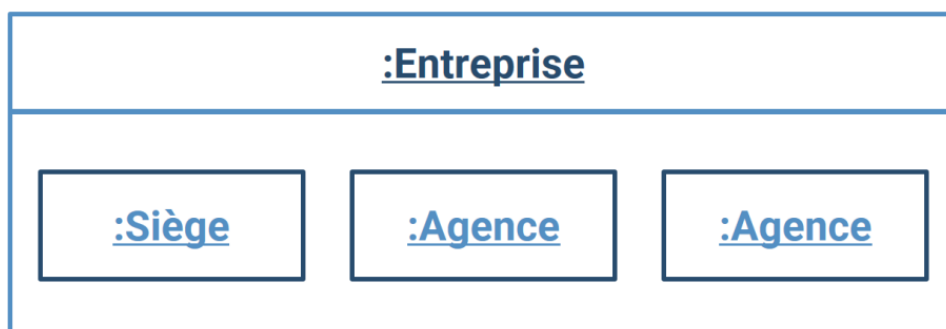
Dans un diagramme d'objets, outre le fait de créer des objets uniques représentant une instance de classe, il est tout à fait possible de créer des objets que l'on appelle « *objet composite* ». Voyons ce qu'est un objet composite un peu plus en détail.

Un objet composite

Un objet composite, c'est tout simplement un objet formé d'autres objets représentés de manière visuelle. Si nous prenons l'exemple d'un objet voiture, il peut être composé d'autres objets tels que des roues ainsi que d'un moteur.



Exemple d'objet composite d'une classe Voiture



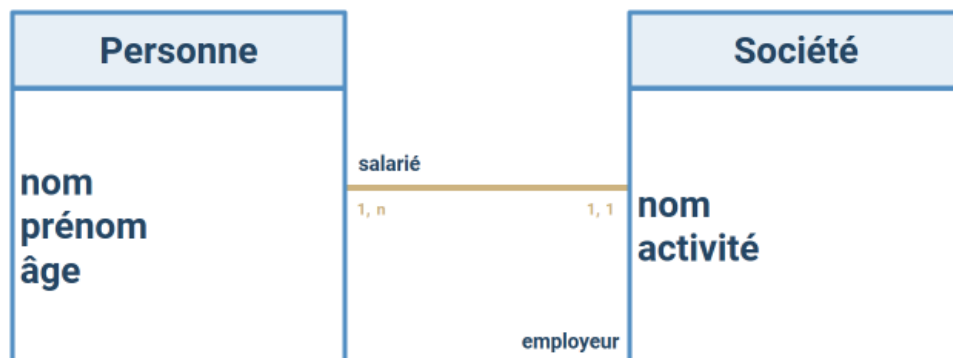
Exemple d'objet composite d'une classe Entreprise

Maintenant que nous avons passé en revue les différents types d'objets qu'il est possible de mettre en place dans un diagramme d'objets, voyons ensemble les liens entre ces objets.

Les liens

Tout comme les diagrammes de classe, les diagrammes d'objets vont avoir des liens dans leur structuration.

Les liens renvoient aux relations entre les différents objets. Ils sont représentés par une ligne reliant simplement différents objets. Un exemple simple ci-dessous avec un diagramme de classe :



Liens entre deux entités dans un diagramme de classe

Dans l'exemple de notre diagramme de classe, nous avons représenté deux entités, une entité **Personne** et une entité **Société**. Nous avons, pour la classe **Personne**, les attributs « *nom* », « *prénom* » et « *âge* ». Pour la classe **Société**, nous avons les attributs « *nom* » et « *activité* ».

Nous pourrions être un peu plus précis dans notre diagramme de classe en indiquant la nature des différents attributs. Cela peut se faire en définissant par exemple des chaînes de caractères avec une longueur définie ou des nombres entiers avec une option non nullable.

Nous avons une relation entre ces deux entités. Une personne peut travailler au maximum dans une seule société, mais une société peut recruter plusieurs personnes. Petit rappel, aucune limite n'est donnée au sujet du nombre d'employés, alors nous notons avec un « *n* » notre relation.

Maintenant, voyons comment il est possible d'agencer cette relation au sein d'un diagramme d'objets.

Nous allons mettre de côté le détail des relations *one-to-one* et *one-to-many*, afin de le schématiser directement sur notre diagramme d'objets. Nous allons alors nous retrouver avec des liens simples, à savoir les lignes simples qui connectent les objets entre eux. Nous allons créer 2 instances de notre classe **Personne**, afin de signaler que nous pouvons avoir, comme notre diagramme de classe l'indique à la base, plusieurs personnes dans une société. Pour éviter de surcharger le diagramme, nous allons représenter deux personnes afin que la relation soit facile à distinguer, sans surcharger notre diagramme d'objets. On se doit de garder nos diagrammes UML les plus lisibles possibles.

