

Logique de description II

Exercices

Cette semaine nous allons continuer notre apprentissage de la logique de description avec l'outil **Protégé** et le vérificateur **Hermit**.

1. Pour débiter, complétez les étapes suivantes.

- (a) Pour les *concepts* (**Classes**) :

- i. Ajoutez une *sous-classe* **Personne** à **Thing** (**T**).

- ii. Ajoutez une *sous-classe* **Homme** et une autre **Femme** à la classe **Personne**.

- (b) Pour les *rôles* (**Object Properties**) :

- i. Ajoutez le rôle **estUnDesParentsDe**,

- (c) Définissez (*equivalent to, class expression editor*) maintenant la sous-classe **Parent** de la classe **Personne** par l'expression

estUnDesParentsDe some Personne

$(\exists \text{ estUnDesParentsDe } \textit{Personne}).$

- (d) Ajoutez une classe **MereMonoparentale** en vous assurant qu'une **MereMonoparentale** soit une **Femme** *et* un **Parent** (à droite **SubClass Of**).

- (e) Ajoutez finalement une sous-classe **Mere** de **Femme** *définie* par

Femme and Parent

- (f) Inspecter la hiérarchie de classes et remarquez que la classe **MereMonoparentale** n'est pas placée sous la classe **Mere**.

- (g) Sauvegardez et démarrez **Hermit** et observez que la classe **MereMonoparentale** est bien placée sous la classe **Mere** *dans la hiérarchie inférée*.

2. Définissez une sous-classe **ParentDUneFille** de la classe **Personne** et vérifiez que le raisonneur la placera bien sous la classe **Parent** (lorsqu'il sera resynchronisé).

3. Définissez une classe **GrandParent** comme la classe des personnes qui sont des parents de parents. Vérifiez que le raisonneur place bien cette classe sous **Parent**.

4. Est-ce qu'une sous-classe **Parent2** de la classe **Personne** définie par l'expression

estUnDesParentsDe some owl:Thing

serait équivalente à la classe **Parent** ? Vérifiez avec l'outil ! Pouvez-vous expliquer la réponse obtenue ?

-
5. Si vous ajoutez maintenant une sous-classe `Parent3` à la classe `Parent` définie par l'expression

`estUnDesParentsDe some owl:Thing`

quelles seront les relations entre les classes `Parent`, `Parent2` et `Parent3`? Vérifiez avec l'outil! Pouvez-vous expliquer la réponse obtenue?