**CM 4330**

**Semantic Web and Ontological Modeling**

**Assignment 02**

**Application Ontology**

|  |  |
| --- | --- |
| **Name** | **Index** |
| D.A.P Madhubhashini | 164081J |
| K.G.C Sandeepani | 164120F |

**Faculty of Information Technology**

**University of Moratuwa**

**2020**

**Content**

[**Introduction**](#_ryxproruf7mc) **3**

[1.1 What is an Ontology](#_71v0ata0q397) 3

[1.2 Why the Movie needs an ontology](#_9s4noa5hgtu1) 4

[1.3 How the film industry can use a ontology](#_9s4noa5hgtu1) 4

[**The Ontology for Movie**](#_vva97jp3nk9u) **5**

[2.1 Class Ontology](#_9s4noa5hgtu1) 5

[**Top Level classes**](#_okole3ipbvqn) **8**

[Primitive Classes](#_dldtveaig76o) 8

[Named Classes (Define Classes)](#_v562jqa9qlku) 9

[2.2 Object Property Ontology](#_chxfm8cgjy3u) 9

[2.2.1 Transitive Property](#_hpo5lck4du0x) 12

[2.2.2 Functional Property](#_dys3ecj73uw2) 12

[2.2.3 Inverse Functional Property](#_2kl9vmu5iu27) 13

[2.2.4 Symmetric Property](#_cimjanjxd6n6) 13

[2.3 Data Property Ontology](#_f2v4txevuz7v) 13

[2.4 Instances](#_lv8dmpl0te2c) 14

[**Result**](#_vjjhmwwqtiop) **17**

[**Conclusion**](#_abqia2heg1co) **20**

**Figure 2.1 - class hierarchy**

**Figure 2.2 - class hierarchy**

**Figure 2.3 - Object property**

**Figure 2.4 - Data type properties**

**Figure 2.5 - Instances of Movie Ontology**

**Figure 3.1 - Ontology visualized on Ontograf**

**Figure 3.2 - KidsCartoonMovie class view with Reasoner**

**Figure 3.3 - OldMovie class view with Reasoner**

**Figure 3.4 - SPARQL query to get all individuals**

**Figure 3.5 - SPARQL query to get only crew instances**

**Figure 3.6 - SPARQL query to get get movie instances and their created year**

# Introduction

The film industry, or the motion picture industry, includes film making technology and commercial institutions such as film production companies, film studios, cinematography, animation, film production, screenwriting, pre-production, post-production, film festivals, distribution and actors, film directors and other film crew staff. Although the cost of producing films almost immediately led to the consolidation of film production under the auspices of standing production companies, developments in inexpensive film making equipment, and expansion of opportunities to acquire investment capital from outside the film industry itself, enabled the growth of independent film production.

If data is a key to the future, the film and television industry is already on its way to unlock the future for the better. Every organization like film production companies is busy integrating existing data systems and innovative new systems designed to give decision-makers more and better insights.These initiatives are constantly improving the data capabilities at the enterprise level across the industry. However, as organizations are increasingly adding data sources to power machine learning and other potential to create even deeper perspectives, the industry needs to do more, not only better systems for every business but better systems and tools for the entire industry.

An ontology offers a machine-to - machine structure to simplify the ties between the various data structures that occupy the media world.

Here, we describe a creative work ontology for the movie industry.

## 1.1 What is an Ontology

An ontology describes the objects, concepts and relationships that characterize a domain in wisdom. It offers specific machine-to - machine concepts that can be used in the organization

of and connect multiple sources of data within a given domain. The ontology discussed here focuses on the objects, concepts and relationships typically used to define creative works in the films, and is intended as a general resource for metadata creation, analytics, distribution, archiving, academic research and other industry use cases.

A successful ontology asks,"What do we care about the data and connections? "The ontology then takes advantage of the answers to minimize ambiguity and add clarification. It focuses on the things that matter to us ("entities")

And explains it with:

* Types : the kinds of entities (like movies, people, and locations),
* Properties: information that describes and distinguishes the entities (like titles for movies, names for people, and countries for locations),
* Relationships: how entities connect (e.g., a person directed a movie, a movie is set in a location).

The industry has traditionally spent more time discussing individual works, and less time discussing the relationships between works and other individuals. However, relations between works are becoming increasingly significant, as the industry is increasingly relying on the analysis of broad data sets spanning portfolios, promotion of film families in the form of franchises and trademarks, and material manipulation across multiple channels and platforms through movies.

An ontology focuses on the relationships between entities, not simply the properties of entities, comparatively more attention. This emphasis helps to build structures and databases (such as triple stores1) that integrate structured relationships as a core component of architectural design. In numerous technical definitions and hooks an ontology also builds.

## 1.2 Why the Movie needs an ontology

The movie/film industry becomes more data-driven. That trend is set to continue as the industry pulls in data from even more diverse distribution platforms and sources. The increasing focus on data highlights the need to connect the disparate data sources and disconnected data silos that populate the entertainment landscape.

As the business grows, studios will acquire more consumption data from more sources, requiring new and better tools to analyze that data for purposes of competitive analysis, affinity analysis, and marketing decision making across franchises, universes, and channels. Studios and distributors will continue to deploy more powerful machine learning and AI tools that thrive on massive aggregations of data from multiple sources. The challenges an ontology addresses will only increase. Adoption and deployment of a common, machine-to-machine ontology will result in faster, more efficient integration of data, faster, more efficient development of applications to keep the industry competitive, and faster, more meaningful insights from data that includes connections surfaced by the ontology.

## 1.3 How the film industry can use a ontology

An ontology addresses data structuring questions with solutions already vetted and tested by others in the industry, allowing data technologists to focus on new extensions and innovations, not problems that have already been solved.

As an example, the ontology has a pre-defined model for describing both the relationships between movies and other things like books, cartoons.

The ontology of the film was inspired by the fact that most ontologies of the film centered primarily on concepts and semant connections between those concepts. Furthermore, current film ontologies superficially describe the individuals or instances of a concept. We assume a semantic ontology of films should have hierarchies of concepts (e.g. for movie categorization and navigation purposes) on the one hand and a sufficient set of individuals that can be used to describe movies. This allows user-friendly film descriptions to be described in the required detail and the opportunity to describe films taking advantage of the rich variety of controlled vocabulary in film ontology. We are therefore trying to provide in-depth concepts and individuals, amongst other things , particularly for genre types