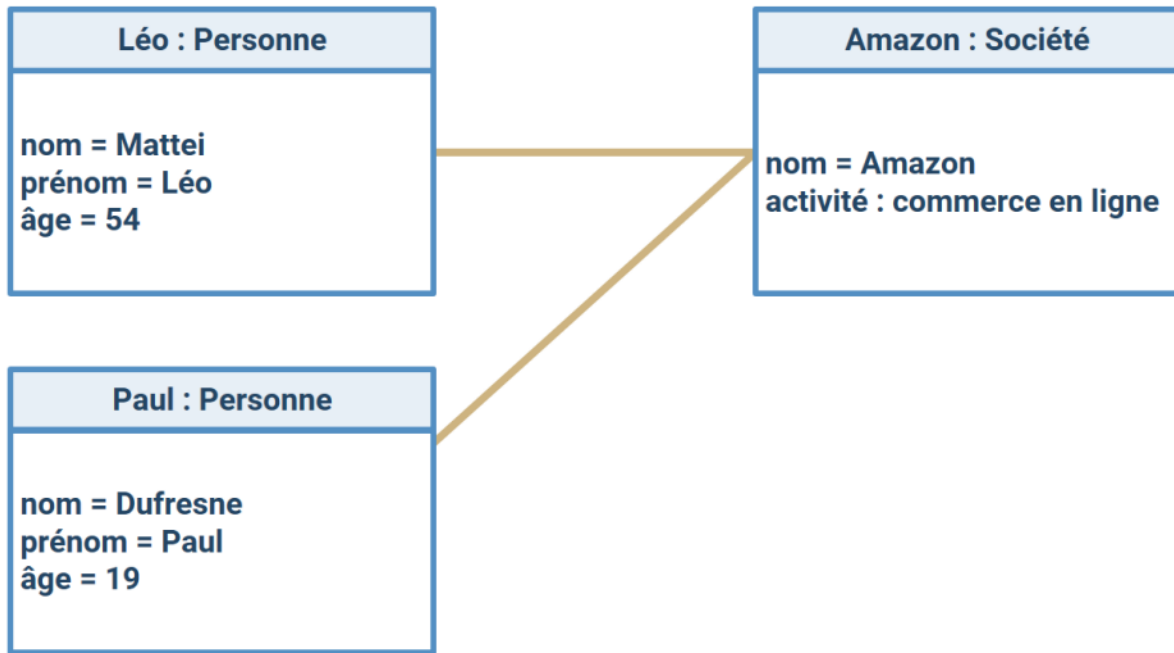


Diagramme d'objets représentant notre diagramme de classe

:Ordinateur [Calcule]

Exemple d'une instance de la classe Ordinateur avec l'état Calcule



Exemple d'une instance de la classe Ampoule qui passe de l'état Allumée à l'état Éteinte

Différences et similitudes entre diagramme de classe et d'objets

En UML, les diagrammes d'objets fournissent un instantané des instances dans un système et des relations entre les instances. En séquençant les éléments du modèle dans un diagramme de classes, vous pouvez explorer le comportement d'un système à un moment donné. Les différences et les similitudes entre les diagrammes d'objets et les diagrammes de classe sont les suivantes :

- Un diagramme d'objets est un diagramme structurel UML qui montre les instances des classificateurs dans les modèles.
- Les diagrammes d'objets utilisent une notation qui est similaire à celle utilisée dans les diagrammes de classes.
- Les diagrammes de classes montrent les classificateurs réels et leurs relations dans un système.
- Les diagrammes d'objets montrent des instances spécifiques de ces classificateurs et les liens entre ces instances à un moment donné.
- On peut créer des diagrammes d'objets en instanciant les classificateurs dans des diagrammes de classes, de déploiement, de composants et de cas d'utilisation.

Exercice : Quiz

[solution n°1 p.17]

Question 1

Un diagramme d'objets possède beaucoup de similitudes avec un diagramme de classe.

- ☐ Vrai
- ☐ Faux

Question 2

Pour représenter une classe dans un diagramme d'objets, il faut passer par une instance de cette classe.

- ☐ Vrai
- ☐ Faux

Question 3

Cet objet représente :



nom de l'objet : Classe

- ☐ Une instance nommée d'une classe nommée
- ☐ Une instance anonyme d'une classe nommée
- ☐ Une instance nommée d'une classe anonyme
- ☐ Une collection d'instances anonymes de classe nommée

Question 4

Si la classe contient des attributs, faut-il assigner des valeurs aux attributs en créant une instance de cette classe ?

- ☐ Non, pas besoin
- ☐ Oui, c'est nécessaire

Question 5

Un diagramme d'objets représente un environnement dynamique.

