

Курсова робота за темою: «Реалізація бази знань за допомогою системи PROTEGE»

Роботу виконав Вавдійчик Віктор

Онтологічне моделювання

1. Задачі онтологічного моделювання

На сьогоднішній день, більшість систем бізнес-аналізу, насправді, побудовані на основі реляційних баз даних.. Проте зростає потреба в системах, які здійснюють інтерпретацію даних, а не тільки їх зберігання та обробку. Тут в гру вступають семантичні технології, які пропонують новий підхід до моделювання даних, який ґрунтується на онтологічних моделях.

- Головна задача для онтологічного моделювання є забезпечення однозначної інтерпретації та розуміння даних. В цьому полягає їх відмінність від реляційних баз даних, що семантичні моделі дозволяють включати у себе не тільки дані, але й їхню семантику та взаємозв'язки між ними. Це дає змогу зберігати та обробляти не просто дані, але інформацію, з якою можна взаємодіяти, шукати залежності та отримувати нові знання.
- Іншим важливим задаванням онтологічного моделювання є забезпечення інтеграції даних з різних джерел та їх стандартизація.
- Ще одною задачею онтологічного моделювання є забезпечення можливості автоматичної обробки великих об'ємів складно структурованих даних та їхньої інтерпретації. Семантичні технології можуть створювати програмні платформи, які здійснюють автоматичне зчитування та обробку даних, зокрема використовуючи машинне навчання та штучний інтелект.
- Крім того, онтологічне моделювання допомагає вирішити проблему неоднозначності та нечіткості даних. Семантичні технології дозволяють описувати дані за допомогою онтологічних термінів та надають можливість автоматичної інтерпретації даних відповідно до заданих правил та контексту.

2. Формальний опис онтологічної моделі

Модель – це інформаційне представлення будь-якої сукупності об'єктів та явищ реального світу, що характеризується:

- Спрощенням
- Концептуалізованістю – кожен об'єкт, явище, властивість виражені за допомогою понять абстракції різного рівня.
- Взаємопов'язаністю – всі елементи моделі пов'язані між собою.
- Наявністю логічних правил взаємодії елементів.
- Прогностичним потенціалом – ми можемо виконати уявний експеримент, додавши до моделі ті чи інші параметри чи події, і за допомогою зазначених вище логічних правил зробити висновок про те, що відбудеться в тому чи іншому випадку.

Процес створення формальної онтологічної моделі

Цей процес іде за наступною послідовністю:

- 1) Декомпозиція – виділення сутностей, які приймають участь в моделі.
- 2) Ідентифікація – створення індивідуальних сутностей.
- 3) Класифікація – створення класів, які відповідають групам сутностей. Тобто, додання сутностей в певний клас.
- 4) Опис властивостей – визначення способів задання інформації про характеристики і відносини сутностей.
- 5) Значення, зв'язки – присвоєння значень властивостям, створення зав'язків.

3. Особливості реалізації онтологічної моделі

Створення класу

Сутності в онтологічних моделях представляються у вигляді триплету. Отже, факт існування класу буде представлений наступним чином:

Суб'єкт	Предикат	Об'єкт
Унікальний ідентифікатор класу	Має тип	Клас

В RDF файлі це буде виглядати наступним чином:

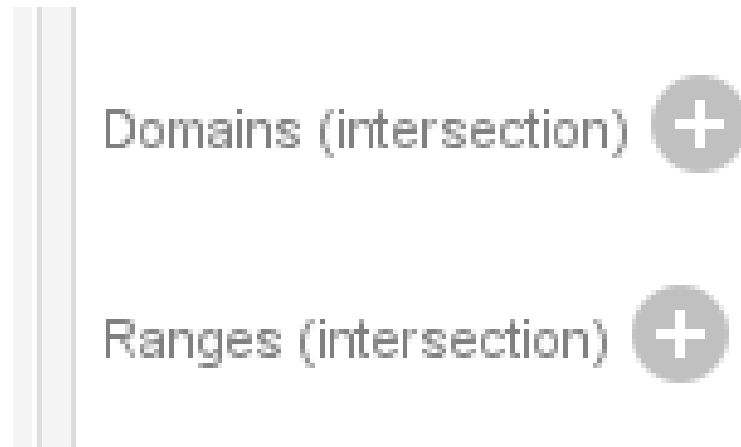
```
<owl:Class  
rdf:about="http://www.semanticweb.org/vavdi/ontologies/2023/  
4/untitled-ontology-3#Product"/>
```

Створення властивості

Щоб створити властивість, значення якої це літерал, потрібно перейти в «Data properties» та натиснути «Add sub property». В правій частині вікна буде видно кнопки Domain і Range.

Domain – це клас, до екземплярів якого можна застосувати ця властивість.

Range – це тип значень, які властивість може набувати.



Data property

Було створено демонстративну властивість `productName`, яка застосовується до екземплярів класу “`Product`” і має тип значення `string`.

