МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки та програмної інженерії

Кафедра інженерії програмного забезпечення

Лабораторна робота № 1

з дисципліни

«Технології доменної інженерії»

**Виконав:** *студент гр. ПІ-122М*

*Барило Олександр Григорович*

**Викладач:** *Олександр Андрійович Ткаченко.*

**Київ 2024**

# Мета роботи

# Створити онтологічну модель у редакторі Protégé, що містить класи / підкласи, індивіди, властивості та обмеження у відповідності до своєї предметної області. Використовувати Reasoner для перевірки онтології та виведення нових знань.

# **Завдання**

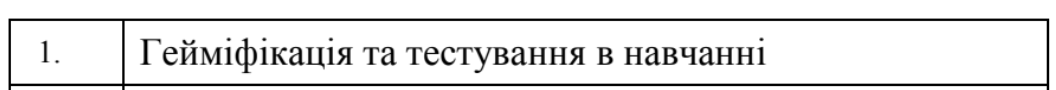
Встановіть Protege версії 5.x (http://protege.stanford.edu) та запустіть його. Створіть

онтологічну модель у відповідності до своєї предметної області (див. далі Варіанти). Збережіть проект у своєму .owl-файлі на локальному диску. Для створення моделі виконайте такі дії:

1. Створіть ієрархію класів (не менше 5 класів і підкласів).
2. Створіть не менше 2 індивідуалів (екземплярів) для кожного класу.
3. Створіть принаймні один клас шляхом перерахування всіх індивідуалів (екземплярів) класу.
4. Встановіть властивість Disjoint with.
5. Створіть не менше 4 об’єктних властивостей.
6. Створіть не менше 4 властивостей даних.
7. Створіть не менше 2 додаткових характеристик об’єктних властивостей: Functional, Inverse, Inverse Functional, Transitive, Symmetric, Asymmetric, Reflexive, Irreflexive Property, Property chains.
8. Вкажіть обмеження Domain та Range для об’єктних властивостей та властивостей даних.
9. Створіть кванторні обмеження (Quantier Restrictions) з використанням їх видів existential (ключове слово some) та universal (ключове слово any) для об’єктних властивостей.
10. Створіть обмеження кардинальності (Cardinality Restrictions) (ключові слова min, max, exactly) для об’єктних властивостей.
11. Створіть обмеження кардинальності (<, >, =, <=, >=) для властивостей даних.
12. Створіть обмеження на значення (hasValue Restrictions) з ключовим словом value.
13. Створіть обмеження з використанням операцій над множинами: об’єднання (or), перетин (and), доповнення (not) – як мінімум по одному обмеженню на кожну операцію.
14. Створіть аксіоми еквівалентності для класів (Equivalent To).
15. Створіть для декількох класів та індивідуалів Анотації (Annotation) типу «коментар» (comment) та «мітка» (label) на англійській та українській мовах, щоб додати додаткову текстову інформацію до цих сутностей.
16. Використовуйте Reasoner для виявлення нових знань та перевірки онтології на невідповідність.
17. Вивести в графічному вигляді ієрархію класів за допомогою OWLViz або OntoGraph.

# Виконання

Варіант 1:



1. У редакторі Protege створимо наступу ієрархію класів:

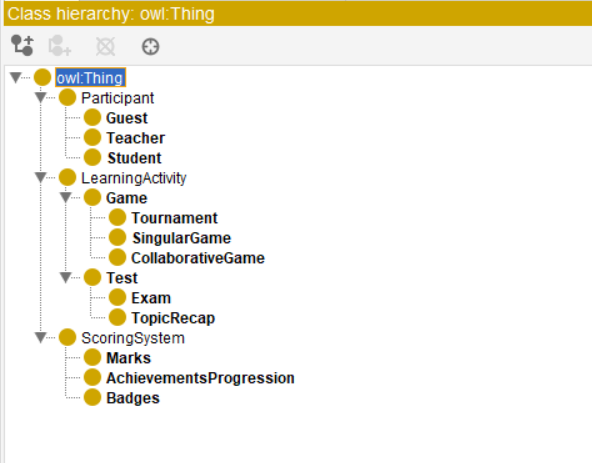
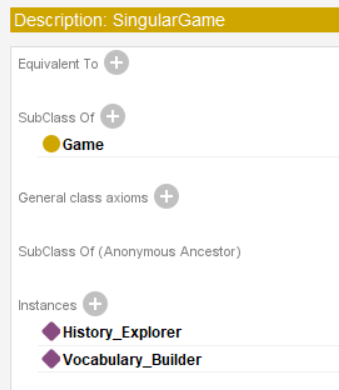
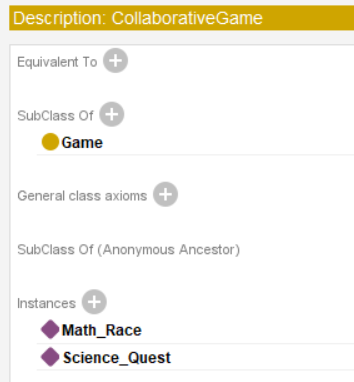
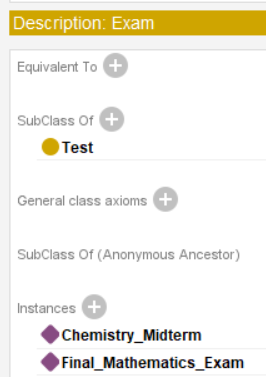
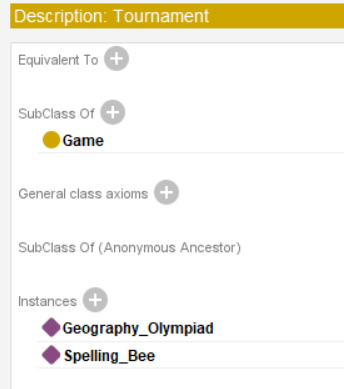
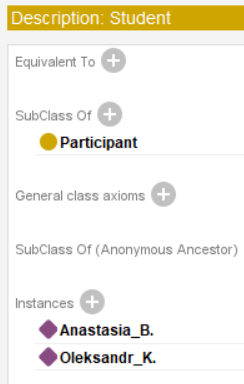
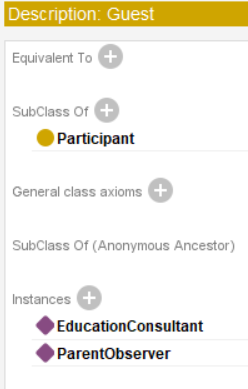
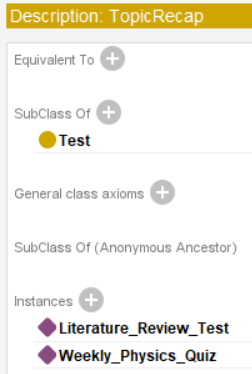