- ☐ Vrai
- ☐ Faux

III. Pratique

Processus de création d'un diagramme d'objets

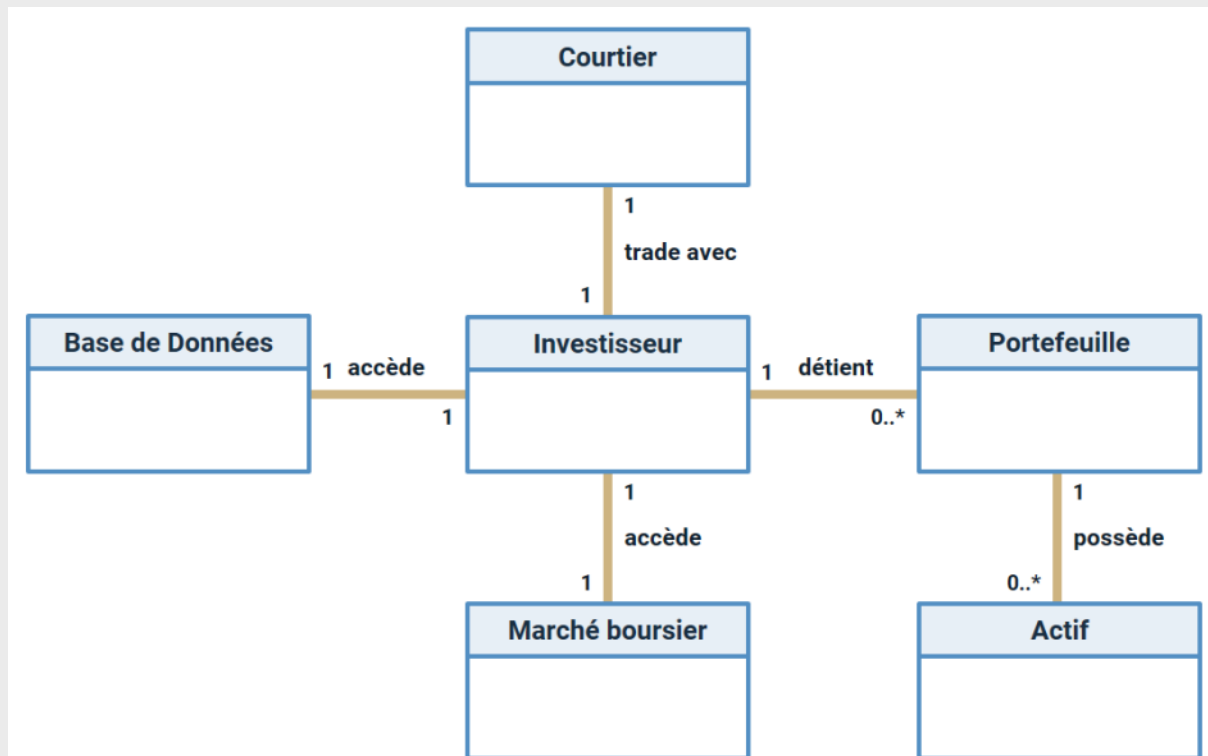
D'abord, il faut identifier le mécanisme que vous souhaitez modéliser. Un mécanisme représente certaines fonctions ou certains comportements de la partie du système que vous modélisez, et qui résultent de l'interaction d'une société de classes, d'interfaces et d'autres éléments.

Pour chaque mécanisme, identifiez les classes, interfaces et autres éléments qui participent à cette collaboration ; identifiez également les relations entre ces éléments.

Considérez un scénario qui parcourt ce mécanisme. Figez ce scénario à un moment donné et faites un rendu de chaque objet qui participe au mécanisme.

Exposez les valeurs d'état et d'attribut de chacun de ces objets, si nécessaire, pour comprendre le scénario.

Exemple



**Diagramme de classe
simplifié**

Ici, nous avons donc un diagramme de classe nettement simplifié, mais toutefois suffisamment compréhensible pour nous projeter au travers d'un diagramme d'objets.

Prenons un instant pour décrire ce qu'il se passe au sein de notre diagramme de classe avant de nous attaquer à la conception de notre diagramme d'objets. Le diagramme de classe représente les entités qui vont constituer notre système d'investissement en bourse.

Nous avons un bon nombre d'entités qui ont des relations entre elles. Nous pouvons notamment retrouver l'entité Investisseur qui peut dans un premier temps accéder à la base de données et au marché boursier, mais peut aussi effectuer du trading avec un courtier. De plus, notre entité Investisseur peut détenir un portefeuille. Un portefeuille peut posséder un ou plusieurs actifs en son sein et sera évidemment rattaché à un seul investisseur.

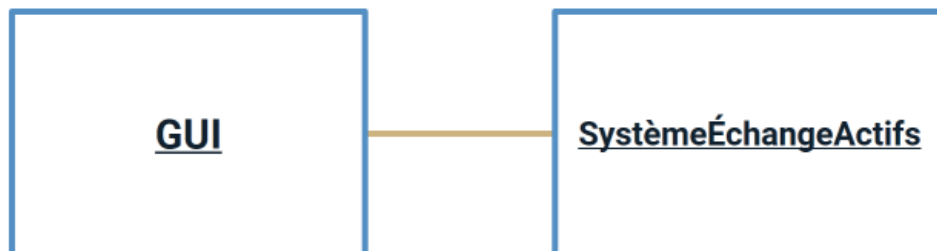
Pour représenter notre diagramme d'objet, nous allons poser un contexte à notre système.

Nous avons un système d'échange d'actifs qui est relié à une interface graphique. Ce système d'échange d'actifs peut interagir aussi bien avec le marché boursier, la base de données, l'investisseur ou même le courtier. Afin que l'investisseur puisse avoir ses relations correctement représentées, cet objet qui sera notre système d'échange d'actifs servira de passerelle. Voyez cela un peu comme une application de trading.

L'investisseur, lui, sera relié à son portefeuille et son portefeuille sera relié au(x) actif(s) qu'il possède.

