

Σημασιολογικός Ιστός και Ευφυείς Εφαρμογές

Εργασία B:

- OWL-

Τσίγκας Αντωνιος - 2852

Σαρρή Ευθυμία - 3124

Github repository: https://github.com/atsigkas/SemanticWeb_SecProj_OWL

Εισαγωγή

Στα πλαίσια της δεύτερης εργασίας μεταβάλλαμε και εμπλουτίσαμε την οντολογία “CoffeeShor” με στόχο να ενσωματώσουμε κάποιες από τις δομές της OWL. Συνοπτικά, χρησιμοποιήσαμε **αντίστροφες** και **συναρτησιακές** ιδιότητες, πραγματοποιήσαμε αλλαγές σε κάποιες κλάσεις ώστε να ενσωματώσουμε **περιορισμούς κλάσεων** και προσθέσαμε μία νέα κλάση ώστε να αναπαραστήσουμε μία **μεταβατική ιδιότητα**. Ακόμη, χρησιμοποιήσαμε την **τομή κλάσεων** ώστε να ορίσουμε μία νέα κλάση ενώ μέσω των περιορισμών υλοποιήσαμε έμμεσα **ισοδυναμίες κλάσεων**.

Η πρώτη εργασία του μαθήματος αφορούσε την μοντελοποίηση ενός καφέ-μπαρ. Υπενθυμίζουμε ότι η οντολογία καφε-μπαρ (CoffeeShor), χωρίς τις προσθήκες που κάναμε για τις ανάγκες της δεύτερης εργασίας, αποτελούνταν από 7 κύριες κλάσεις. Αυτές μοντελοποιούν:

1. ένα άτομο(Person)
2. κάποιο ποτό (Drink)
3. χρήματα (Money)
4. μία παραγγελία (Order)
5. μία βάρδια (Shift)
6. κάποια πρώτη ύλη (Supply)
7. μία άφιξη στο κατάστημα (Arrival)

Η κλάση **Drink** έχει ως υποκλάσεις **αλκοολούχα (Alcoholic Drink)** και **μη αλκοολούχα ποτά (non Alcoholic Drink)** και κάθε μία από τις υποκλάσεις έχει ως υποκλάσεις κάποια ποτά. Αυτά είναι τα **Alcoholic Drinks** Beer, Rum, Tequila, Whisky και Wine και τα **Non Alcoholic Drinks** Coffee, Juice και Tea. Η κλάση **Person** αποτελείται από τις υποκλάσεις **Client, Staff και Supplier** που μοντελοποιούν τους πελάτες, το προσωπικό και τους προμηθευτές αντίστοιχα. Η κλάση **Staff** έχει ως υποκλάσεις τις **Barista, Cashier και Server**. Κάθε υποκλάση έχει κάποια Object και Data Properties που περιγράφονται αναλυτικά στην αναφορά της πρώτης εργασίας.

Για τις ανάγκες της δευτερης εργασίας υλοποιήθηκαν περιληπτικά τα παρακάτω:

- η κλάση **Rank** που αναπαριστά ένα αξίωμα ενός υπαλλήλου (Staff),
- **προσθήκη υποκλάσεων** στην κλάση Supply
- **δημιουργία νέου τύπου ποτού** (Cocktail)
- **προσθήκη των ιδιοτητων** serves, supplies, makes, isAlcoholic, packed, oversees και hasRank

Η υλοποίηση όλων των παραπάνω περιγράφεται αναλυτικά στη συνέχεια.