Ось так це представлено в RDF файлі:

```
<owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-
3#productName">
  <rdfs:domain
rdf:resource="http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-
3#Product"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

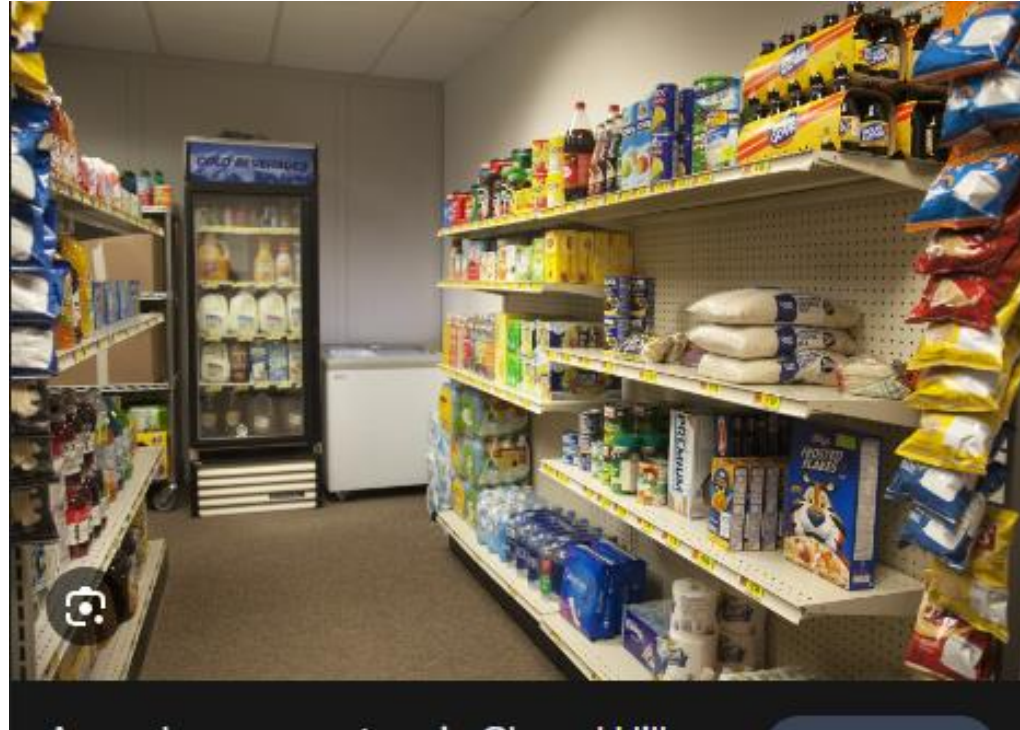
Object property

Далі було створено властивість, яка відображає відносини між об'єктами в онтології (Object properties). Для цього я перейшов у вкладку «Object properties» та натиснув кнопку «Add sub property». Я створив властивість “isLocatedIn”. В Domains я вказав клас Product, а в Ranges вказав Store. Тобто виходить така властивість: Product isLocatedIn Store. В RDF це виглядає наступним:

```
<owl:ObjectProperty
  rdf:about="http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#isLocatedIn">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Product"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Store"/>
</owl:ObjectProperty>
```

Реалізація онтології за допомогою редактора Protégé.

1. Загальний опис онтології «Grocery store»



Данна онтологія призначена для пошуку товарів в магазині, їхніх характеристик, пошуку товарів за їхніми характеристиками або за їх категорією, а також для пошуку куплених товарів деяким покупцем.

2. Реалізація онтології «Grocery store» в редакторі Protégé.

Для реалізації онтології «Grocery store» спочатку була реалізована ієрархія класів, яка зображена на Рис. 2.1.

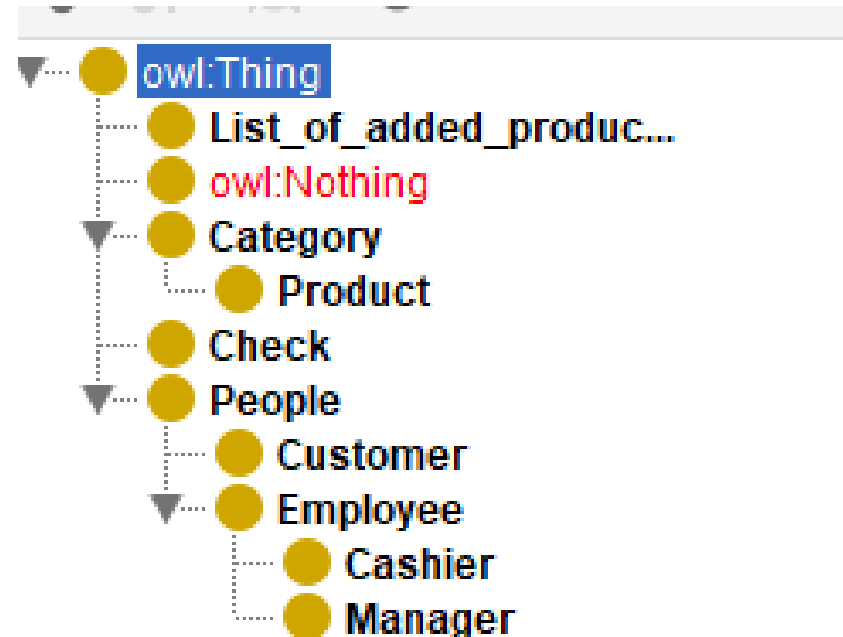
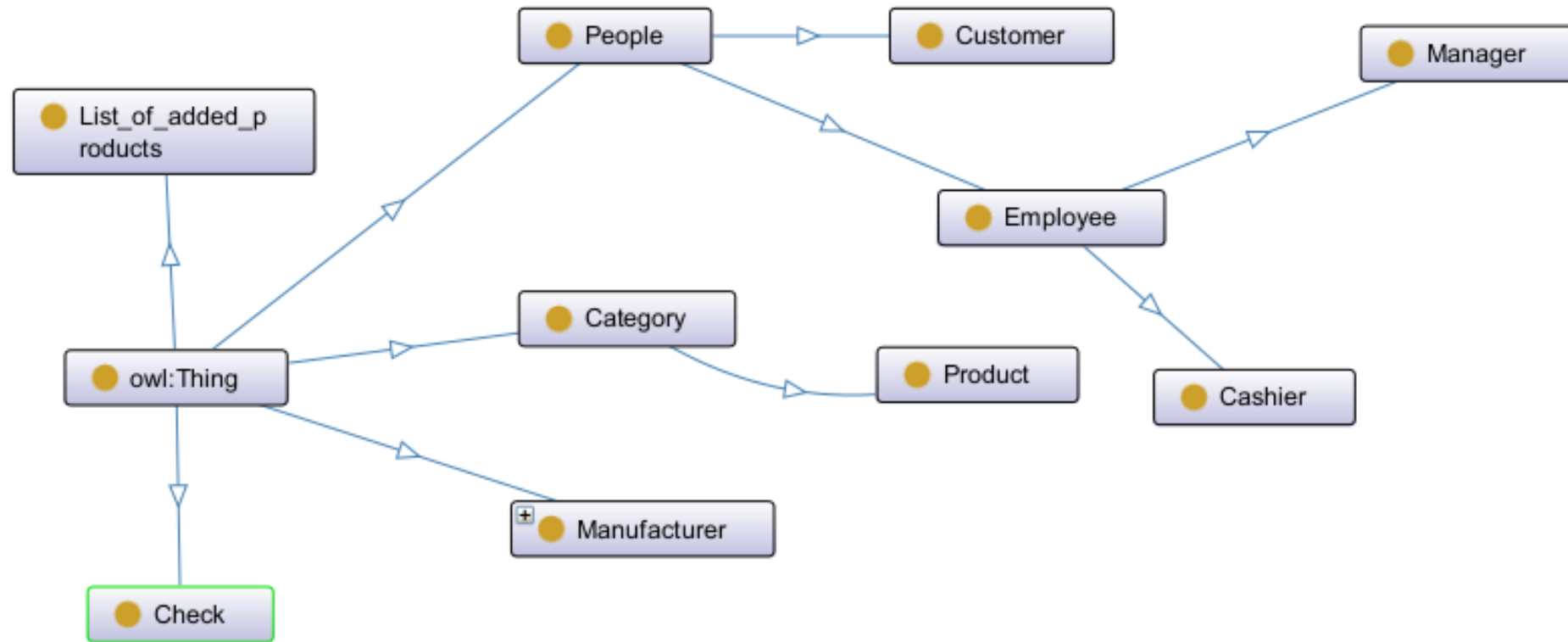


Рис. 2.1.



Список всіх object properties надано на Рис. 2.3.

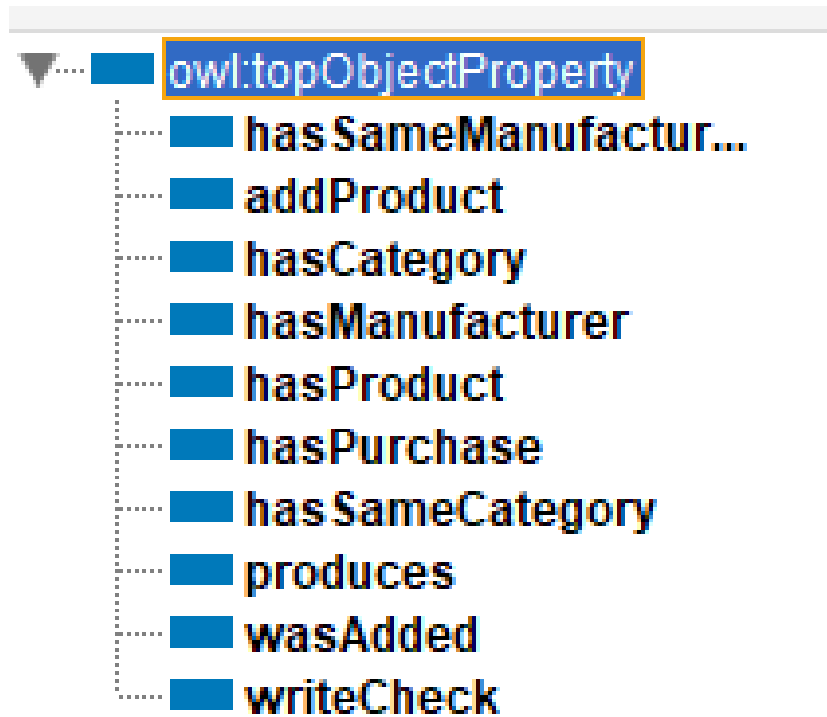


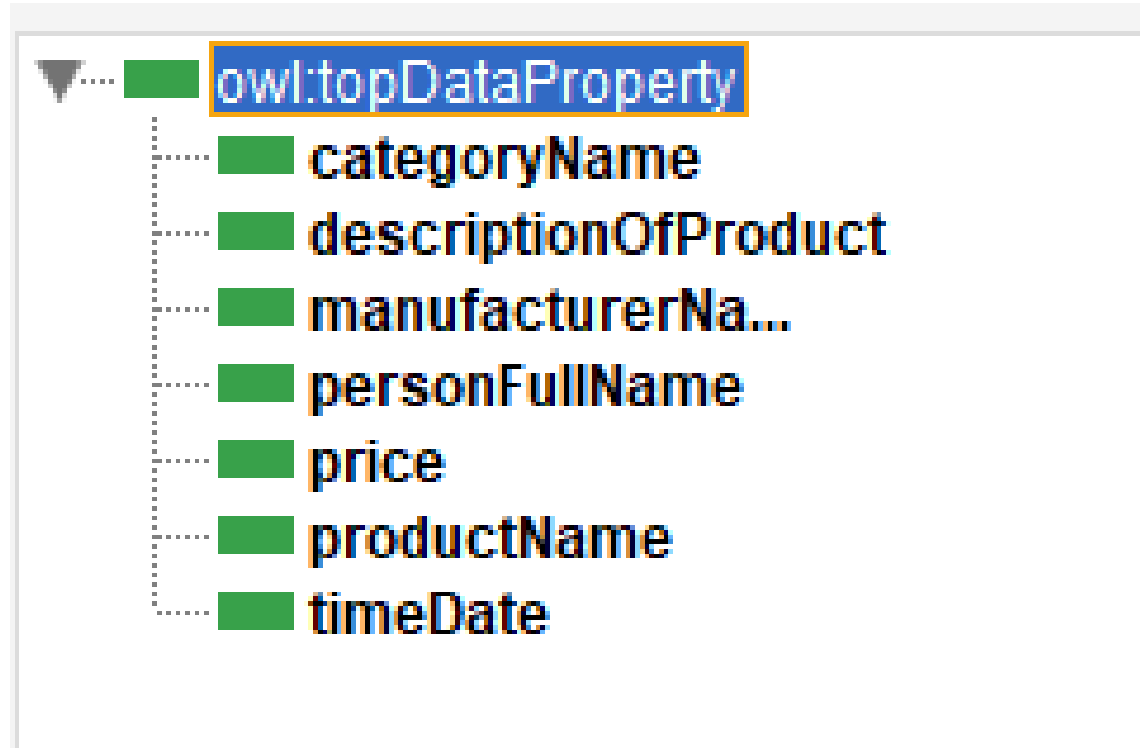
Рис. 2.3

Отже, описуючи object properties, які були представлені на Рис. 2.3, можна зазначити наступне:

- wasAdded позначає властивість, яка зв'язує клас List_of_added_products та клас Product. Тобто, доменом цієї властивості є List_of_added_products, а діапазоном значень є Product.
- hasSameCategory позначає властивість, яка зв'язує продукти однієї категорії. Тобто, доменом цієї властивості є Product і діапазоном значення є, також, Product. Ця властивість симетрична транзитивна.
- hasManufacturer позначає властивість, яка зв'язує продукт та виробника цього продукту. Тобто, доменом цієї властивості є клас Product, а діапазоном значень є Manufacturer. Ця властивість також є функціональною, адже у конкретного продукту буде лише один виробник.
- produces позначає властивість, яка зв'язує виробника та продукти, які він виготовив. Ця властивість є оберненою до властивості hasManufacturer. Позначається це через “InverseOf”, як показана на Рис. 2.4

- `addProduct` позначає властивість, яка зв'язує клас `Manager` та клас `List_of_added_products`. Тобто, доменом цієї властивості є `Manager`, а діапазоном значень є `List_of_added_products`.