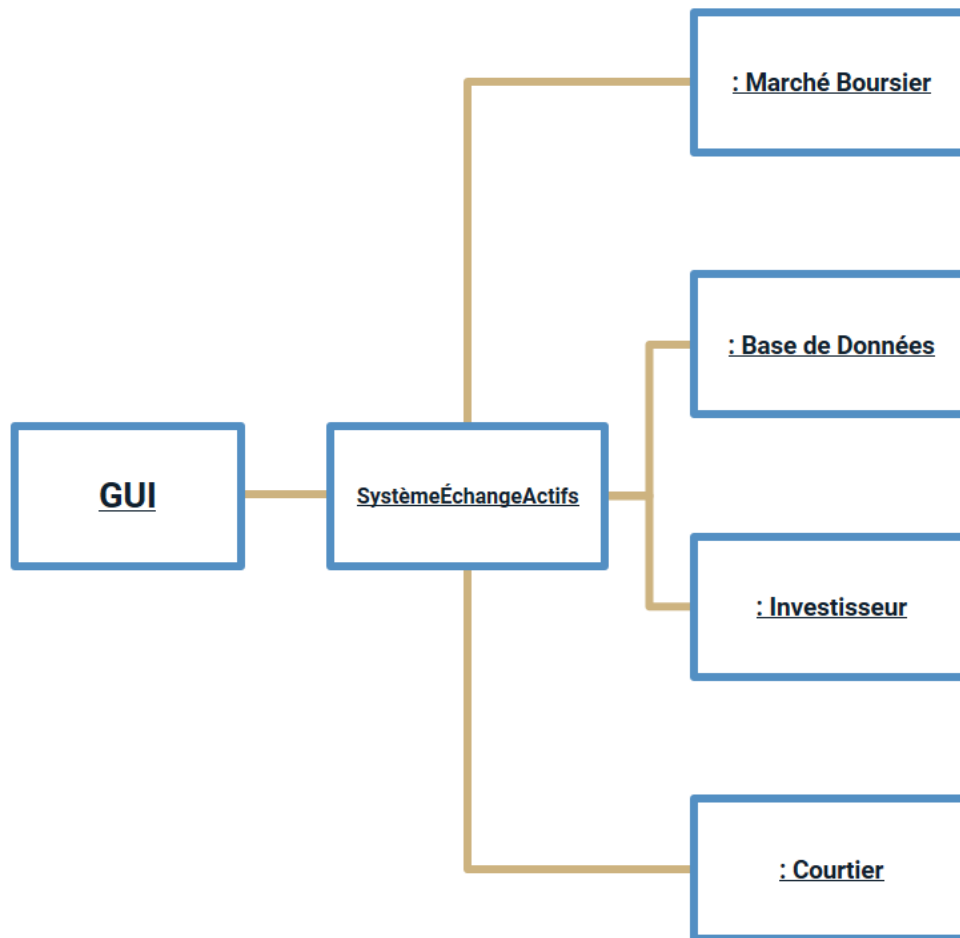
Étape par étape

Tout d'abord, il faut mettre en place le principal, notre système d'échange d'actifs ainsi que l'interface graphique.

**Mise en place de l'interface graphique et du système d'échange
d'actifs**

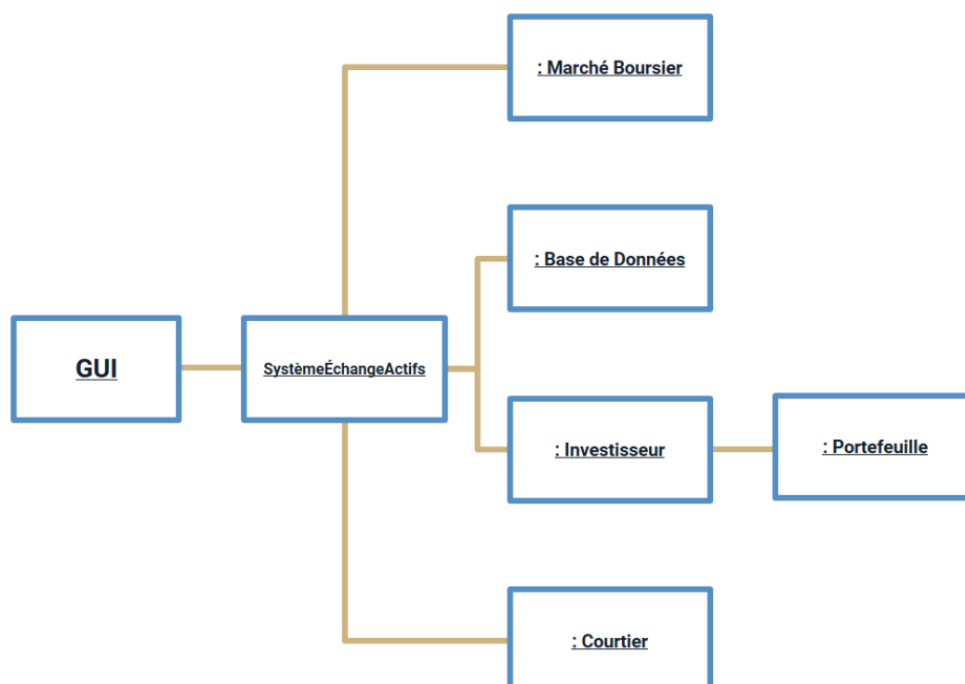
On abrège généralement le terme d'interface graphique par l'acronyme GUI qui signifie en anglais *Graphical User Interface*. Il faut noter que nous avons nommé les objets de sorte que ce soient des objets et non des instances de classes.