Υλοποίηση

Η υλοποίησή μας ξεκίνησε με την μετατροπή ιδιοτήτων της οντολογίας μας σε **αντίστροφες** (inverse) και **συναρτησιακές** (functional) όπου ήταν εφικτό. Συγκεκριμένα, δημιουργήθηκε η νέα ιδιότητα **serves** με domain έναν Server και range κάποιο Order και ορίστηκε ως αντίστροφη (inverse Of) της υπάρχουσας ιδιότητας servedBy που έχει αντίστροφα domain και range από την serves. Παρόμοια, προστέθηκαν οι ιδιότητες **supplies** και **makes** ως αντίστροφες των ιδιοτήτων suppliedBy και madeBy αντίστοιχα. Σε κάθε περίπτωση, τα domain και range των νέων ιδιοτήτων αντιστρέφονται σε σχέση με εκείνα των αντίστροφων ιδιοτήτων τους. Δηλαδή, η ιδιότητα supplies έχει domain και range Supplier και Supply αντίστοιχα, ενώ η makes Barista και Order αντίστοιχα. Τέλος, η υπάρχουσα ιδιότητα **birthday** ορίστηκε ως συναρτησιακή, καθώς ένα άτομο μπορεί να έχει μόνο μία ημερομηνία γέννησης αλλά μία ημερομηνία γέννησης μπορεί να αντιστοιχεί σε παραπάνω από ένα άτομα.

Με στόχο να υλοποιήσουμε μία **μεταβατική ιδιότητα** δημιουργήσαμε την **κλάση Rank** η οποία αναπαριστά το αξίωμα ενός υπαλλήλου καθώς και την **μεταβατική ιδιότητα oversees (επιβλέπει)** με domain και range κάποιο Rank. Ακόμα, ορίσαμε την **ιδιότητα hasRank** με domain έναν υπάλληλο (Staff) και range ένα Rank. Για την κλάση Rank δημιουργήθηκαν τρία στιγμιότυπα, που αναπαριστούν τα τρία διαφορετικά αξιώματα που θεωρούμε ότι ορίζονται στο κατάστημα. Αυτά είναι τα supervisor, head και junior, με σειρά από το ανώτερο προς το κατώτερο αξίωμα. Ορίσαμε επίσης ότι ο υπάλληλος supervisor επιβλέπει τον υπάλληλο head και ο υπάλληλος head επιβλέπει τους υπαλλήλους junior.

Επιπλέον, για κάθε υπάρχον στιγμιότυπο τύπου Staff ορίστηκε η ιδιότητα hasRank. Συγκεκριμένα, ορίστηκε ένας supervisor και ένα head υπάλληλος για κάθε τύπου υπαλλήλου (Server, Cashier και Barista) και οι υπόλοιποι ορίστηκαν ως junior. Έτσι, καθώς η ιδιότητα oversees είναι μεταβατική, μέσω του γράφου της οντολογίας προκύπτει ότι ένας supervisor επιβλέπει τόσο τους υπαλλήλους με head rank όσο και τους υπαλλήλους με junior rank, που επιβλέπονται από τους υπαλλήλους με head rank.

Στη συνέχεια, με στόχο να προσθέσουμε περιορισμούς στην οντολογία μας, πραγματοποιήσαμε αλλαγές σε υπάρχουσες κλάσεις προσθέτοντας νέες ιδιότητες και υποκλάσεις. Δημιουργήσαμε τη νέα **boolean** ιδιότητα **isAlcoholic** που έχει domain κάποιο Drink και **range** true ή false. Επιπλέον, στην κλάση Supply προσθέσαμε **υποκλάσεις ανάλογα με τον τύπο supply**, και συγκεκριμένα τις υποκλάσεις Beer, Coffee, Juice, Rum, Tea, Tequila, Whiskey, Wine και Other Supply.

Συνολικά, ορίσαμε δύο τύπους περιορισμών: περιορισμούς hasValue και someValuesFrom. Οι **περιορισμοί hasValue** έγιναν στην ιδιότητα isAlcoholic και διαχωρίζουν δύο κλάσεις, τις AlcoholicDrink (αλκοολούχο ποτό) και NonAlcoholicDrink (μη αλκοολούχο ποτό), που ορίζονται από τους περιορισμούς τιμής true και false αντίστοιχα. Οι παραπάνω δύο κλάσεις ορίστηκαν ως ισοδύναμες με τις κλάσεις περιορισμού επειδή

θέλουμε να ισχύουν τόσο η αναγκαία όσο και η ικανή συνθήκη. Για παράδειγμα, για ένα αλκοολούχο ποτό, **ισχύουν**:

1. Η **αναγκαία συνθήκη**, που ορίζει ότι εάν ένα στιγμιότυπο ανήκει στην κλάση `AlcoholicDrink` τότε υποχρεωτικά στην ιδιότητα `isAlcoholic` έχει την τιμή `true`.
2. Η **ικανή συνθήκη**, που ορίζει ότι εάν ένα στιγμιότυπο έχει στην ιδιότητα `isAlcoholic` την τιμή `true` τότε πρέπει υποχρεωτικά να ανήκει στην κλάση `AlcoholicDrink`.

Πιο αναλυτικά, οι δύο περιορισμοί ορίζονται ως εξής:

<code>q:AlcoholicDrink owl:equivalentClass [a owl: Restriction; owl: onProperty q:isAlcoholic; owl: hasValue true]</code>	<code>q:nonAlcoholicDrink owl:equivalentClass [a owl: Restriction; owl: onProperty q:isAlcoholic; owl: hasValue false]</code>
---	---

Οι **περιορισμοί `someValuesFrom`** έγιναν πάνω στην ιδιότητα `'made With'`, η οποία προϋπήρχε στην οντολογία μας. Υπενθυμίζουμε ότι έχει ως **domain** την κλάση `Drink` και ως **range** στιγμιότυπα της κλάσης `Supply`. Οι παραπάνω περιορισμοί διαχωρίζουν τα διάφορα είδη ποτών, όπου κάθε ποτό φτιάχνεται από κάποιο τύπο `supply`, αλλά όχι αποκλειστικά από αυτόν τον τύπο. Για παράδειγμα, η κλάση `Wine` (κρασί) ορίζεται ως η κλάση των ποτών που φτιάχνονται από τον τύπο `supply WineSupply`. Η κλάση `Wine` ορίζεται ως ισαδύναμη της κλάσης περιορισμού επειδή θέλουμε να ικανοποιείται και η αναγκαία συνθήκη αλλά και η ικανή. Πιο συγκεκριμένα:

1. **Ικανοποιείται η αναγκαία συνθήκη** που ορίζει ότι εάν ένα στιγμιότυπο ποτού `Drink` ανήκει στην κλάση `Wine` τότε θα πρέπει τουλάχιστον μία από τις τιμές της ιδιότητας `'made With'` να προέρχεται από την κλάση `WineSupply`.
2. **Ικανοποιείται η ικανή συνθήκη** που ορίζει ότι εάν ένα στιγμιότυπο ποτού `Drink` έχει στην ιδιότητα `'made With'` τιμές και από άλλες κλάσεις εκτός της `WineSupply` τότε και πάλι είναι ποτό `Wine` (μπορεί π.χ. να είναι κάποιο `Cocktail`).

Ενδεικτικά, παρακάτω παρουσιάζεται ο ορισμός των κλάσεων `Wine` και `Juice`. Όμοια ορίζονται και οι υπόλοιπες υποκλάσεις της `Drink`, δηλαδή οι `Beer`, `Rum`, `Tequila`, `Whisky`, `Wine`, `Coffee`, `Juice` και `Tea`.

<code>q:Wine owl:subClassOf [a owl: Restriction; owl: onProperty q:madeWith; owl: someValuesFrom q: WineSupply]</code>	<code>q:Juice owl:subClassOf [a owl: Restriction; owl: onProperty q:madeWith; owl: someValuesFrom q: JuiceSupply]</code>
--	---

Οι περιορισμοί `someValuesFrom` θα μπορούσαν να μοντελοποιηθούν και ως `allValuesFrom`, ώστε κάθε τύπος ποτού να ορίζεται αυστηρά από έναν συγκεκριμένο τύπο `Supply`. Ωστόσο, προτιμήσαμε να χρησιμοποιήσουμε περιορισμούς `someValuesFrom` ώστε να μπορέσουν να ενσωματωθούν στην ιδιότητα `made With` και άλλοι τύποι `Supply`, όπως για παράδειγμα πάγος, γάλα κτλ χωρίς να χρειάζεται η δημιουργία νέων ιδιοτήτων και κλάσεων. Ταυτόχρονα, η χρήση περιορισμού `someValuesFrom` επέτρεψε τον ορισμό των ποτών τύπου `Cocktail` που περιγράφονται αναλυτικά στη συνέχεια. Γενικά, κάνοντας τις κατάλληλες προσαρμογές θα μπορούσαμε να χρησιμοποιήσουμε οποιονδήποτε από τους παραπάνω

τύπους περιορισμών, ωστόσο για λόγους απλότητας της οντολογίας επιλέξαμε τον περιορισμό `someValuesFrom`.