- `hasCategory` позначає властивість, яка зв'язує клас `Product` та клас `Category`. Тобто, доменом цієї властивості є `Product`, а діапазоном значень є `Category`.
- `hasProduct` позначає властивість, яка зв'язує клас `Check` та клас `Product`. Тобто, доменом цієї властивості є `Check`, а діапазоном значень є `Product`.
- `hasPurchase` позначає властивість, яка зв'язує клас `Customer` та клас `Check`. Тобто, доменом цієї властивості є `Customer`, а діапазоном значень є `Check`.
- `writeCheck` позначає властивість, яка зв'язує клас `Cashier` та клас `Check`. Тобто, доменом цієї властивості є `Cashier`, а діапазоном значень є `Check`.

Властивості даних (Data property), які були створені



Назва	Domains	Тип даних	Призначення
timeDate	Check, List_of_added_products	dateTime	Для запису часу створення чеку або листа доданих нових продуктів
categoryName	Category	string	Для назви категорії
descriptionOfProduct	Product	string	Для опису продукту. (Маса нетто та інші характеристики товару)
productName	Product	string	Назва продукту (Молоко, Рогалики і т. д. і т. п.)
price	Product	decimal	Ціна продукту
personalFullName	Person	string	ПІБ людини
manufacturerName	Manufacturer	string	Назва виробника

3 Робота Reasoner та деякі SWRL правила

Reasoner (машина логічного висновка) – це клас програмного забезпечення, яке призначене для роботи з онтологічними моделями. Він дозволяє вичислити значення логічних виразів, перевірити правильність онтології та автоматично додавати до неї нову інформацію у відповідності з правилами.

Для роботи був використаний reasoner «Pellet», який, на мою думку, є найбільш оптимальним рішенням в рамках даної онтології.

- Задавши характеристики властивості `hasSameCategory` та вказавши значення, які представлені в таблиці нижче:

Об'єкт	Предикат	Суб'єкт
Bread01	<code>hasSameCategory</code>	Bread02
Bread03	<code>hasSameCategory</code>	Bread01
Bread04	<code>hasSameCategory</code>	Bread01

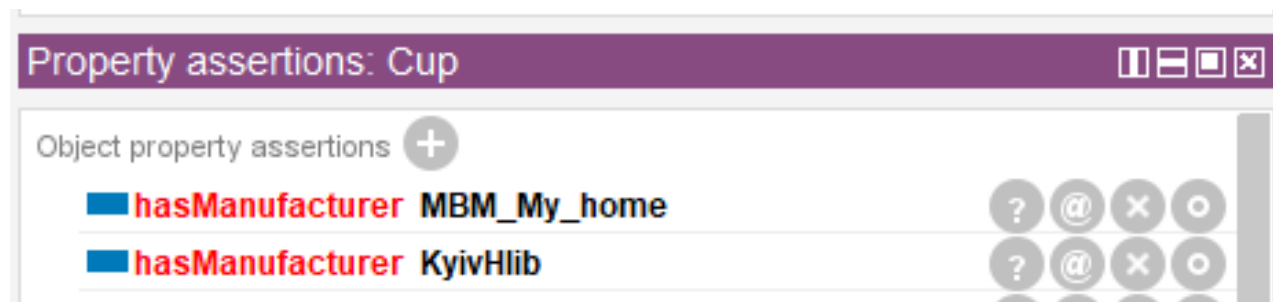
Було отримано наступний результат:

Property assertions: Bread02

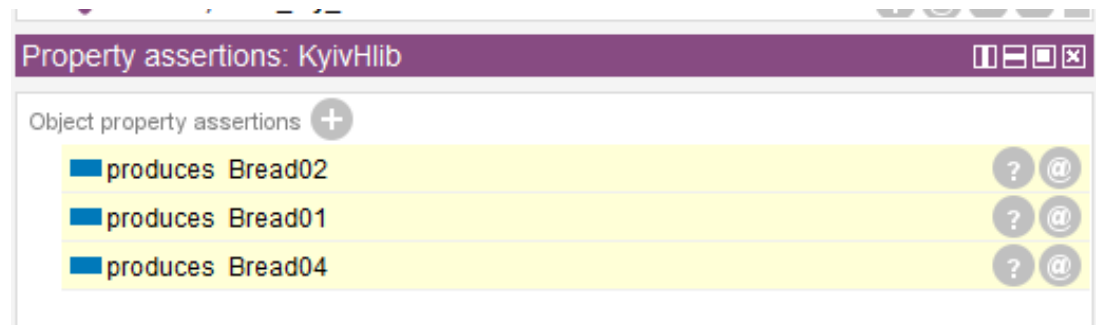
Object property assertions +

- hasCategory Bakery_products
- hasSameCategory Bread02