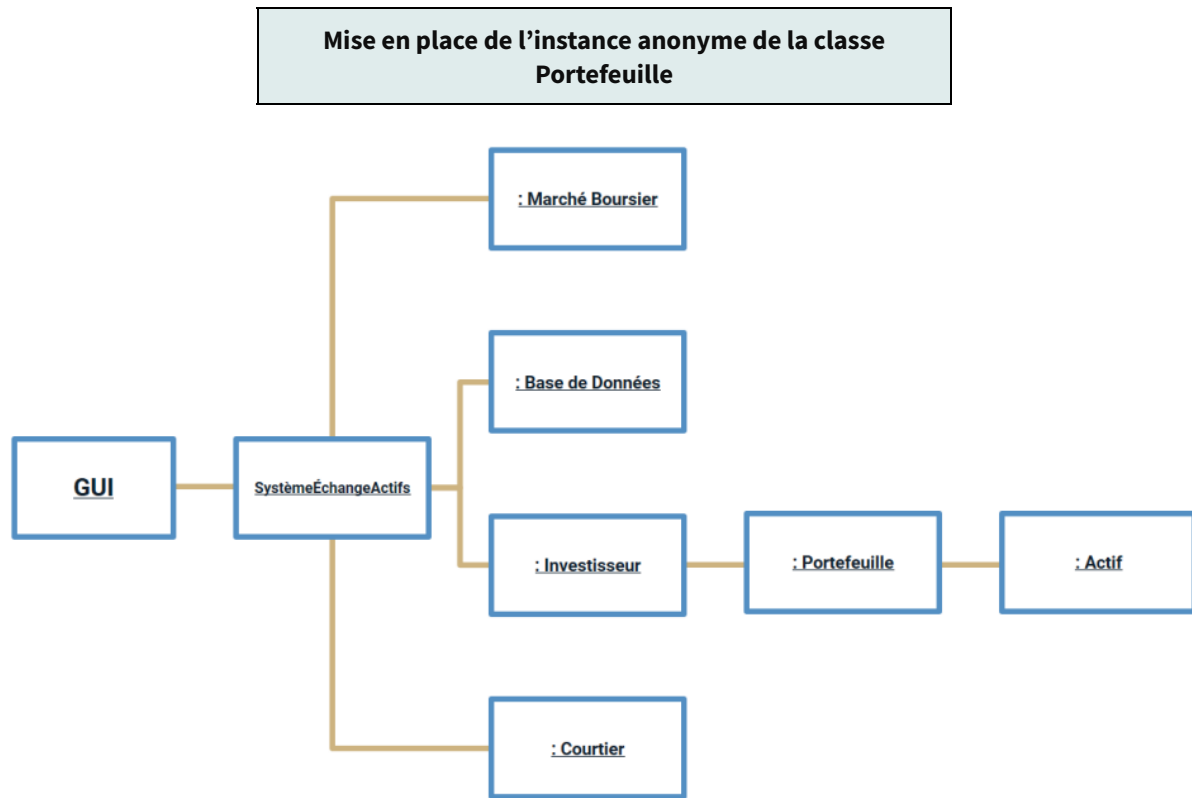
Une fois tout ceci mis en place, il faut maintenant installer le marché boursier, la base de données, l'investisseur et le courtier. Pour simplifier la chose, nous allons instaurer des instances anonymes de classe.



Mise en place des instances anonymes de classes

Une fois les instances mises en place et que tout est relié à notre système d'échange d'actifs, nous allons pouvoir mettre en place respectivement le portefeuille de l'investisseur, puis une instance anonyme de la classe Actif.