# The Ontology for Movie

## 2.1 Class Ontology

Thing

Movie

OldMovie

Below18Movie

ComedyMovie

InterestingMovie

TraditionalKidsMovie

NamedMovie

ActionComedyMovie

KidsCartoonMovie

RomComMovie

SuperHumanMovie

WarMovie

MovieGenre

ActionGenre

AdventureGenre

DisasterGenre

MilitaryActionGenre

SuperHeroGenre

AnimationGenre

ModernAnimationGenre

TraditionalAnimationGenre

ComedyGenre

CrimeGenre

DramaGenre

ExperimentalGenre

FantasyGenre

HistoricalGenre

HorrorGenre

GhostGenre

MonsterGenre

VampireGenre

WerewolfGenre

RomanceGenre

ScienceFictionGenre

ThrillerGenre

WesternGenre

Crew

Actor

ArtDirector

CostumeDesigner

Director

DirectorOfPhotography

Editor

MusicSupervisor

Producer

ProductionDesigner

ScreenWriter

Country

Afric

EasternAfric

MiddleAfric

NorthernAfric

SouthernAfric

WesternAfric

America

Caribbean

LatinAmerica

CentralAmerica

SouthAmerica

NorthernAmerica

Asia

EasternAsia

CentralAsia

SoutheasternAsia

SouthernAsia

WesternAsia

Europe

EasternEurope

NorthernEurope

SouthernEurope

WesternEurope

Oceania

AustraliaAndNewZealand

Malanesia

Micronesia

Polynesia

PresentationInfo

ColorInfo

LanguageInfo

SoundInfo

ValuePartition

AgeValuePartition

Above18

Below18

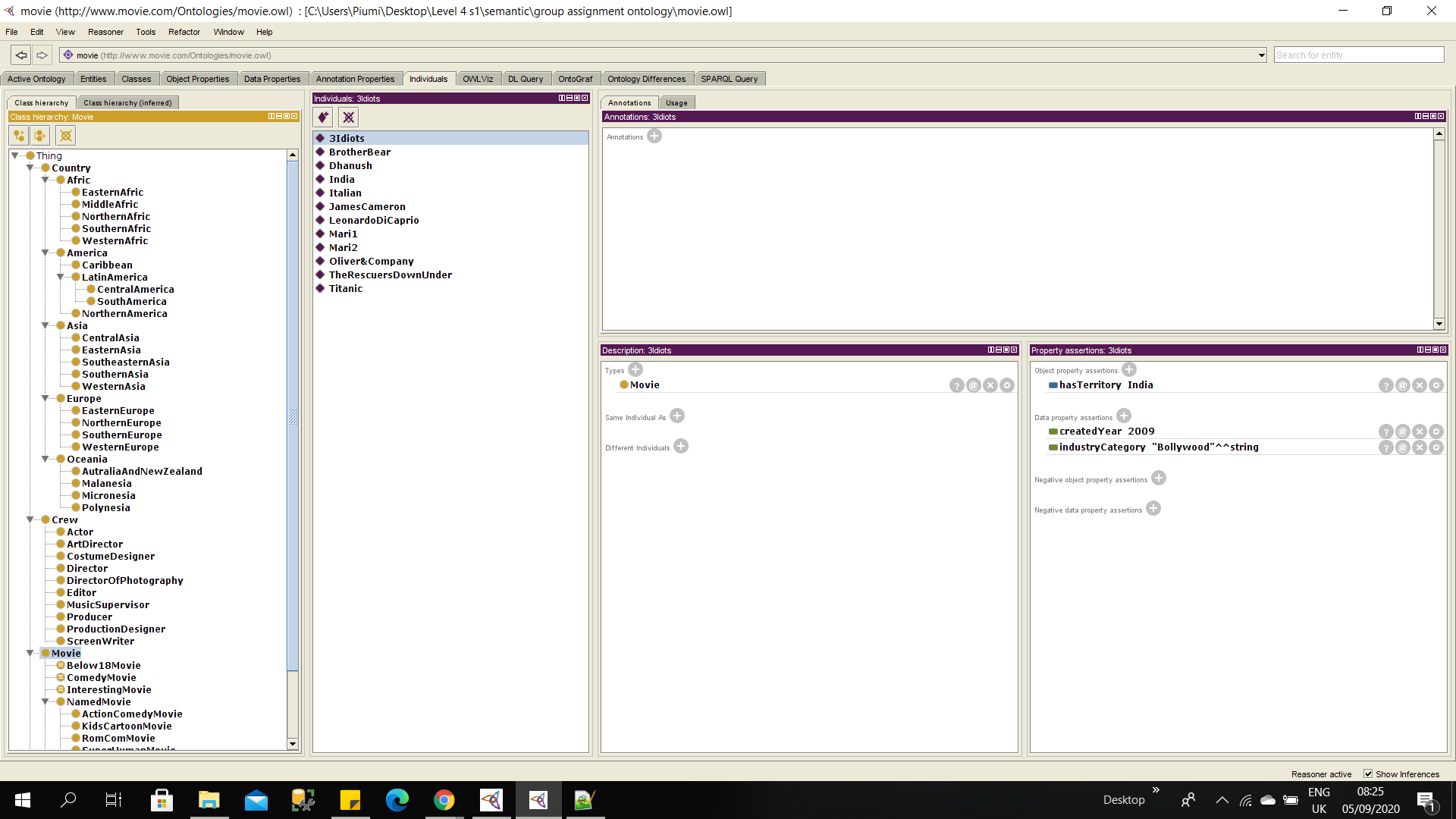


Figure 2.1 - class hierarchy

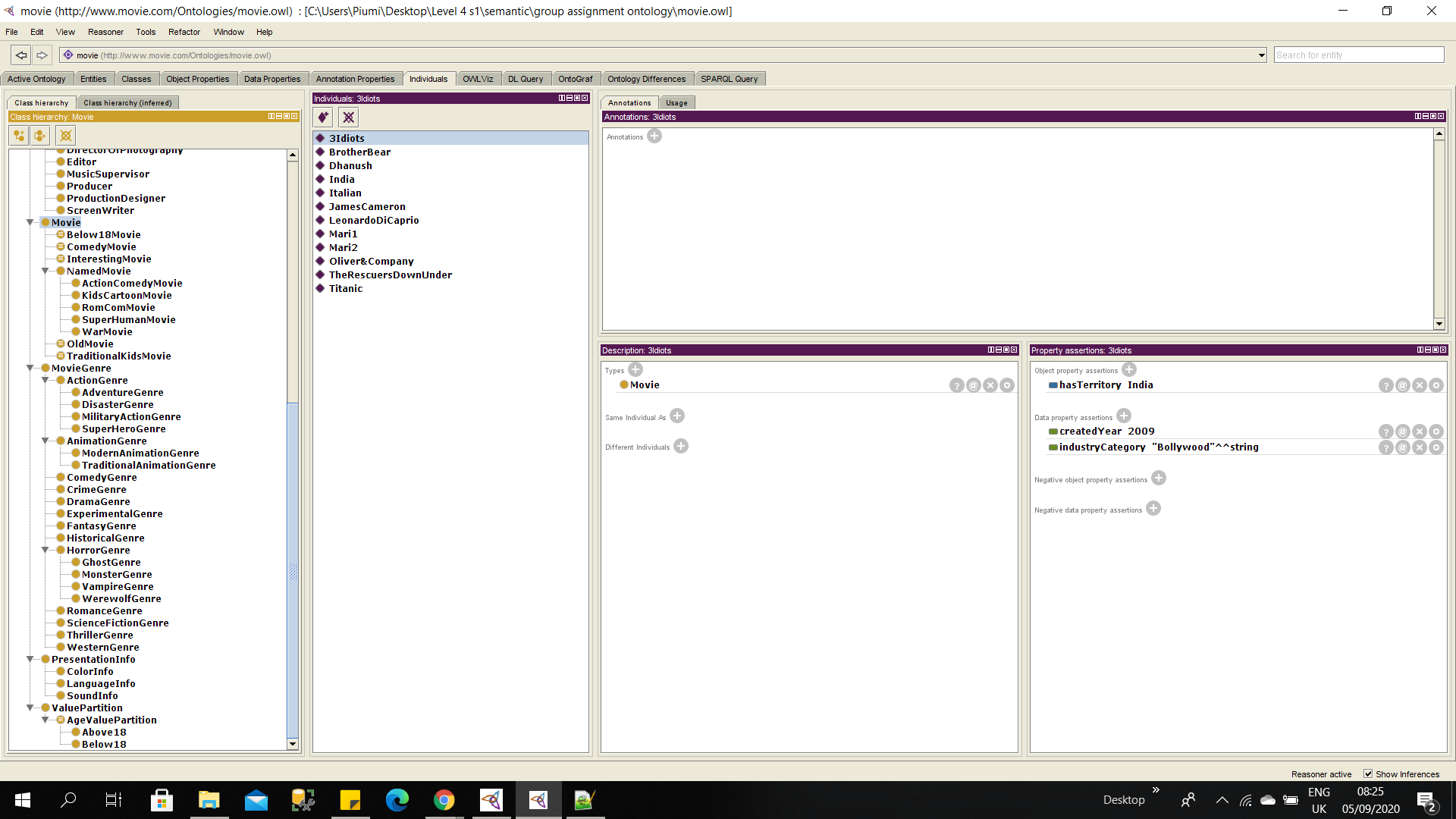


Figure 2.2 - class hierarchy

### Top Level classes

In our scenario, we developed primitive classes as well as named classes (define classes) to describe the film industry. Primitive classes include necessary conditions and named classes or define classes include necessary as well as sufficient conditions. These classes are mentioned below.