- hasSameCategory Bread01
- hasSameCategory Bread04
- hasSameCategory Bread03

- Властивість `hasManufacturer` є функціональною, це означає те, що у одного товару може бути лише один виробник. Отже, якщо задати певному продукту два виробника, то `reasoner` має видати помилку. Хоча, якщо не задано, що всі об'єкти класу `Виробник` є різними об'єктами, то `reasoner` припустить, що `КиївНліб` та `МВМ_Му_home` це один і той самий об'єкт.



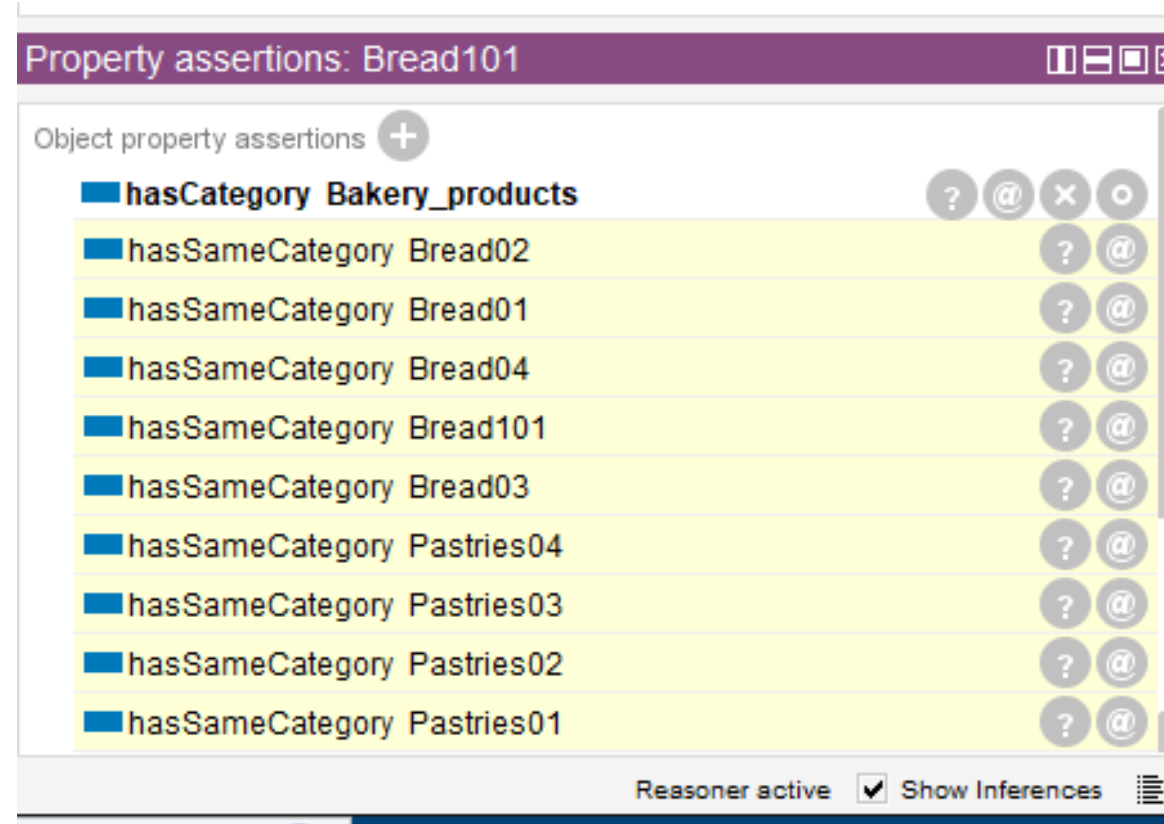
- Властивість `produces` (виробляє) є оберненою до властивості `hasManufacturer`. Отже, після запуску `reasoner` буде видавати висновки представлені



SWRL правила

SWRL — це мова правил семантичної павутини , яка може використовуватися для вираження правил, а також логіки, поєднуючи OWL DL або OWL Lite з підмножиною мови розмітки правил.

- 1) `hasCategory(?x, ?category)`,
`hasCategory(?y, ?category) ->`
`hasSameCategory(?x, ?y)`



2) $\text{hasSameCategory}(?x, ?y), \text{hasCategory}(?y, ?category) \rightarrow \text{hasCategory}(?x, ?category)$

Property assertions: Bread102

Object property assertions +

- hasSameCategory Bread01
- hasCategory Bakery_products**
- hasSameCategory Bread02
- hasSameCategory Bread102
- hasSameCategory Bread04
- hasSameCategory Bread101
- hasSameCategory Bread03
- hasSameCategory Pastries04
- hasSameCategory Pastries03
- hasSameCategory Pastries02

4. Запити SPARQL та отримання інформації з онтології «Grocery store»

Мова запитів SPARQL – це мова запитів до даних, представлених по моделі RDF, а також протокол для передачі цих запитів і відповідей на них. SPARQL є рекомендацією консорціуму W3C і одною з технологій семантичної павутини.

Для кращої читабельності SPARQL запитів використовують префікси (PREFIX). Вони потрібні аби зайвий раз не прописувати повний URI. В усіх запитах будуть використані наступні префікси:

PREFIX xsd: <<http://www.w3.org/2001/XMLSchema#>>

PREFIX owl: <<http://www.w3.org/2002/07/owl#>>

PREFIX rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>

PREFIX rdfs: <<http://www.w3.org/2000/01/rdf-schema#>>

PREFIX type: <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>


```

1) SELECT ?x ?y ?z WHERE {
    ?x ?y ?z
}
LIMIT 100

```

Результатом виконання першого запиту на онтології «Grocery store» стане виведення всіх триплетів даної онтології.

x	y	z	
1	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/2000/01/rdf-schema#dom...	<http://www.semanticweb.org/vavdi/ontologie...
2	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/2000/01/rdf-schema#dom...	<http://www.semanticweb.org/vavdi/ontologie...
3	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.w3.org/2002/07/owl#FunctionalPr...
4	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.w3.org/2002/07/owl#DatatypePro...
5	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/2000/01/rdf-schema#rang...	<http://www.w3.org/2001/XMLSchema#dateTi...
6	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	"39.69"^^<http://www.w3.org/2001/XMLSchema#decimal>
7	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...
8	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.semanticweb.org/vavdi/ontologie...
9	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.w3.org/2002/07/owl#NamedIndiv...
10	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Кефір
11	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	2.5% жиру, маса нетто: 900г
12	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Яготинський
13	b0	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	b1
14	b0	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.semanticweb.org/vavdi/ontologie...
15	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	"40.99"^^<http://www.w3.org/2001/XMLSchema#decimal>
16	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...
17	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.semanticweb.org/vavdi/ontologie...
18	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.w3.org/2002/07/owl#NamedIndiv...
19	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Молоко
20	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/2002/07/owl#differentFro...	<http://www.semanticweb.org/vavdi/ontologie...
21	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Ультрапастеризоване 2.6% жиру, маса нетто: ...
22	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Яготинське
23	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	"30.49"^^<http://www.w3.org/2001/XMLSchema#decimal>
24	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...
25	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.semanticweb.org/vavdi/ontologie...
26	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.w3.org/1999/02/22-rdf-syntax-ns...	<http://www.w3.org/2002/07/owl#NamedIndiv...
27	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Рогалики, маса нетто: 360г
28	<http://www.semanticweb.org/vavdi/ontologie...	<http://www.semanticweb.org/vavdi/ontologie...	Киви...

2) SELECT DISTINCT ?name ?manuf ?price ?descrip WHERE {

?x type: <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Product>.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#hasCategory>

<http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Dairy_products>.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#productName> ?name.

OPTIONAL {

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#manufacturerOfProduct> ?manuf.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#price> ?price.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#descriptionOfProduct>

?descrip

}

}

LIMIT 100

Результатом виконання другого запиту стане виведення всіх назв товарів категорії Dairy_products (молочні продукти). Також, якщо у об'єкта класу продукт, який належить до згаданої вище категорії, вказана ціна, виробник або опис, це теж буде виведено. Саме для ситуацій, коли результатом запиту може бути порожня множина, використовують OPTIONAL.

Кефір	Яготинський	39.69	2,5% жиру, маса нетто: 900г
Молоко	Яготинське	40.99	Ультрапастеризоване 2,6% жиру, маса нетто: 900г
Вершки	Яготинські	23.22	10% жиру, маса нетто: 200г
Кефір	Ферма	40.39	2,5% жиру, маса нетто: 840г
Масло	Галичина	76.59	Жирність 82,5%, маса нетто 180г
Вершки	Простоквашино	30.29	15% жиру, маса нетто: 200г
Сир	Ферма	49.59	Сметанковий, маса нетто 150г
Йогурт	Фанні	23.22	Десерт сирковий "Вершковий", маса нетто: 150г