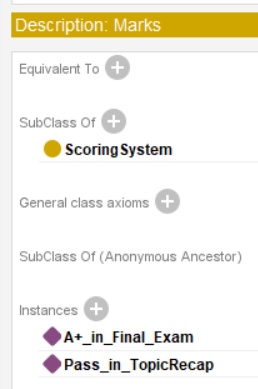
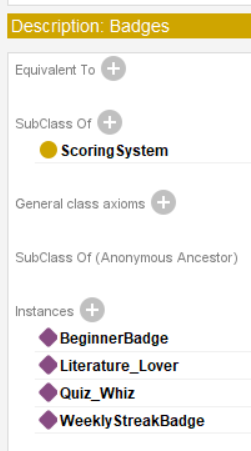
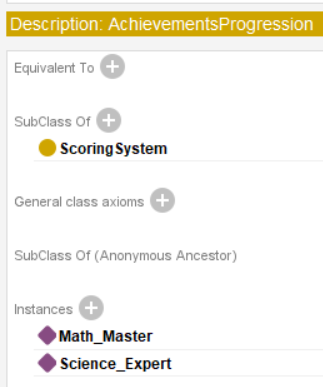
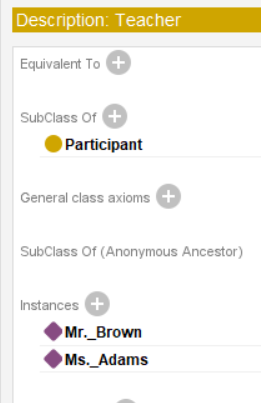
Рис. 1. Ієрархія класів Protege

1. Створимо не менше двох екземплярів кожного класу:

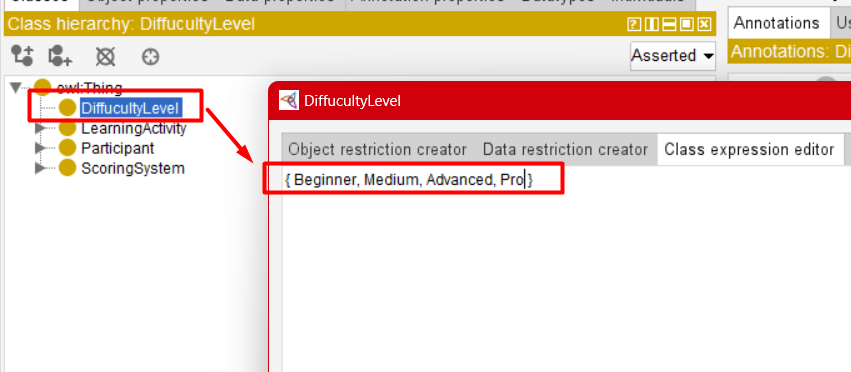


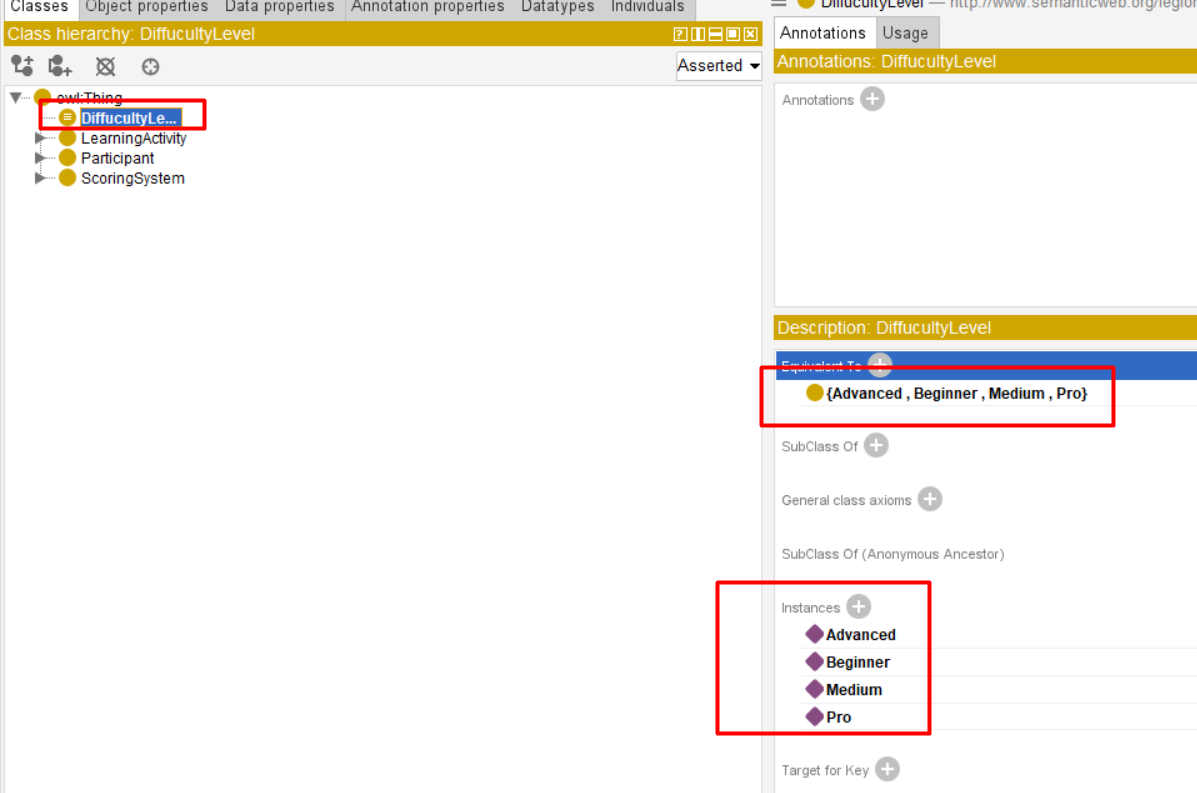




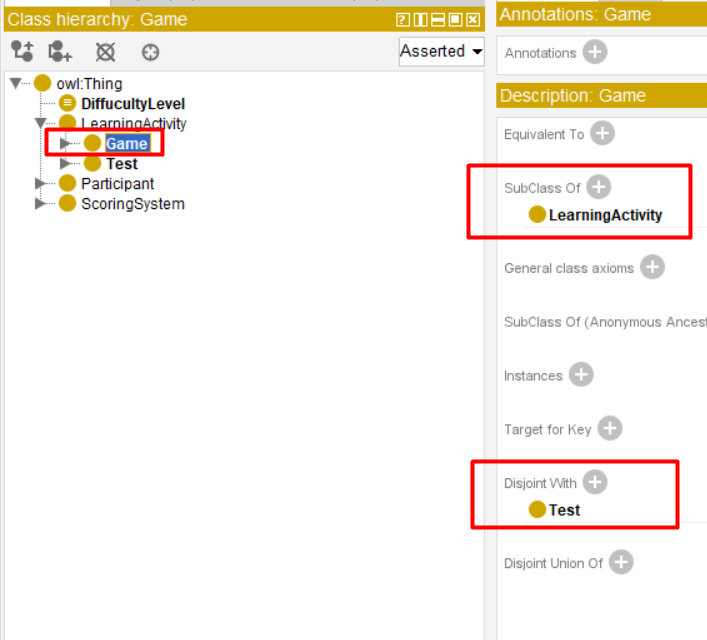


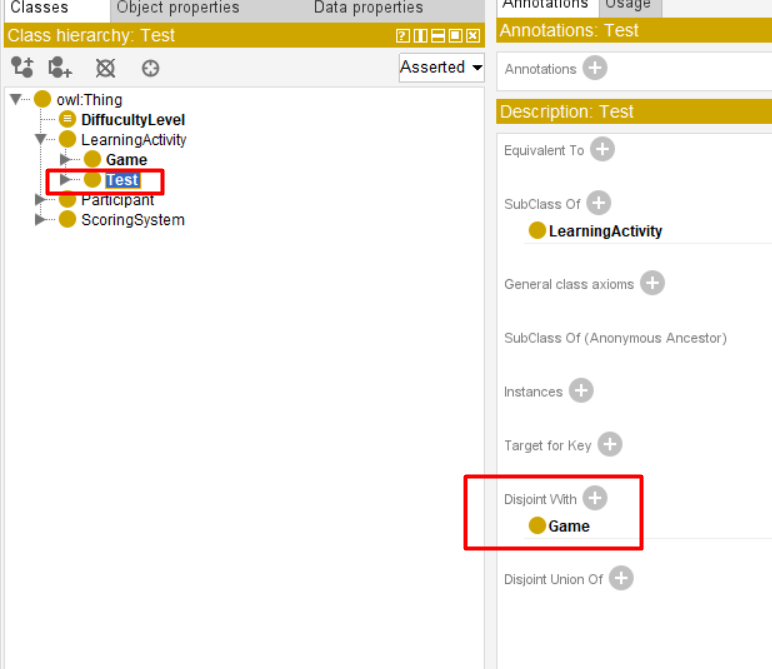
1. Створимо клас шляхом перерахування всіх індивідуалів (екземплярів) класу:



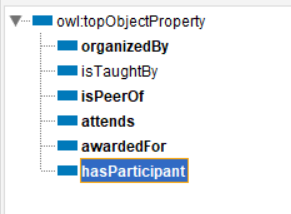


1. Встановимо властивість Disjoint With для класів Game та Test:

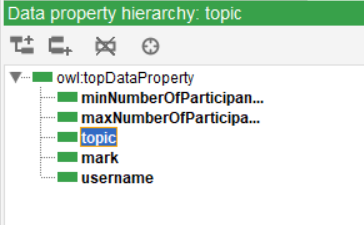




1. Створимо об’єктні властивості:

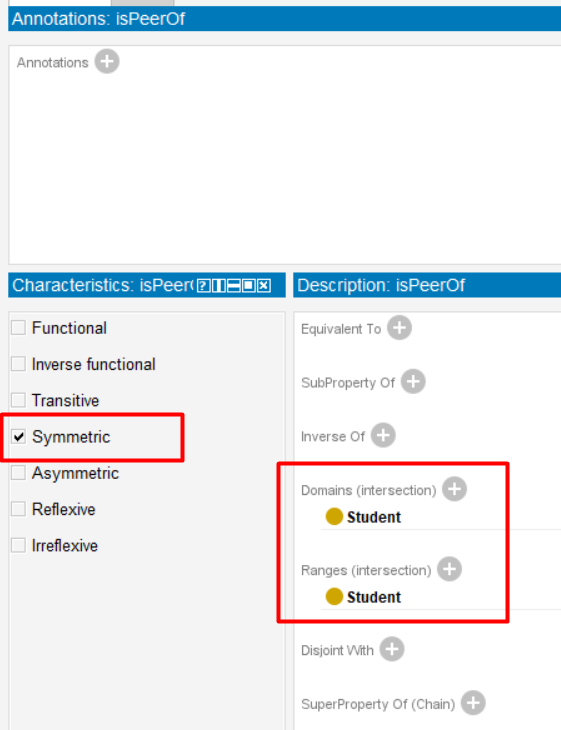


1. Створимо властивості даних:

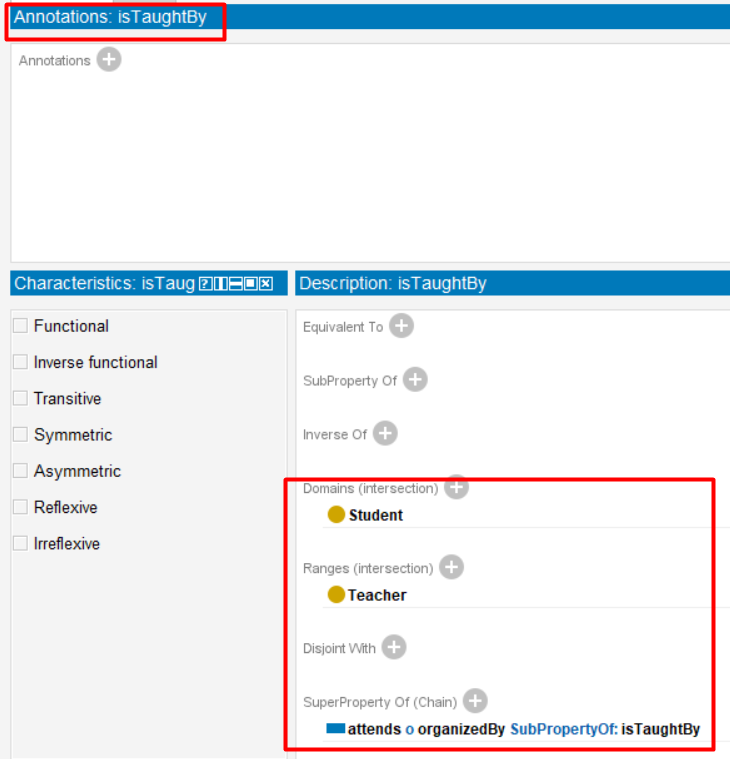


1. Створимо додаткові характеристики об’єктних властивостей:

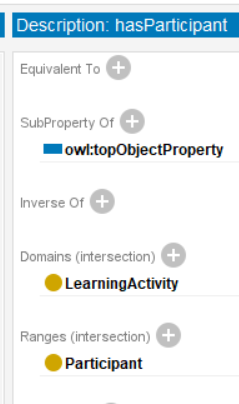
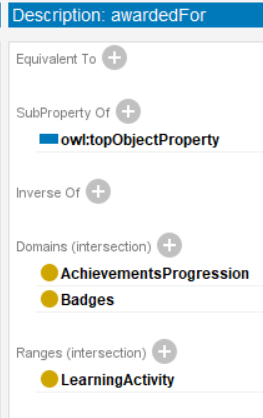
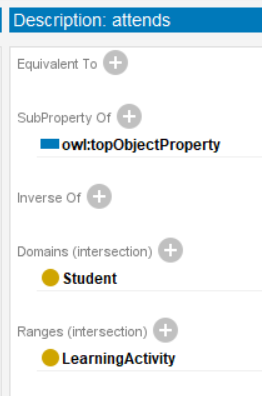
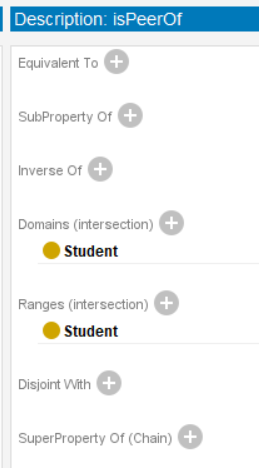
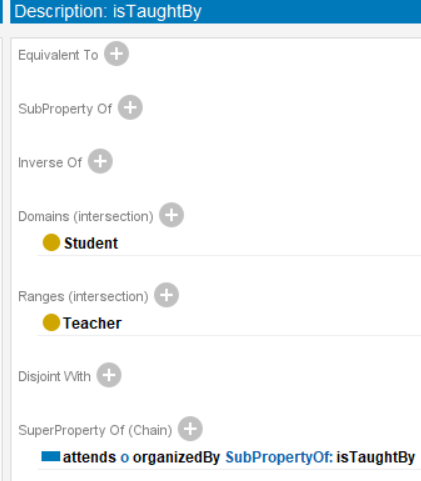
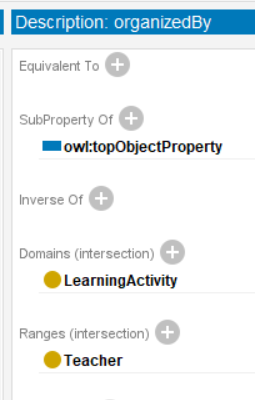
Симетрична властивість isPeerOf вказує, що якщо студент є одногрупником іншого студента, то інший студент також є одногрупником першого студента:

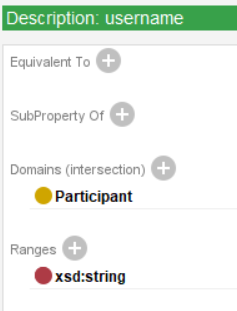
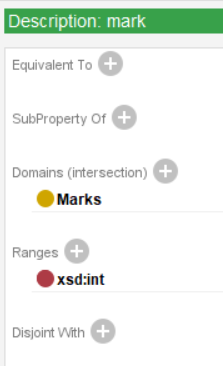
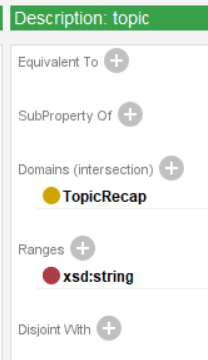
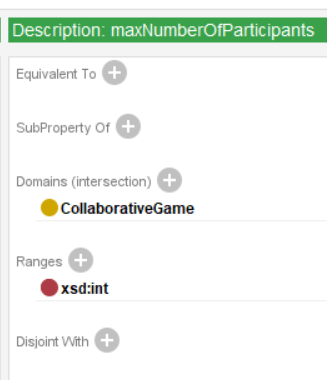
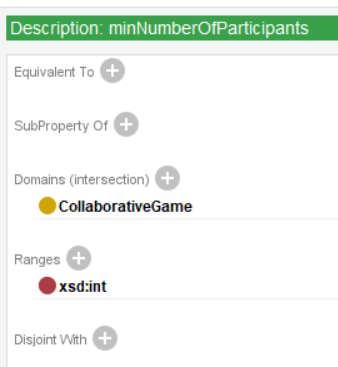