#### Primitive Classes

* ActionComedyMovie - A movie that can be classified as an action and comedy genre.
* KidsCartoonMovie - A movie that can only be classified as a fantasy and traditional animation genre because it consists of universal quantifiers.
* RomComMovie - A movie that can be classified as a romance and comedy genre.
* SuperHumanMovie - A movie that can be classified as a crime, fantasy and superHero genre.
* WarMovie - A movie that can be classified as adventure, crime, disaster and military action genre

#### Named Classes (Define Classes)

* OldMovie - A movie that was created before 2000.
* Below18Movie - A movie that is suitable for children under 18. For example, the traditional animation genre defined as it suitable for children under 18 and romance and horror genre defined as it suitable for people above 18.
* ComedyMovie - A movie that can be classified as a comedy genre.
* InterestingMovie - A movie that can be classified as it consists of at least 3 movie genres.
* TraditionalKidsMovie - A movie that can only be classified as a fantasy and animation genre.

## 2.2 Object Property Ontology

This object property hierarchy view displays the asserted and inferred object property hierarchies between classes. An object property connects two classes. Here Movie ontology, following object properties have been used.

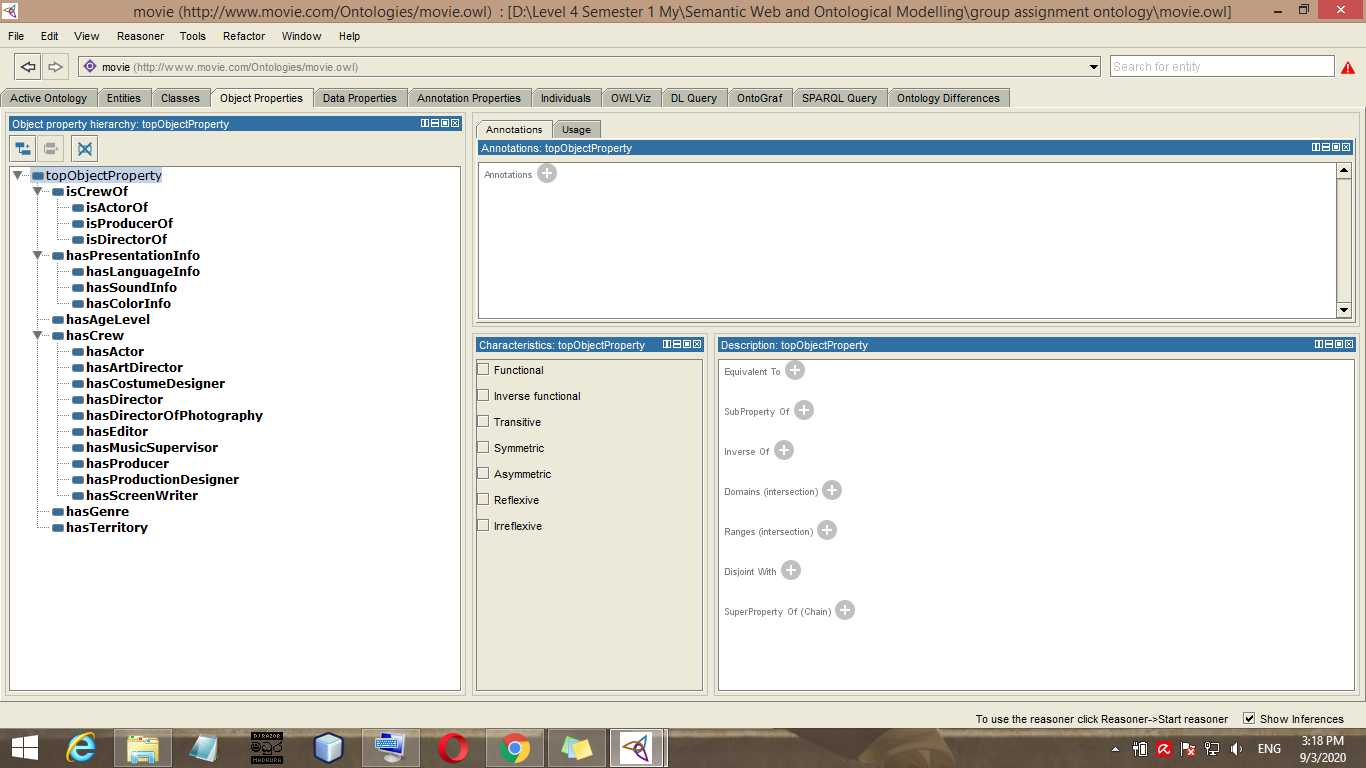


Figure 2.3 - Object property

Above object properties can describe as follows.



Each and every movie has a crew. It is denoted by ‘hascrew’ property. Inverse property of ‘hasCrew’ is ‘isCrewOf’.

### http://www.movie.com/Ontologies/movie.owl#hasCrew

:hasCrew rdf:type owl:ObjectProperty ;

rdfs:range :Crew ;

rdfs:domain :Movie ;

owl:inverseOf :isCrewOf .