```
3) SELECT ?name ?manuf ?price ?descrip WHERE {  
  <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Check000000>  
<http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#hasProduct> ?x.  
  ?x          <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-  
3#productName> ?name.  
  ?x          <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-  
3#manufacturerOfProduct> ?manuf.  
  ?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#price> ?price.  
  ?x          <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-  
3#descriptionOfProduct> ?descrip  
}  
LIMIT 100
```

Результатом виконання третього запиту стане знаходження всіх продуктів за унікальним ідентифікатором чеку, в даному прикладі за Check000000. Так само, як і в попередньому прикладі запиту тут будуть виводитися назва продукту, його опис, ціна та бренд. Результат виконання третього запиту наведений нижче.

Назва	Бренд	Ціна	Опис
Молоко	Яготинське	40.99	Ультрапастеризоване 2,6% жиру, маса нетто: 900г
Хліб	Київхліб	21.49	Звичайний нарізний, маса нетто 500г
Чашка	МВМ Му home	195	Червона з малюнком, 300 мл
Йогурт	Галичина	23.99	Карпатський, вишня-черешня, маса нетто: 260г

4) SELECT ?name ?manuf ?price ?descrip WHERE {

<http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#Customer1>

<http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#hasPurchase> ?y.

?y <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#hasProduct> ?x.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#productName> ?name.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#manufacturerOfProduct>

?manuf.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#price> ?price.

?x <http://www.semanticweb.org/vavdi/ontologies/2023/4/untitled-ontology-3#descriptionOfProduct>

?descrip

}

LIMIT 100

Результатом виконання четвертого запиту стане знаходження всіх продуктів які купляв певний покупець, в даному прикладі Customer1. Так само, як і в попередніх прикладах запитів, тут будуть виводитися назва продукту, його опис, ціна та бренд. Результат виконання третього запиту наданий нижче:

Назва	Бренд	Ціна	Опис
Молоко	Яготинське	40.99	Ультрапастеризоване 2,6% жиру, маса нетто: 900г
Хліб	Київхліб	21.49	Звичайний нарізний, маса нетто 500г
Чашка	МВМ My home	195	Червона з малюнком, 300 мл
Йогурт	Галичина	23.99	Карпатський, вишня-черешня, маса нетто: 260г
Булочки	Київхліб	14.19	"Малютко", маса нетто: 250г
Масло	Яготинське	75.25	Жирність 82,5%, маса нетто 180г

Розділ 3. Реалізація демонстраційної програми на Python

3.1 Отримання інформації з RDF файлу

Для отримання даних з онтології було використано бібліотеку rdflib, яка надає можливості для задання SPARQL запитів до онтології. На зображено підключення бібліотеки rdflib та прасинг RDF файлу.

```
from rdflib import Graph

g = Graph()
g.parse("D:\course_work\ontology_model.rdf")
```


3.2 SPARQL запити в програмі

Весь SPARQL запит задається у зміню. Для зручності всі PREFIXES було винесено в окрему зміню, яка виглядає наступним чином:

```
prefixes = """  
PREFIX owl: <http://www.w3.org/2002/07/owl#>  
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-  
ns#>  
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
PREFIX type: <http://www.w3.org/1999/02/22-rdf-syntax-  
ns#type>  
"""
```

Задання запиту до онтології відбувається використовуючи метод `g.query()`, якому передається запит у вигляді рядка. Це виглядає в коді наступним чином:

```
g.query(all_products())
```

Для прикладу було взято запит, який виводить інформацію про всі продукти в магазині.

Результат запиту, який показує інформацію про всі продукти в магазині

New Window

Назва	Виробник	Опис	Ціна
Хліб	Київхліб	Звичайний нарізний, маса нетто 500г	21.49
Хліб	Київхліб	"Супертост", маса нетто 500г	26.59
Хліб	Кулиниці	Звичайний нарізний, маса нетто 250г	14.99
Хліб	Київхліб	"Оксамитовий", маса нетто: 350г	26.99
Масло	Яготинське	Жирність 82,5%, маса нетто 180г	75.25
Масло	Ферма	Жирність 82,5%, маса нетто 180г	92.99
Масло	Галичина	Жирність 82,5%, маса нетто 180г	76.59
Сир	Ферма	Сметанковий, маса нетто 150г	49.59
Сир	Комо	Старий Голагдець, маса нетто 100г	48.40
Сир	Комо	Традиційний, маса нетто 150г	57.99
Сир	Комо	Парменталь, маса нетто: 160г	59.99
Вершки	Яготинське	10% жиру, маса нетто: 200г	23.22
Чашка	MBM_My_home	Червона з малюнком, 300 мл	195
Кефір	Яготинське	2,5% жиру, маса нетто: 900г	39.69
Кефір	Ферма	2,5% жиру, маса нетто: 840г	40.39
Молоко	Яготинське	Ультрапастеризоване 2,6% жиру, м	40.99
Молоко	Селянське	Ультрапастеризоване 3,2% жиру, м	33.59
Молоко	Селянське	Пастеризоване 2,6% жиру, маса нетто: 200г	61.99
Булочки	Київхліб	"Малятко", маса нетто: 250г	14.19
Рогалики	Кулиниці	Маса нетто: 360г	30.49
Кекс	Кулиниці	"Рум'янець" вишневий, маса нетто: 300г	15.99
Пиріг	Кулиниці	"Рум'янець" з маком, маса нетто: 300г	59.99
Сметана	President	15% жиру, маса нетто: 325г	41.59

Результат запиту, який показує інформацію про всі продукти певної категорії (в даному випадку Молочні продукти)

Назва	Виробник	Опис	Ціна
Масло	Яготинське	Жирність 82,5%, маса нетто 180г	75.25
Масло	Ферма	Жирність 82,5%, маса нетто 180г	92.99
Масло	Галичина	Жирність 82,5%, маса нетто 180г	76.59
Сир	Ферма	Сметанковий, маса нетто 150г	49.59
Сир	Комо	Старий Голагдець, маса нетто 100г	48.40
Сир	Комо	Традиційний, маса нетто 150г	57.99
Сир	Комо	Парменталь, маса нетто: 160г	59.99
Вершки	Яготинське	10% жиру, маса нетто: 200г	23.22
Кефір	Яготинське	2,5% жиру, маса нетто: 900г	39.69
Кефір	Ферма	2,5% жиру, маса нетто: 840г	40.39
Молоко	Яготинське	Ультрапастеризоване 2,6% жиру, м	40.99
Молоко	Селянське	Ультрапастеризоване 3,2% жиру, м	33.59
Молоко	Селянське	Пастиризоване 2,6% жиру, маса не	61.99
Сметана	President	15% жиру, маса нетто: 325г	41.59
Сметана	Яготинське	20% жиру, маса нетто 400г	42.63
Сметана	Молокія	20% жиру, маса нетто: 300г	35.99
Йогурт	Галичина	Карпатський, вишня-черешня, ма	23.99
Йогурт	Фанні	Десерт сирковий "Вершковий", ма	23.22
Вершки	Простоквашено	15% жиру, маса нетто: 200г	30.29

Результат запиту, який показує інформацію про всі продукти, які були куплені деяким покупцем (в прикладі наведено id:Customer1)

New Window

Назва	Виробник	Опис	Ціна
Хліб	Київхліб	Звичайний нарізний, маса нетто 50	21.49
Чашка	MBM_My_home	Червона з малюнком, 300 мл	195
Молоко	Яготинське	Ультрапастеризоване 2,6% жиру, м	40.99
Йогурт	Галичина	Карпатський, вишня-черешня, макс	23.99

Висновки

Отже, заплановані цілі були успішно виконані. У ході роботи було проаналізовано існуючі підходи та методи розробки онтологій. Також було проведено аналіз існуючих технологій, середовищ розробки, фреймворків та плагінів. Програма Protégé, на моє переконання, найкраще підходить для створення онтологій.

У результаті роботи було успішно створено онтологію "Grocery store" та реалізовано її в системі Protégé. Було протестовано створену онтологію за допомогою reasoner, а також за його допомогою було отримано нові знання. Було розглянуто мову запитів SPARQL та виконано декілька запитів до побудованої онтології. Також, було розглянуто реалізацію демонстраційної програми на мові Python. Досягнення поставлених цілей підтверджується результатами моєї роботи, а саме: успішна реалізація бази знань та можливість ефективного пошуку та аналізу інформації про продукти в магазині.

Список використаної літератури

- Gruber T.R. The role of common ontology in achieving sharable, reusable knowledge bases // Principles of Knowledge Representation and Reasoning. Proceedings of the Second International Conference. J.A. Allen, R. Fikes, E. Sandewell – eds. Morgan Kaufmann, 1991, 601-602.
- Gruber T.R. A translation approach to portable ontologies. Knowledge Acquisition, 5(2): 199-220, 1993.
- The KAON. – Режим доступу: <http://kaon.semanticweb.org/>
- The OilEd. – Режим доступу: <http://oiled.man/>
- The OntoEdit. – Режим доступу: <http://www.ontoprise.de/com/ontoedit.htm>
- The Ontosaurus. – Режим доступу: <http://www.isi.edu/isd/ontosaurus.html>
- The OpenСус. – Режим доступу: <http://www.opencyc.org/>
- The Protege Project. – Режим доступу: <http://protege.stanford.edu>.
- Briukhov D.O., Shumilov S.S. Ontology Specification and Integration Facilities in a Semantic Interoperation Framework. Proc. of the Second Intern. Workshop ADBIS'95. Moscow, 1995. - P. 195-200.
- Natalya F. Noy and Deborah L. McGuinness. «Ontology Development 101: A Guide to Creating Your First Ontology». Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001

Дякую за увагу!