Ланцюг властивостей isTaughtBy вказує, що якщо студент відвідує навчальну активність, організовану певним вчителем, то він є учнем цього вчителя:



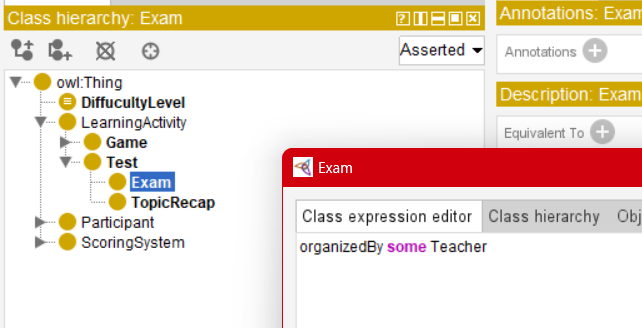
1. Вкажемо обмеження Domain та Range для об’єктних властивостей та властивостей даних:



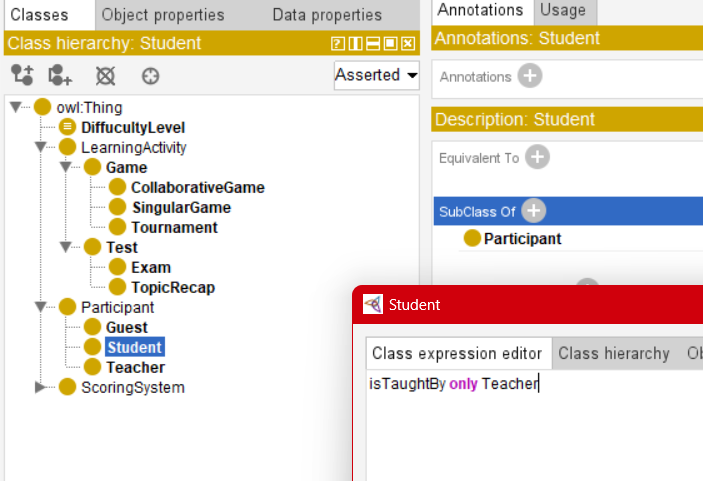


1. Створіть кванторні обмеження (Quantier Restrictions) з використанням їх видів existential (ключове слово some) та universal (ключове слово any) для об’єктних властивостей:

Екзистенційне обмеження, що вказує на те, що екзамен повинен проводитись принаймні одним вчителем:

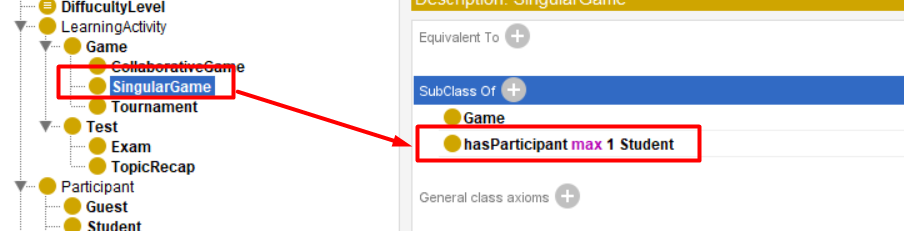


Універсальне обмеження, що вказує на те, що тільки вчителі можуть навчати студентів:

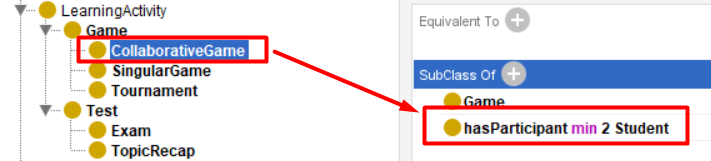


1. Створимо обмеження кардинальності (Cardinality Restrictions) (ключові слова min, max, exactly) для об’єктних властивостей:

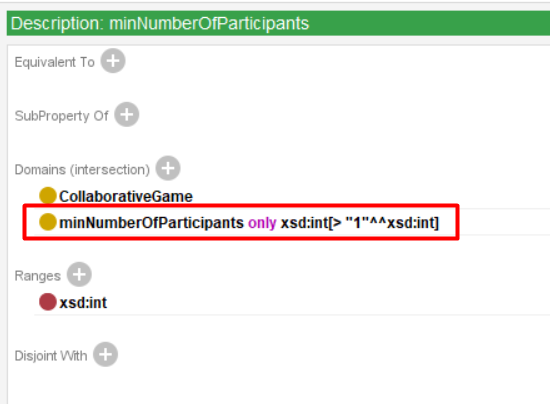
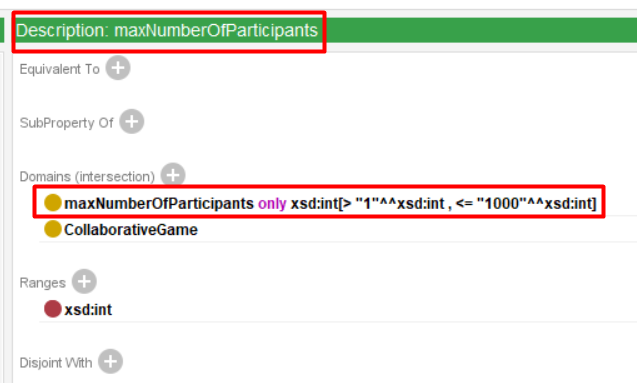
Обмеження кардинальності, що обмежує одиночну гру до лише одного учасника:

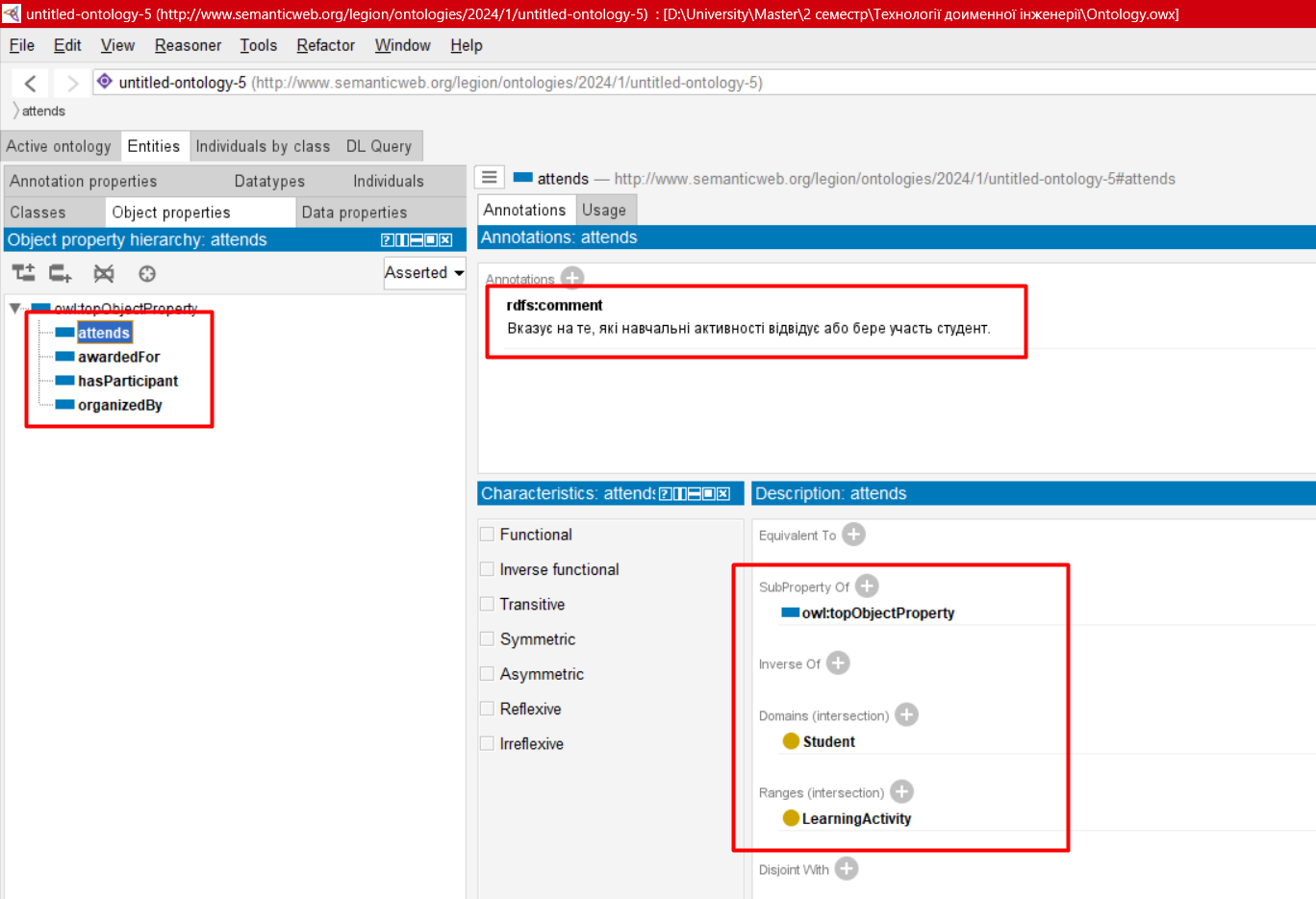


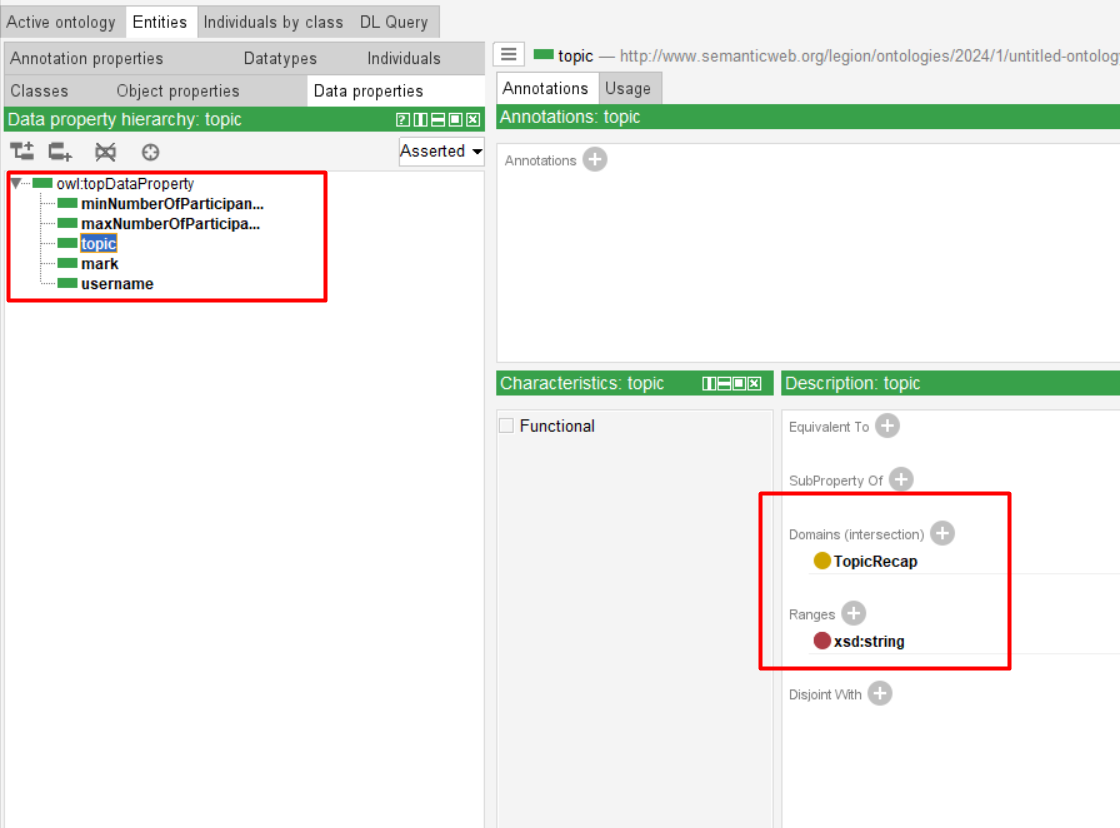
Обмеження кардинальності, що обмежує командну гру мінімум двома учасниками:



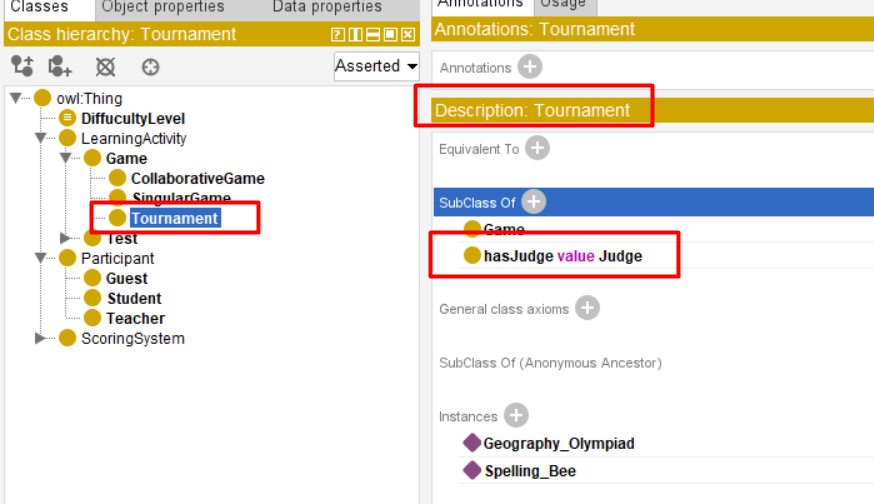
1. Створимо обмеження кардинальності (<, >, =, <=, >=) для властивостей даних:



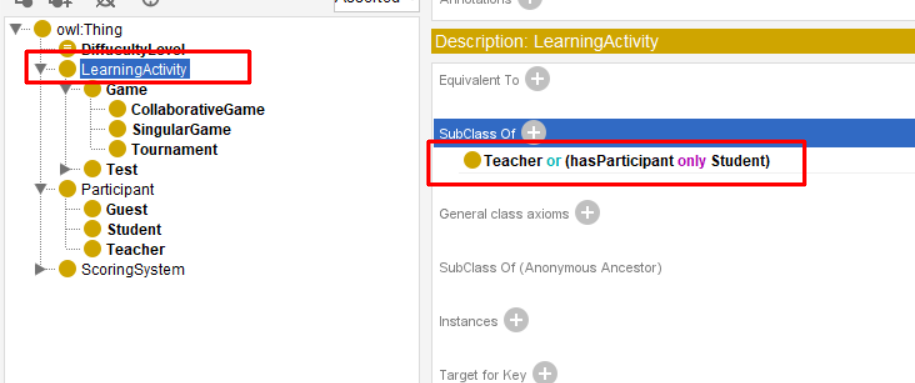


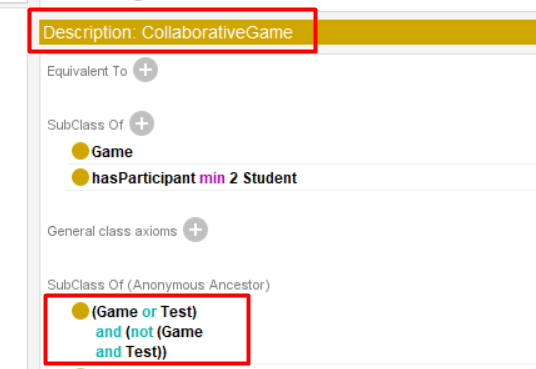
Властивості даних  


1. Створимо обмеження на значення (hasValue Restrictions) з ключовим словом value:

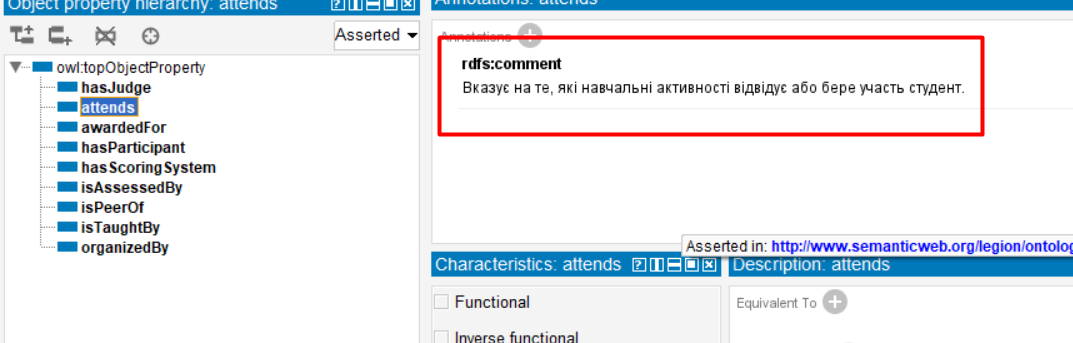


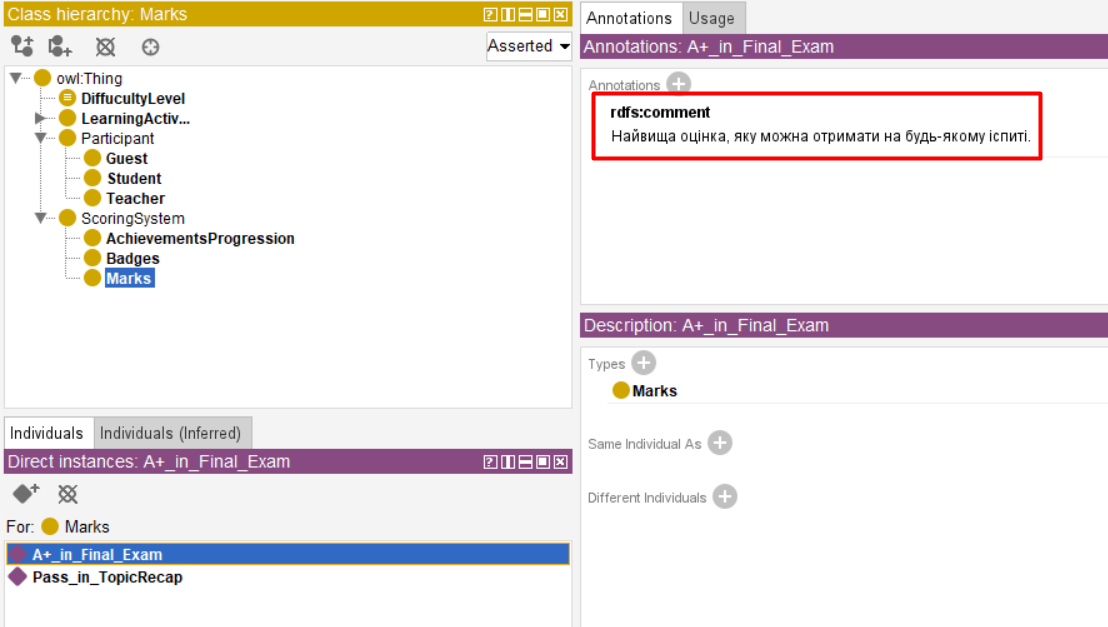
1. Створимо обмеження з використанням операцій над множинами: об’єднання (or), перетин (and), доповнення (not):



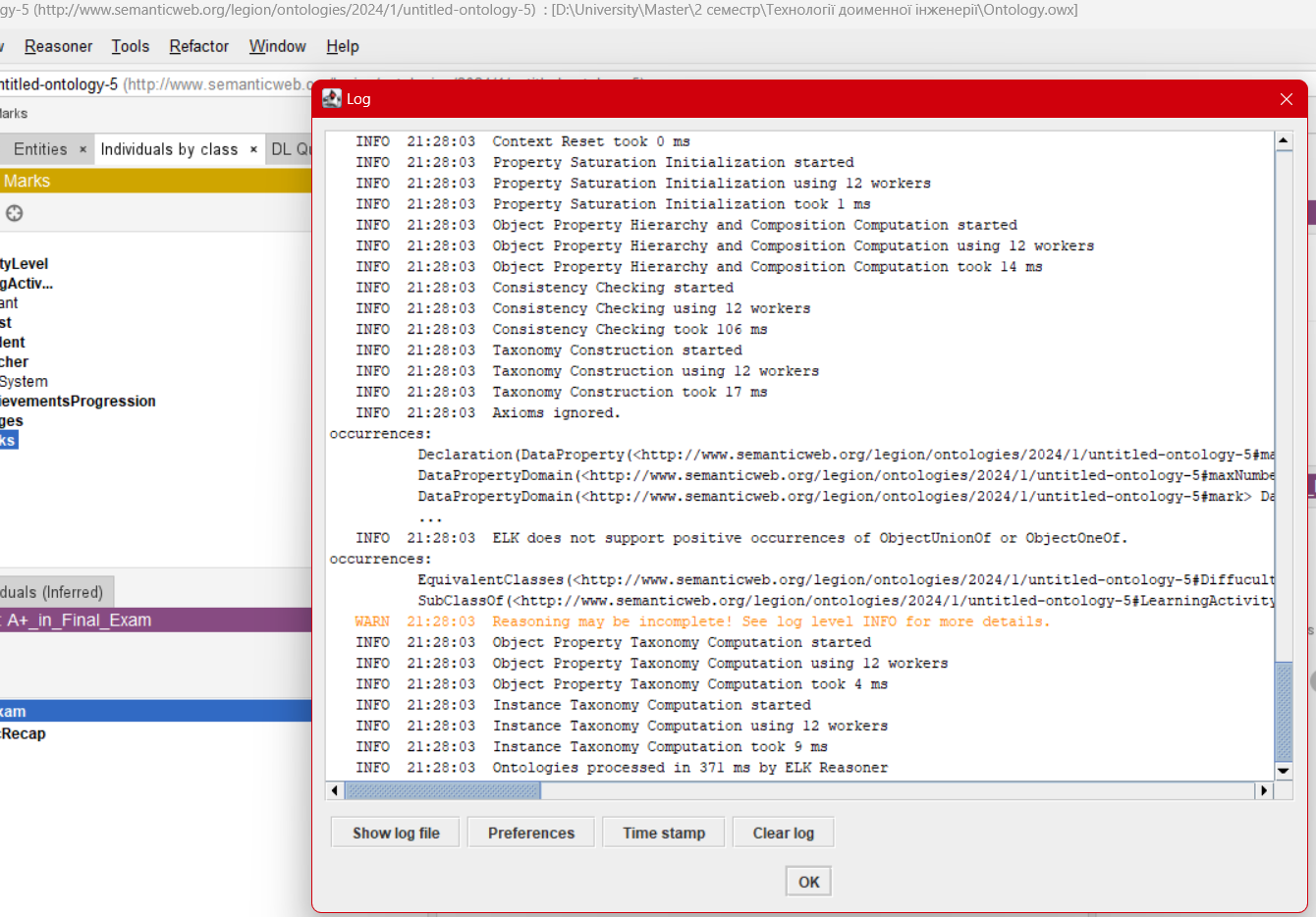


1. Створимо аксіоми еквівалентності для класів (Equivalent To):
2. Створимо для декількох класів та індивідуалів Анотації (Annotation) типу «коментар» (comment) та «мітка» (label) на англійській та українській мовах, щоб додати додаткову текстову інформацію до цих сутностей:





1. Використаємо Reasoner:



1. Використаємо OntoGraf для візуалізації онтології:

