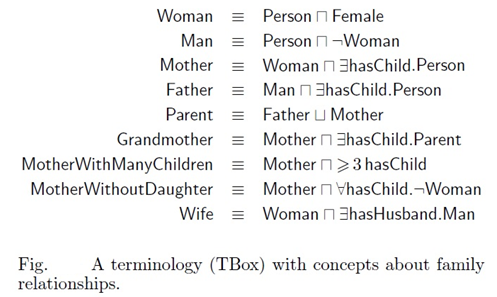
**Лабораторная работа № 7.**

**Создание Web-онтологии на языке SWRL**

*Цель работы*: изучение основных элементов SWRL-онтологий, освоение навыков работы в системе Protégé, разработка простой SWRL-онтологии семейных отношений.

Порядок выполнения работы:

0) Изучить онтологию семейных отношений, заданную формулами дескриптивной логики. Данная онтология задается на уровне классов (*TBox*, терминологический словарь).



1) Создать классы онтологии (*Classes*). Следует заметить, что базовыми классами будут классы *Person* и *Female*. Остальные классы будут вычисляемыми.

2) Создать отношения онтологии (*Object Properties*, объектные свойства).

3) Создать *SWRL*-правила для каждого вычисляемого класса, используя встроенный редактор правил. Чтобы добраться до редактора правил, следует пройтись по цепочке меню *Window*->*Views*->*Ontology views*->*Rules*.

4) Разработать простое генеалогическое дерево.

5) В соответствии с этим деревом, создать экземпляры классов и отношений (*individuals, class assertion, object property assertion*).

6) Протестировать созданную онтологию, используя встроенный ризонер (*reasoner*) *Hermit*. При тестировании решается задача классификации, когда экземпляры автоматически распределяются по классам. Сопоставить полученную классификацию с исходным генеалогическим деревом.

7) Просмотреть и изучить сгенерированный код *OWL*/*SWRL*-онтологии в различных форматах (*OWL*/*XML*, *RDF*/*XML*, *OWL functional syntax*), используя цепочку меню: *Window*->*Views*->*Ontology views.*

8) Сгенерировать визуальное представление онтологии, используя плагин *OntoGraf*. Для активизации данного плагина использовать цепочку меню: *Window*->*Tab*->*OntoGraf.* Построить несколько визуальных представлений: а) только на уровне классов; б) только на уровне экземпляров; в) смешанное представление, использующее классы и экземпляры.

9) Сохранить разработанную онтологию на диске в формате *RDF*/*XML*.

10) Составить отчет о проделанной работе в Word.

Методические указания

Для ускорения выполнения работы № 2 рекомендуется использовать онтологию из работы № 1 «Создание Web-онтологии на языке OWL», из которой удалены только аксиомы, а остальные элементы оставлены без изменения. Большинство шагов в обеих лаб. работах совпадают, за исключением шага 3.

*Вариативное задание*. Добавить в онтологию описание следующего родственника:

|  |  |
| --- | --- |
| **Вариант** | **Тип родственника** |
| 1 | дядя |
| 2 | тетя |
| 3 | племянник |
| 4 | племянница |
| 5 | двоюродный брат |
| 6 | двоюродная сестра |
| 7 | правнук |
| 8 | прабабушка |
| 9 | свояк |
| 10 | свояченица |
| 11 | тесть |
| 12 | свекор |
| 13 | потомок (любого уровня) |
| 14 | предок (любого уровня) |

*Примечание*: Для определения родственников №№ 1-8 необходимо ввести и использовать объектное свойство hasParent, которое является обратным по отношению к свойству hasChild. При определении характеристик свойства hasParent в Protégé в поле Inverse of необходимо ввести hasChild.

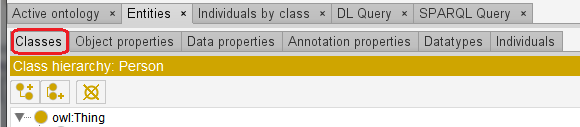
**ПРИЛОЖЕНИЕ 1.**

**Основные сведения по работе в системе Protege**

**Создание классов (Classes)**

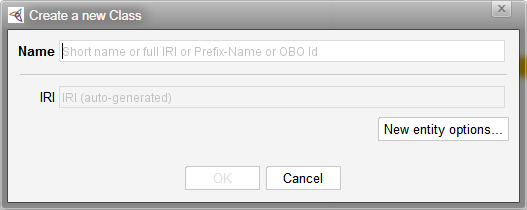
Запускаем Protégé

Входим во вкладку Entities->Classes



Первоначально в системе только один общий класс owl:Think. Он включает в себя абсолютно все существующие (и несуществующие) в мире классы.

Для создания нового класса надо навести курсор на класс owl: Think и нажать кнопку . Появится окно для ввода имени класса.



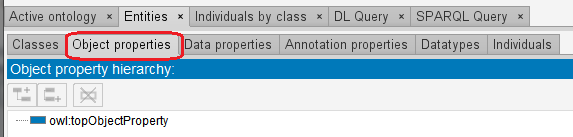
Рекомендуется использовать только латинские буквы, а первую букву лучше делать заглавной (это общепринятые правила хорошего тона).

Для удаления класса надо навести курсор на удаляемый класс и нажать кнопку 

**Создание объектных свойств (Object properties)**

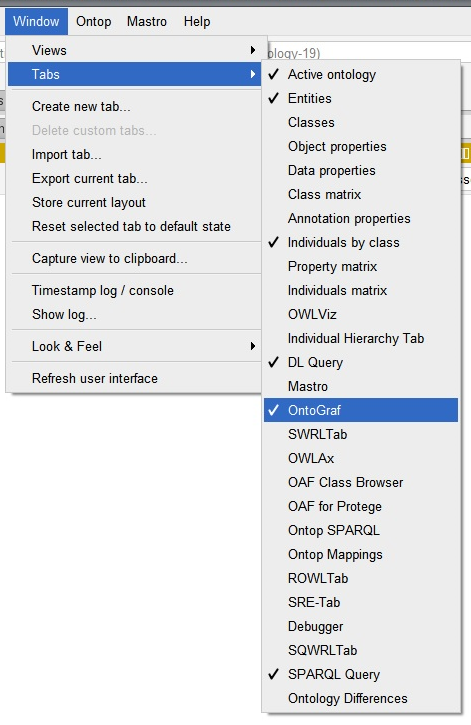
Объектные свойства определяют *отношения* между классами.

Входим во вкладку Entities-> Object properties



Первоначально в системе только одно общее объектное свойство owl:Think. Оно включает в себя абсолютно все существующие (и несуществующие) в мире объектные свойства.

**Как запустить плагин OntoGraf**



**Как запустить редактор SWRL-правил**