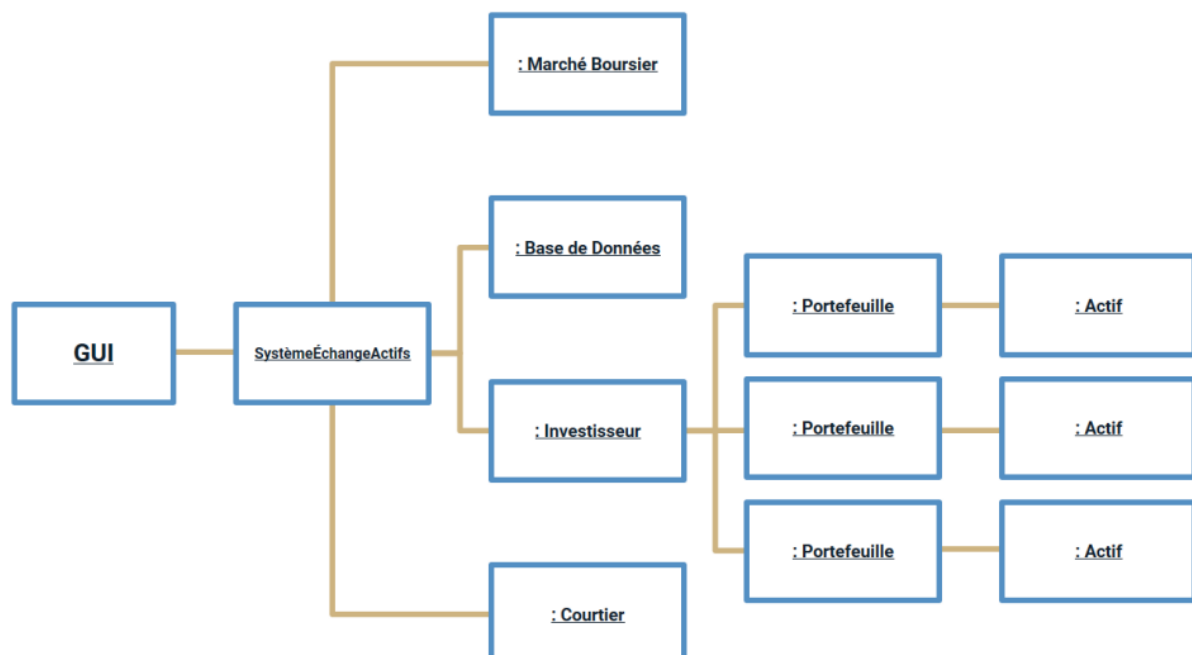




Mise en place de l'instance anonyme de la classe Actif

Voilà notre diagramme d'objets simplement mis en place. Au premier coup d'œil, nous pouvons identifier les différents objets et nous pouvons identifier quelles sont les instances de nos classes.

Nous pourrions même aller un peu plus loin par exemple, dans notre diagramme de classe, un investisseur peut posséder plusieurs portefeuilles, mais nous n'en avons représenté qu'un seul. Il serait alors possible, sur notre diagramme d'objets actuel, de représenter directement cette relation de type One-to-Many (où un investisseur peut être rattaché à plusieurs portefeuilles et plusieurs portefeuilles à un seul investisseur), pour marquer cette relation.



Mise en place de plusieurs portefeuilles pour marquer la relation One-to-Many

À nouveau, dans notre diagramme d'objets actuel, nous n'avons représenté qu'une seule instance d'actif dans le portefeuille de l'investisseur, mais, à nouveau, il est possible de mettre en place pour un seul portefeuille plusieurs instances de classes Actif.

Méthode Des outils pour créer des diagrammes d'objets

Exercice : Quiz

[solution n°2 p.18]

Question 1

On peut représenter une relation many-to-one dans un diagramme d'objets.

- ☐ Vrai
- ☐ Faux

Question 2

Il est impossible d'inclure, dans un diagramme d'objets, des objets qui n'appartiennent à aucune classe d'un diagramme de classe.

- ☐ Vrai
- ☐ Faux

Question 3

Un diagramme d'objets doit forcément contenir les relations précises indiquées dans notre diagramme de classe.

- ☐ Vrai
- ☐ Faux

Question 4

Il est obligatoire de mettre sur papier le scénario de notre diagramme d'objets avant de se lancer dans la conception de celui-ci.

- ☐ Vrai
- ☐ Faux

Question 5

Il est possible de modéliser une seule partie d'un mécanisme si on le souhaite plutôt que de retranscrire l'intégralité du système.

- ☐ Vrai
- ☐ Faux

V. Essentiel

Un diagramme d'objets est un diagramme qui va représenter une classe, mais en ajoutant ses spécificités. Il est utilisé pour montrer un contexte spécifique à un moment précis. À noter que, lorsque l'on parlera d'un diagramme d'objets, il s'agira avant tout d'un outil de documentation et de communication, rien de plus.

Un diagramme d'objets est composé d'objets qui représentent des instances de nos classes et de liens qui représentent des associations. Chaque objet sera représenté dans un rectangle dans lequel apparaîtra son nom souligné, et éventuellement la ou les valeurs d'un ou plusieurs attributs. Nous utiliserons le diagramme d'objets pour donner des exemples ou différents scénarios qui permettront par la suite d'affiner le diagramme de classes.

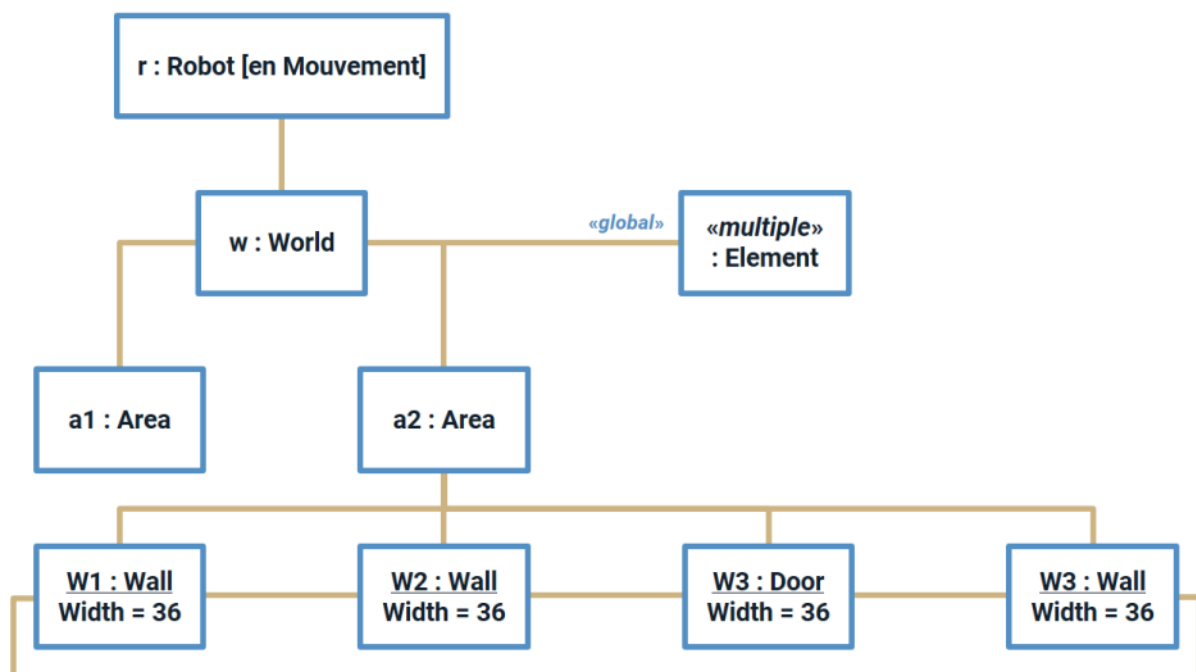
La différence majeure entre les attributs d'une classe au sein d'un diagramme de classes et les attributs d'un objet représentant une instance de classe, dans notre diagramme de classes, seuls les attributs sont listés avec la nature de ceux-ci. Par exemple, le nom de la classe Person peut être une chaîne de caractères de 255 caractères maximum et non nul. Il existe des liens entre les différents objets. Un diagramme d'objets est un diagramme structurel UML qui montre des instances de classificateurs dans des modèles. Dans UML, les diagrammes d'objets fournissent un instantané des instances d'un système et des relations entre les instances. En séquençant les éléments du modèle dans un diagramme de classes, vous pouvez explorer le comportement d'un système à un instant donné.

Pour faire un diagramme, il faut tout d'abord, identifier le mécanisme que vous souhaitez modéliser. Un mécanisme représente certaines fonctions ou certains comportements de la partie du système que vous modélisez qui résultent de l'interaction d'une société de classes, d'interfaces et d'autres éléments. Considérez un scénario qui passe par ce mécanisme. Figer ce scénario à un moment donné et afficher chaque objet qui participe au mécanisme.

VI. Auto-évaluation

A. Exercice

Vous venez d'arriver dans une agence web et la société souhaite observer votre capacité d'analyse d'un diagramme d'objets. À l'aide du diagramme d'objets ci-dessous, vous allez pouvoir répondre aux questions qui suivent.



Question 1

[solution n°3 p.19]

Quel est l'objet représenté dans ce diagramme ? Précisez les différents liens de cet objet à l'aide du diagramme.

Question 2

[solution n°4 p.19]

Expliquez à combien d'instances de Area est relié w.

B. Test

Exercice 1 : Quiz

[solution n°5 p.19]

Question 1

Il est possible de placer des stéréotypes en annotation sur un lien entre deux objets.

- ☐ Vrai
- ☐ Faux

Question 2

Il existe un nommage strict des stéréotypes.

- ☐ Vrai
- ☐ Faux

Question 3

Dans un objet composite, on doit obligatoirement placer des instances de classe.

- ☐ Vrai
- ☐ Faux

Question 4

Il est possible de créer un diagramme d'objets à partir d'un diagramme de déploiement ou même d'un diagramme de cas d'utilisation.

- ☐ Vrai
- ☐ Faux

Question 5

Il est possible d'intégrer directement le nom des fonctions qui sont présentes dans un diagramme de classe au sein d'un diagramme d'objets.

- ☐ Vrai
- ☐ Faux

Solutions des exercices

Exercice p. 9 Solution n°1**Question 1**

Un diagramme d'objets possède beaucoup de similitudes avec un diagramme de classe.

- ☒ Vrai
- ☐ Faux
- ☐ C'est vrai. Les diagrammes de classes et d'objets se ressemblent beaucoup l'un face à l'autre.

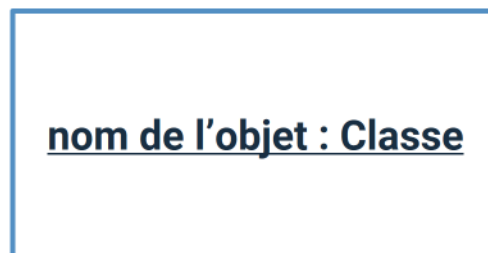
Question 2

Pour représenter une classe dans un diagramme d'objets, il faut passer par une instance de cette classe.

- ☒ Vrai
- ☐ Faux
- ☐ C'est vrai. Il faut passer obligatoirement par une instance de classe, qu'elle soit anonyme ou non, afin de la représenter.