Movie has actors. It is denoted by ‘hasActor’ property and the inverse property of ‘hasActor’ property is ‘isActorOf’.

### http://www.movie.com/Ontologies/movie.owl#hasActor

:hasActor rdf:type owl:ObjectProperty ;

rdfs:range :Actor ;

rdfs:domain :Movie ;

rdfs:subPropertyOf :hasCrew .

Likewise, The movie ontology has following object properties.

:connectedWith rdf:type owl:ObjectProperty ,

owl:SymmetricProperty ;

:hasAgeLevel rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

:hasArtDirector rdf:type owl:ObjectProperty ;

:hasColorInfo rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

:hasCostumeDesigner rdf:type owl:ObjectProperty ;

:hasDirector rdf:type owl:ObjectProperty ;

:hasDirectorOfPhotography rdf:type owl:ObjectProperty ;

:hasEditor rdf:type owl:ObjectProperty ;

:hasGenre rdf:type owl:ObjectProperty ;

:hasLanguageInfo rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

:hasMusicSupervisor rdf:type owl:ObjectProperty ;

:hasPresentationInfo rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

:hasProducer rdf:type owl:ObjectProperty ;

:hasProductionDesigner rdf:type owl:ObjectProperty ;

:hasScreenWriter rdf:type owl:ObjectProperty ;

:hasSoundInfo rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

:hasTerritory rdf:type owl:ObjectProperty ,

owl:TransitiveProperty ;

:isActorOf rdf:type owl:ObjectProperty ;

:isCrewOf rdf:type owl:ObjectProperty ;

:isDirectorOf rdf:type owl:ObjectProperty ;

:isProducerOf rdf:type owl:ObjectProperty ;

### 2.2.1 Transitive Property

Intuitively, this means that if individual x is related to individual y , and individual y is related to individual z , then individual x will be related to individual z.

In Movie ontology, we use ‘hasTerritory’ as Transitive property.

### http://www.movie.com/Ontologies/movie.owl#hasTerritory

:hasTerritory rdf:type owl:ObjectProperty ,

owl:TransitiveProperty ;

rdfs:range :Country ;

rdfs:domain :Movie .

### 2.2.2 Functional Property

This means that for any given individual, the property can have at most one value.

In Movie ontology, we use ‘hasAgeLevel’ as functional property.

### http://www.movie.com/Ontologies/movie.owl#hasAgeLevel

:hasAgeLevel rdf:type owl:FunctionalProperty ,

owl:ObjectProperty ;

rdfs:range :AgeValuePartition ;

rdfs:domain :MovieGenre .

### http://www.movie.com/Ontologies/movie.owl#createdYear

:createdYear rdf:type owl:DatatypeProperty ,

owl:FunctionalProperty ;

rdfs:subPropertyOf :MovieInfo .

### 2.2.3 Inverse Functional Property

This means the inverse property of the selected property (whether it explicitly declared or not) is Functional.

### 2.2.4 Symmetric Property

This means that the property has itself as an inverse, so if individual x is related to individual y then individual y must also be related to individual x along the same property.

Here Movie ontology, there is a Symmetric property called ‘connetedWith’. It is used when two movies are interconnected.

Ex:- Mari 2 movie connected with Mari 1.

### http://www.movie.com/Ontologies/movie.owl#connectedWith

:connectedWith rdf:type owl:ObjectProperty ,

owl:SymmetricProperty ;

rdfs:range :Movie ;

rdfs:domain :Movie .

## 2.3 Data Property Ontology

Datatype properties relate individuals to literal data (e.g., strings, numbers, datetimes, etc.) whereas object properties relate individuals to other individuals.

The Movie ontology has following data type properties.

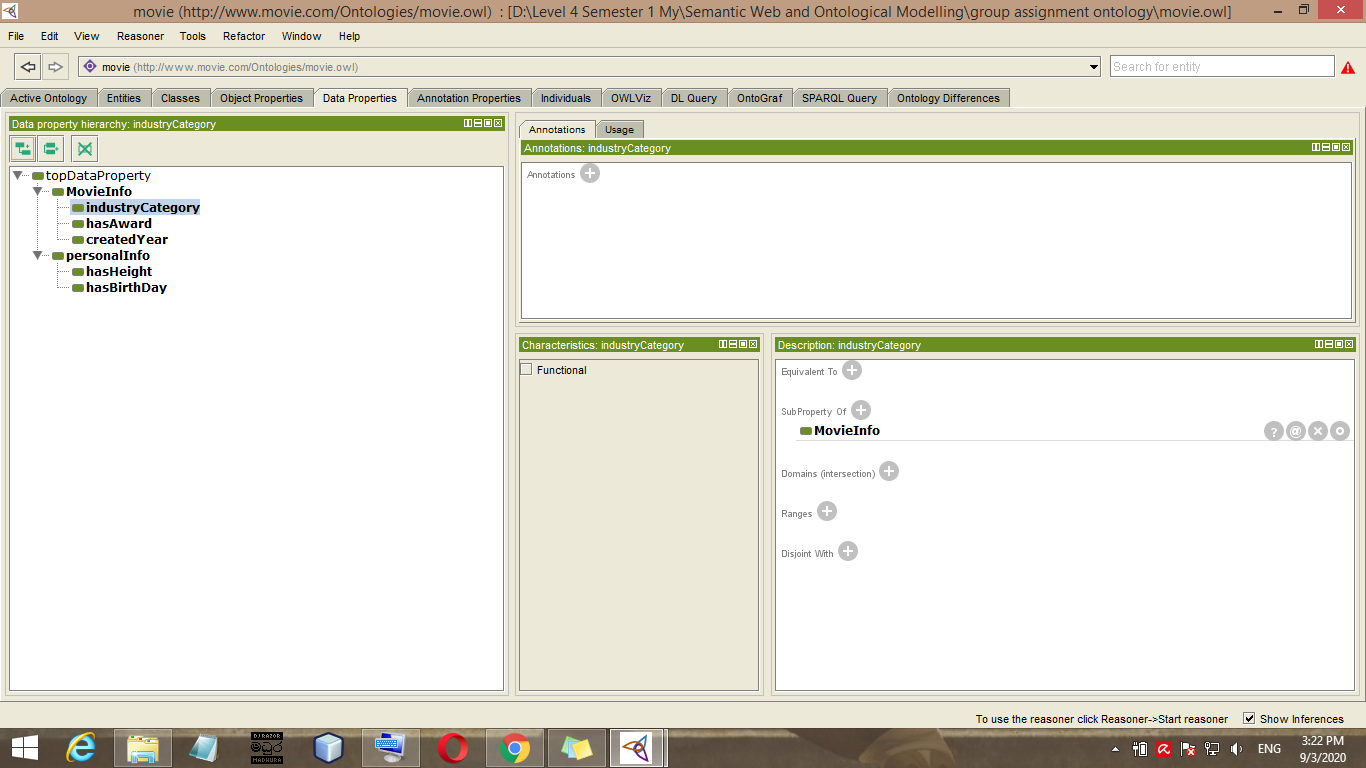


Figure 2.4 - Data type properties

:MovieInfo rdf:type owl:DatatypeProperty - String type property which describes the information of movies.