Question 3

Cet objet représente :



- ☒ Une instance nommée d'une classe nommée
- ☐ Une instance anonyme d'une classe nommée
- ☐ Une instance nommée d'une classe anonyme
- ☐ Une collection d'instances anonymes de classe nommée
- ☐ Cet exemple renvoie bien à une instance nommée d'une classe nommée.

Question 4

Si la classe contient des attributs, faut-il assigner des valeurs aux attributs en créant une instance de cette classe ?


- ☐ Non, pas besoin
- ☒ Oui, c'est nécessaire
- ☐ Oui, c'est bien entendu nécessaire d'assigner des valeurs aux attributs d'une instance de classe.

Question 5

Un diagramme d'objets représente un environnement dynamique.

☐ Vrai

☒ Faux

 Faux. Un diagramme d'objets représente un environnement statique. Il fait partie de la catégorie des diagrammes statiques. Il représente notre système à un instant T et ne peut être modifié.


Exercice p. 14 Solution n°2

Question 1

On peut représenter une relation many-to-one dans un diagramme d'objets.

☒ Vrai

☐ Faux


 C'est vrai. En reliant plusieurs instances d'une même classe sur une autre, on peut représenter ainsi une relation de type many-to-one.

Question 2

Il est impossible d'inclure, dans un diagramme d'objets, des objets qui n'appartiennent à aucune classe d'un diagramme de classe.

☐ Vrai

☒ Faux


 C'est faux, il est bien entendu possible d'inclure des objets qui n'appartiennent à aucune classe de notre diagramme de classe. Ce sont des instances de classes anonymes.

Question 3

Un diagramme d'objets doit forcément contenir les relations précises indiquées dans notre diagramme de classe.

☐ Vrai

☒ Faux


 Faux. Bien que ce soit fortement conseillé de représenter graphiquement les relations dans un diagramme d'objets, car cela améliore la vision globale de notre système, ce n'est pas obligatoire.

Question 4

Il est obligatoire de mettre sur papier le scénario de notre diagramme d'objets avant de se lancer dans la conception de celui-ci.

☐ Vrai

☒ Faux


 Faux. Contrairement à ce qu'on pourrait croire, il n'est pas obligatoire de poser à l'écrit notre scénario, mais il est cependant fortement conseillé de le faire afin de savoir exactement où l'on va et comment construire notre diagramme d'objets.

Question 5

Il est possible de modéliser une seule partie d'un mécanisme si on le souhaite plutôt que de retranscrire l'intégralité du système.

☒ Vrai

☐ Faux

 C'est vrai. Si des précisions doivent être apportées à une petite partie d'un mécanisme, il est tout à fait possible de créer un diagramme d'objets afin de mieux comprendre le mécanisme en question.

p. 15 Solution n°3

Comme l'indique cette figure, un objet représente le robot lui-même (r, une instance de Robot), et r est actuellement dans l'état marqué en mouvement. Cet objet a un lien vers w, une instance de World, qui représente une abstraction du modèle de monde du robot. Cet objet a un lien vers plusieurs objets constitués d'instances d'élément, qui représentent des entités que le robot a identifiées, mais pas encore affectées dans sa vue du monde. Ces éléments sont marqués comme faisant partie de l'état global du robot.

p. 16 Solution n°4


À ce moment, w est lié à deux instances de Area. L'un d'eux (a2) est affiché avec ses propres liens vers trois objets Wall et un objet Door. Afin de mieux comprendre, nous les appellerons par le nom français, à savoir un mur et une porte. Chacun de ces murs est marqué avec sa width (largeur) actuelle, et chacun est montré lié à ses murs voisins. Comme le suggère ce diagramme d'objets, le robot a reconnu cette zone fermée, qui a des murs sur trois côtés et une porte sur le quatrième.

Exercice p. 16 Solution n°5**Question 1**

Il est possible de placer des stéréotypes en annotation sur un lien entre deux objets.

☒ Vrai

☐ Faux


 C'est vrai. Il est tout à fait possible de placer des stéréotypes comme « *global* », « *copie* », « *devient* », ou bien d'autres encore.

Question 2

Il existe un nommage strict des stéréotypes.

☐ Vrai

☒ Faux


 Faux. Il n'existe pas forcément de nommage strict. Il est tout à fait possible de nommer un stéréotype selon une action exécutée entre deux objets.

Question 3

Dans un objet composite, on doit obligatoirement placer des instances de classe.

☒ Vrai

☐ Faux


 C'est vrai. Petit piège, peu importe ce qui compose notre diagramme d'objets, ce seront toujours des instances de classe.

Question 4

Il est possible de créer un diagramme d'objets à partir d'un diagramme de déploiement ou même d'un diagramme de cas d'utilisation.

☒ Vrai

☐ Faux


 C'est vrai. On peut créer des diagrammes d'objets en instanciant les classificateurs dans des diagrammes de classes, de déploiement, de composants et de cas d'utilisation.

Question 5

Il est possible d'intégrer directement le nom des fonctions qui sont présentes dans un diagramme de classe au sein d'un diagramme d'objets.

☐ Vrai

☒ Faux

 Faux. Bien que les fonctions puissent être utiles dans un diagramme de classe, ce n'est pas l'objectif recherché dans le diagramme d'objets.