:createdYear rdf:type owl:DatatypeProperty ; - created year of the movie

:hasAward rdf:type owl:DatatypeProperty ; - Awards achieved by the movie.

:hasBirthDay rdf:type owl:DatatypeProperty ; - Birthday of a person belongs to the film crew.

:hasHeight rdf:type owl:DatatypeProperty ; - Height of a person belongs to the film crew.

:industryCategory rdf:type owl:DatatypeProperty ;

:personalInfo rdf:type owl:DatatypeProperty .

## 2.4 Instances

The instances/individuals in an ontology may include concrete objects such as people, animals, tables, automobiles, molecules, and planets, as well as abstract individuals such as numbers and words

Instances of the Movie ontology as follows.

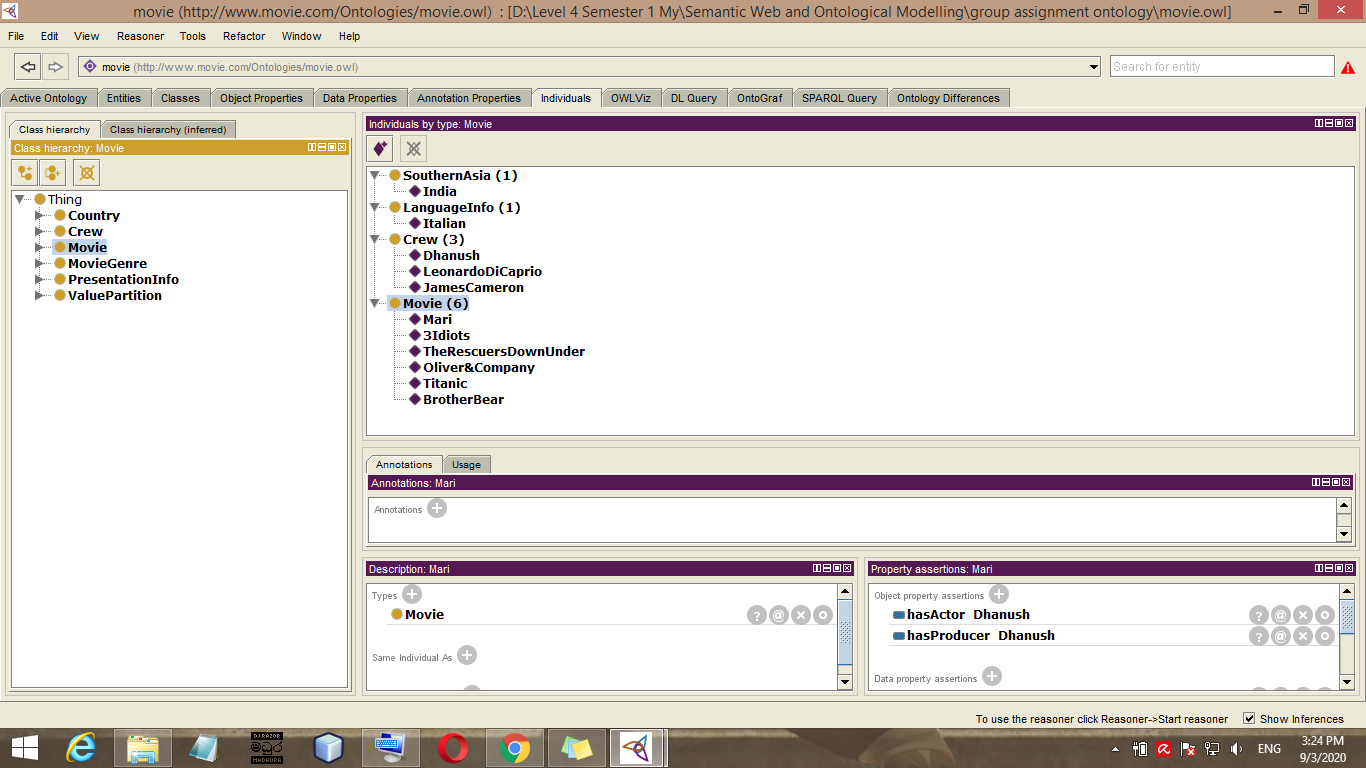


Figure 2.5 - Instances of Movie Ontology



The Titanic instance associated with “1997” literal by using createdYear data property.



The Titanic instance associated with “Hollywood” literal by using industryCategory data property.



The Titanic instance associated with “GoldenGlobeAward” literal by using hasAward data property.

Likewise Crew has data property ‘hasHeight’ and ‘hasBirthDay’.

We build several types of instances such as ‘Titanic’, ‘Mari’ for Movie class and ‘JamesCameron’ for Crew class.

Instance for type: Movie

:3Idiots rdf:type :Movie ,

owl:NamedIndividual ;

:BrotherBear rdf:type :Movie ,

owl:NamedIndividual ;

:Mari1 rdf:type :Movie ,

owl:NamedIndividual .

:Mari2 rdf:type :Movie ,

owl:NamedIndividual ;

:Oliver&Company rdf:type :Movie ,

owl:NamedIndividual ;

:TheRescuersDownUnder rdf:type :Movie ,

owl:NamedIndividual ;

:Titanic rdf:type :Movie ,

owl:NamedIndividual ;

Instance for type:Crew

:JamesCameron rdf:type :Crew ,

owl:NamedIndividual ;

:LeonardoDiCaprio rdf:type :Crew ,

owl:NamedIndividual ;

:Dhanush rdf:type :Crew ,

owl:NamedIndividual ;

Instance for type:Country

:India rdf:type :SouthernAsia ,

owl:NamedIndividual .

Instance for type:PresentationInfo

:Italian rdf:type :LanguageInfo ,

owl:NamedIndividual .

# Result

**View on OntoGraf**

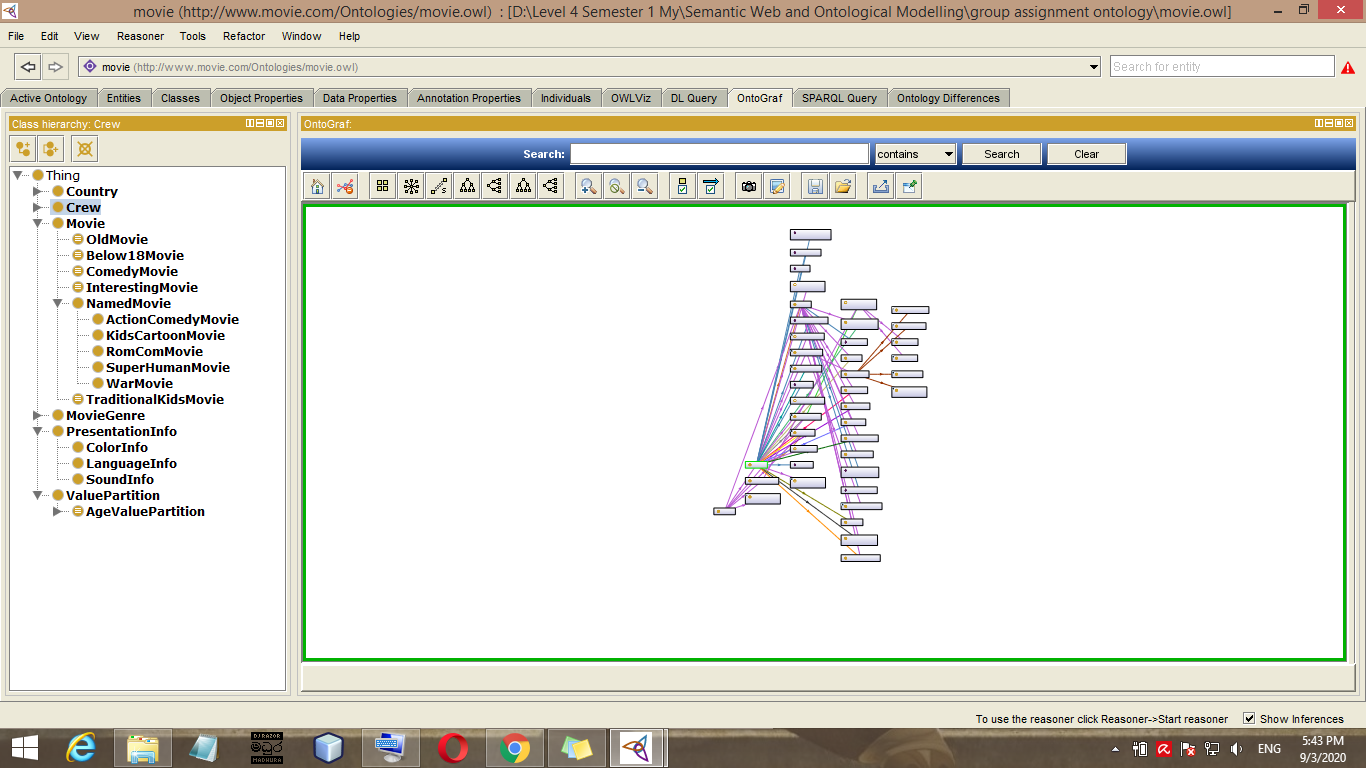


Figure 3.1 - Ontology visualized on Ontograf

**Some result of reasonor**

* KidCartoonMovie

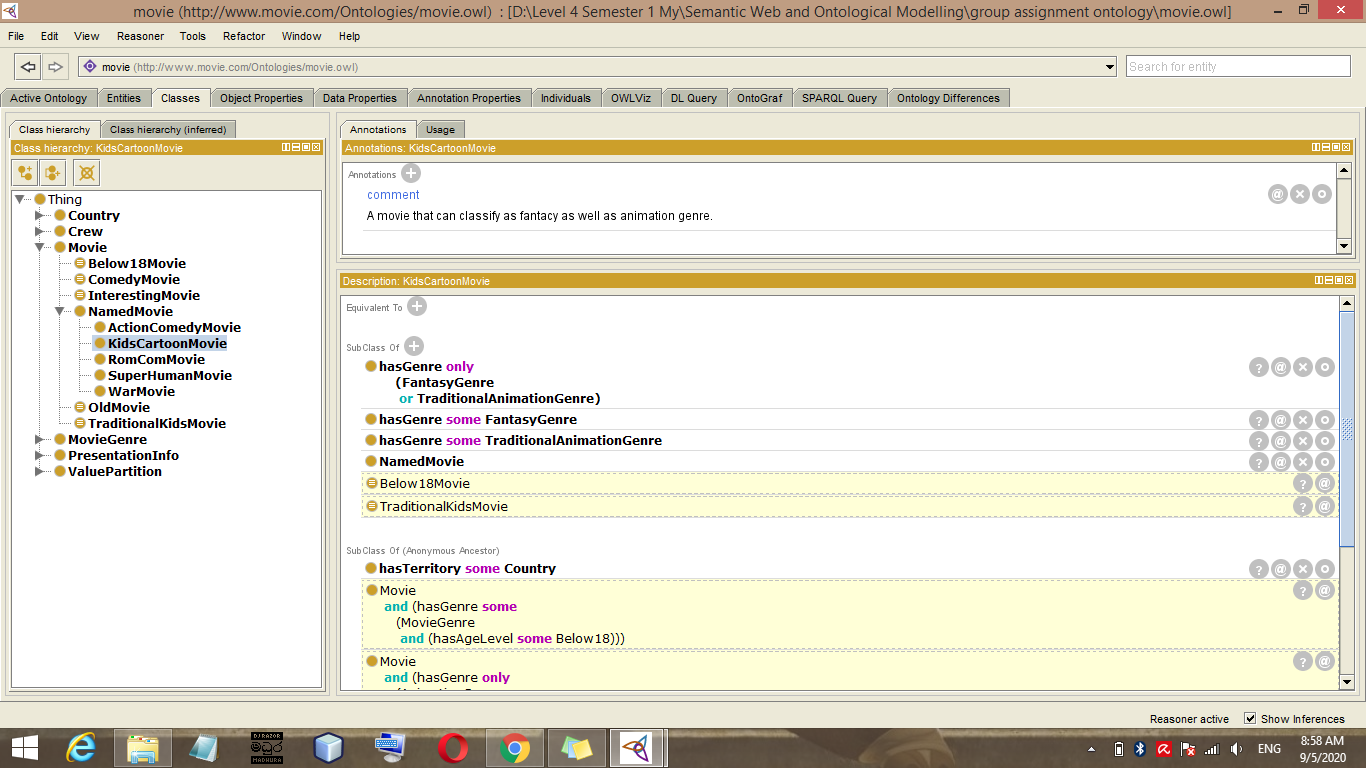


Figure 3.2 - KidsCartoonMovie class view with Reasoner

* OldMovie

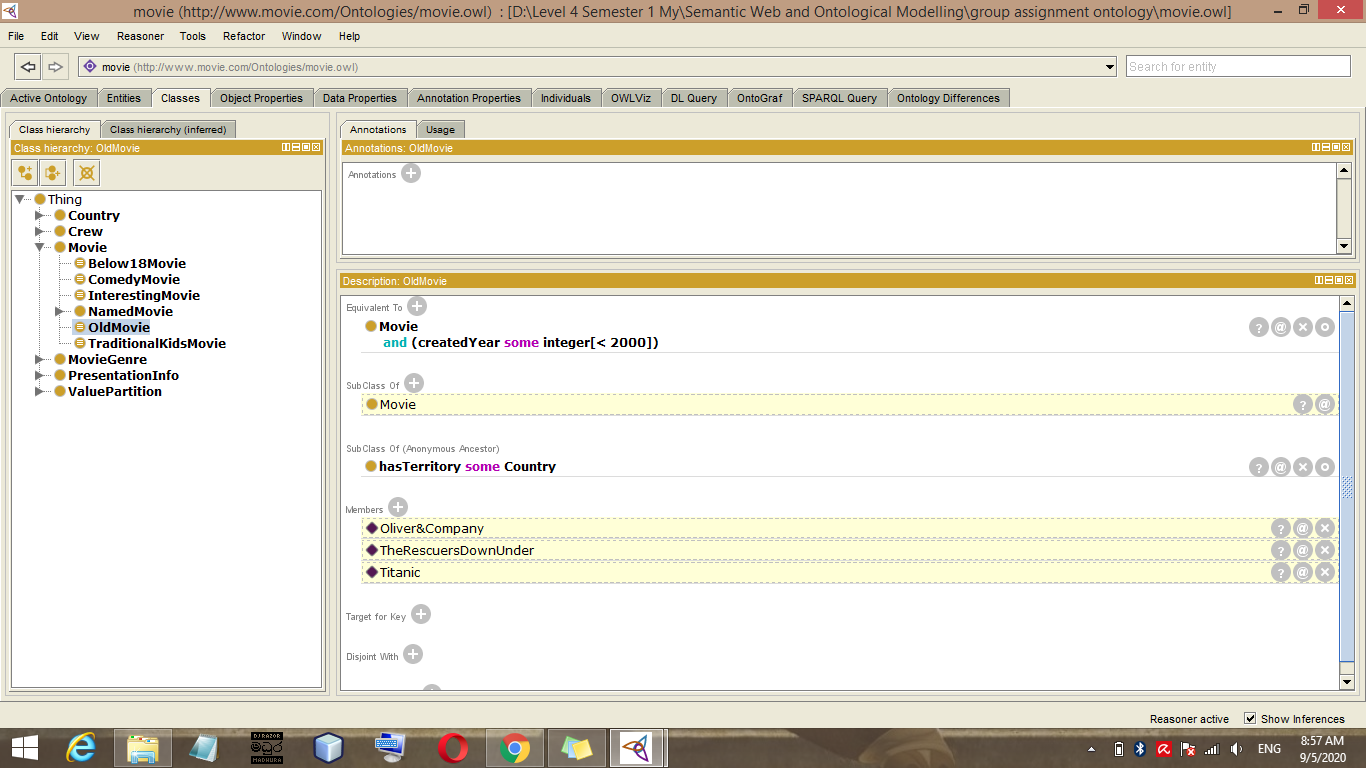


Figure 3.3 - OldMovie class view with Reasoner

**SPARQL Query**

* To get all individuals

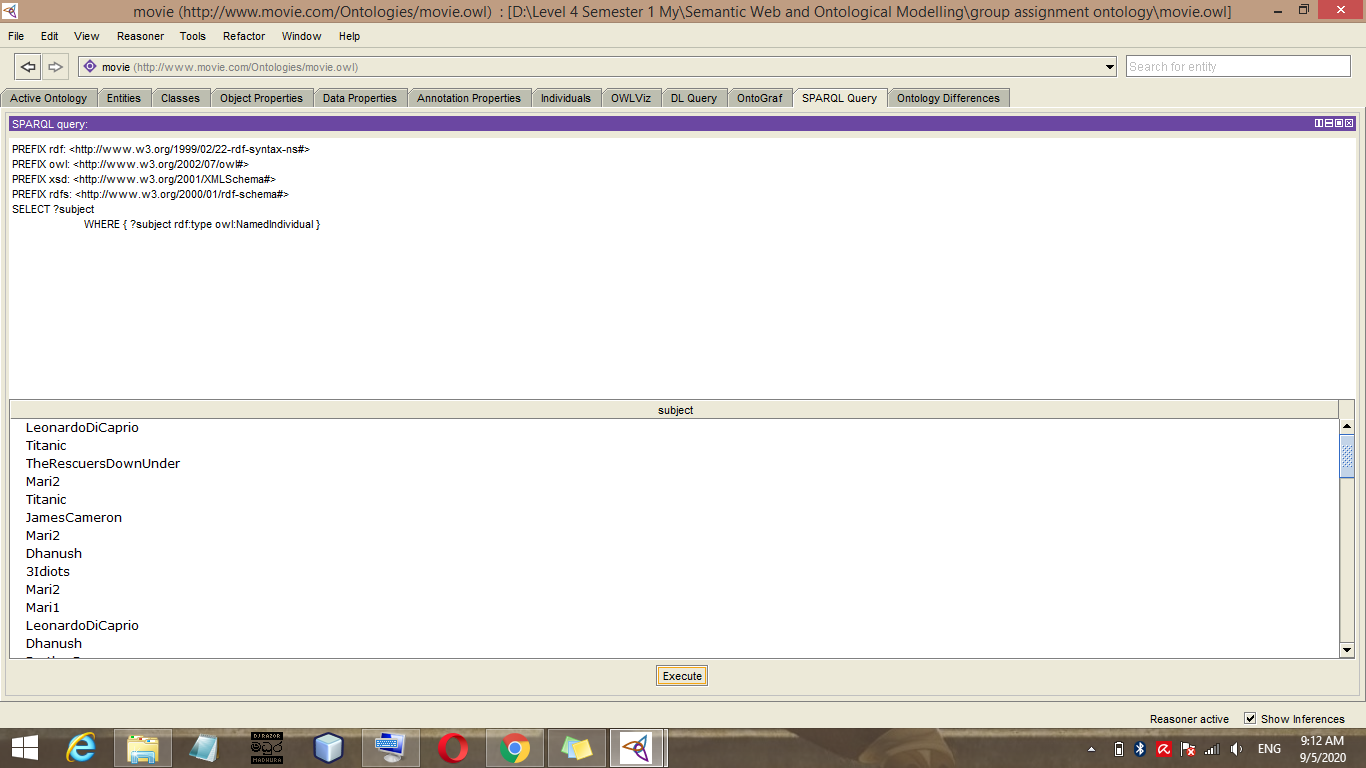


Figure 3.4 - SPARQL query to get all individuals

**Query:-**

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

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

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

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

SELECT ?subject

WHERE { ?subject rdf:type owl:NamedIndividual }

* To get only crew instances

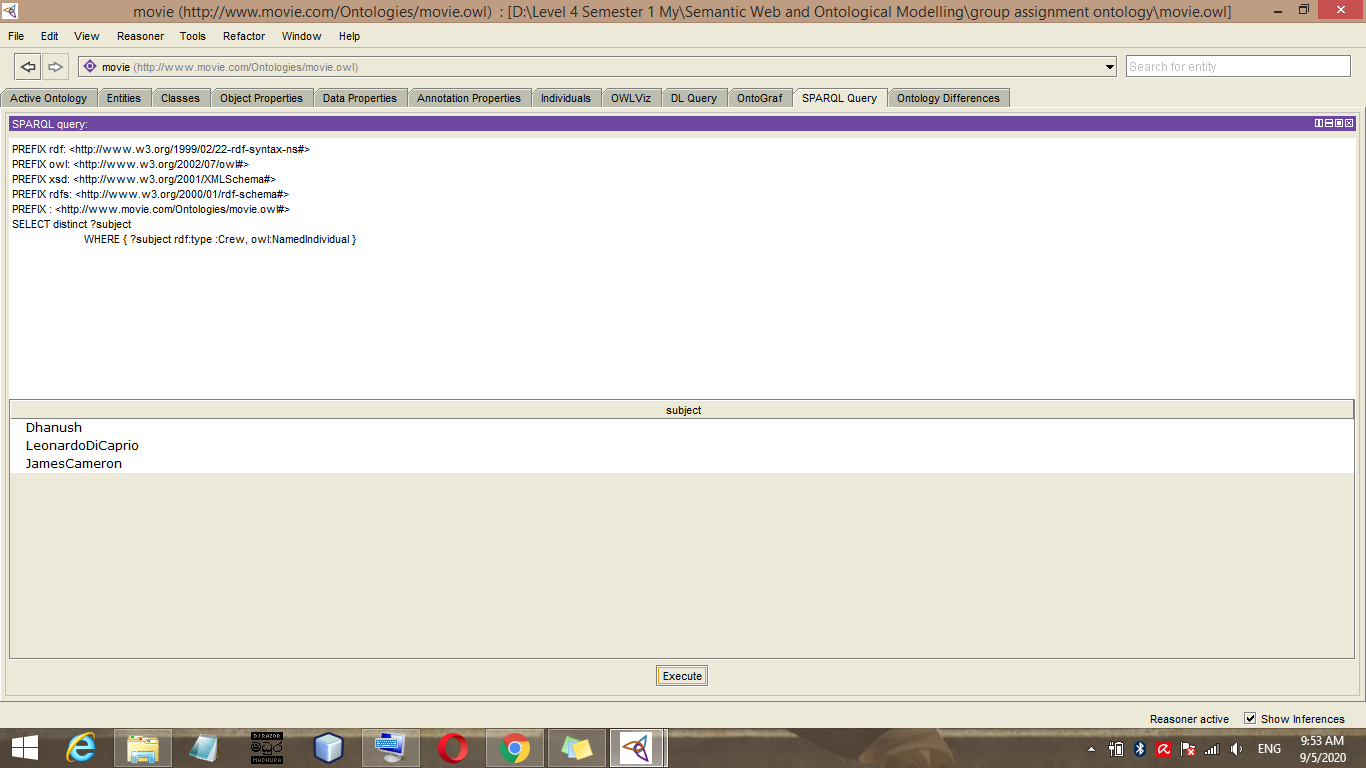


Figure 3.5 - SPARQL query to get only crew instances

**Query:-**

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

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

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

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

PREFIX : <http://www.movie.com/Ontologies/movie.owl#>

SELECT distinct ?subject

WHERE { ?subject rdf:type :Crew, owl:NamedIndividual }

* To get movies and their created year

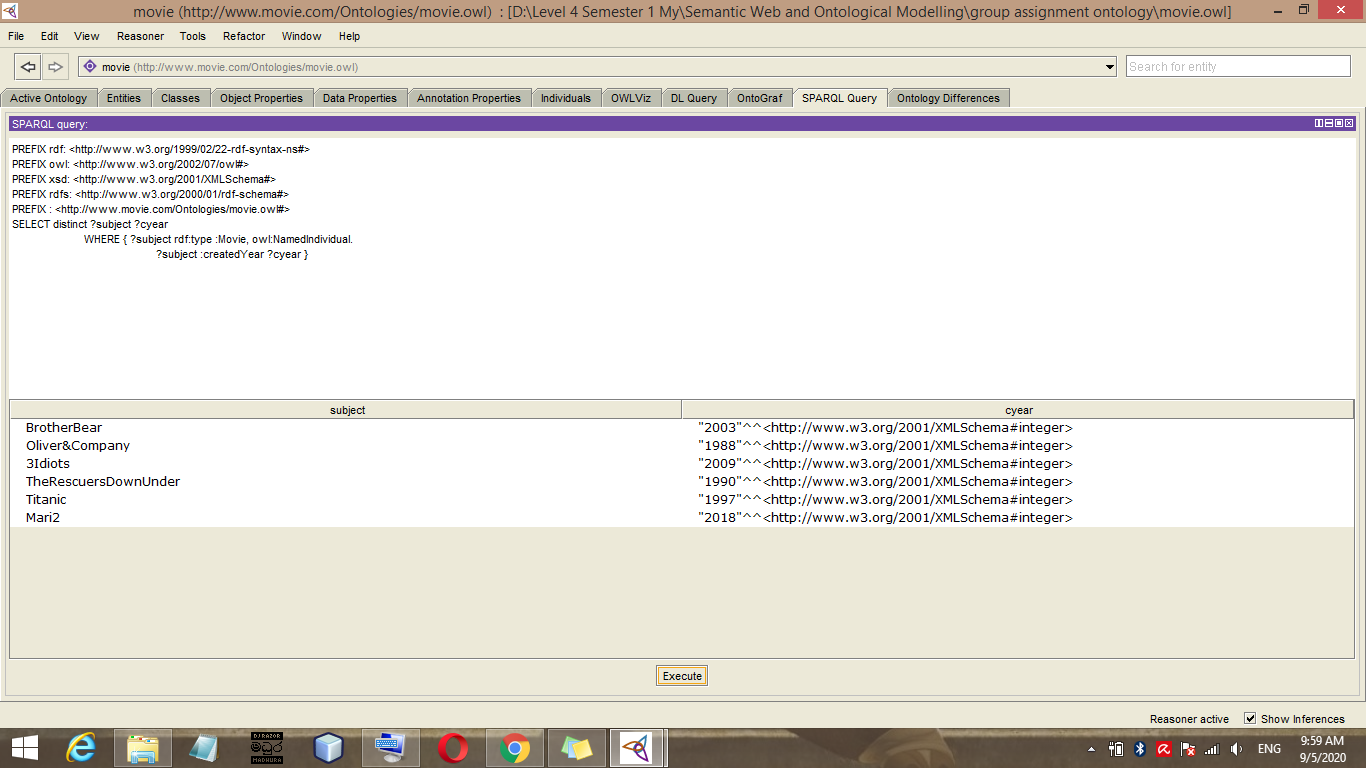


Figure 3.6 - SPARQL query to get get movie instances and their created year

**Query:-**

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

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

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

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

PREFIX : <http://www.movie.com/Ontologies/movie.owl#>

SELECT distinct ?subject ?cyear

WHERE { ?subject rdf:type :Movie, owl:NamedIndividual.

?subject :createdYear ?cyear }

# Conclusion

Data is certainly one important key to the future of the film industry. We have built ontology for the movie/ film industry to support analyzing and understanding data in this industry.

The movie ontology aims to provide a controlled vocabulary to semantically describe movie related concepts (e.g., Movie, Genre, Director, Actor) and the corresponding individuals.