Εκτός από τους περιορισμούς `hasValue` και `someValuesFrom`, χρησιμοποιήσαμε και την δυνατότητα που δίνει η OWL για ορισμό **τομής κλάσεων** δημιουργώντας την κλάση αλκοολούχων ποτών **Cocktail**. Σύμφωνα με τον περιορισμό `someValuesFrom`, με τον οποίο ορίζονται οι υπόλοιπες κλάσεις ποτών, θεωρήσαμε ότι ένα **Cocktail** φτιάχνεται από χυμό (Juice) και ρούμι (Rum) και μπορεί να οριστεί ως ισοδύναμη με την τομή των κλάσεων Juice και Rum. Πιο συγκεκριμένα, εάν ένα ποτό φτιάχνεται από χυμό και ρούμι τότε θεωρείται και χυμός αλλά και ρούμι (λόγω του περιορισμού `someValuesFrom`). Έτσι, ορίσαμε την τομή των δύο κλάσεων ως μία νέα κλάση, την **Cocktail**. Χρησιμοποιήσαμε και πάλι την ισοδυναμία, ώστε εάν ένα στιγμιότυπο ανήκει στην κλάση **Cocktail**, τότε οπωσδήποτε στην ιδιότητα `made With` έχει τιμές που προέρχονται από Supply τύπου Juice και Rum. Αντίστροφα, εάν ένα στιγμιότυπο έχει στην ιδιότητα `made With` τιμές από τους παραπάνω τύπους, τότε ανήκει στην κλάση **Cocktail**.

Τέλος, δημιουργήσαμε την **ιδιότητα packed** που αναπαριστά ένα συσκευασμένο προϊόν. Θεωρήσαμε ότι αφού όλα τα προϊόντα παρέχονται συσκευασμένα στο κατάστημα, η ιδιότητα `packed` ορίστηκε ως `subproperty` στην ιδιότητα `supplies`. Ορίζοντας τις δύο ως ισοδύναμες καταφέραμε να εμφανίσουμε την ισοδυναμία μεταξύ των ιδιοτήτων `supplies` και `packed`.

Ερωτήματα SPARQL

Σύμφωνα με τις απαιτήσεις της εργασίας, υλοποιήσαμε ερωτήματα σε γλώσσα Sparql στο περιβάλλον του Protege. Στόχος μας ήταν να δημιουργήσουμε ερωτήματα που έχουν νόημα και αναδεικνύουν την ορθή σημασιολογία της εμπλουτισμένης οντολογίας. Τα ερωτήματα καθώς και τα αποτελέσματά τους παρατίθενται παρακάτω αναλυτικά

1. “Ποιους υπαλλήλους επιβλέπει ο κάθε υπάλληλος ανάλογα με την ιεραρχία στην οποία βρίσκεται.” Το συγκεκριμένο ερώτημα υλοποιήθηκε ώστε να αναδείξει την σημασιολογία της μεταβατικής ιδιότητας `oversees`. Για παράδειγμα, παρατηρείται ορθά ότι οι υπάλληλοι που βρίσκονται πιο πάνω στην ιεραρχία, ανεξαρτήτου πόστου με το Rank `supervisor` επιβλέπουν όλους τους αντίστοιχους υπαλλήλους πάλι ανεξαρτήτου πόστου με χαμηλότερα ranks (`head` και `junior`).

Snap SPARQL Query:				
PREFIX CoffeeShop: <http://test1.project.rdfs/CoffeeShop#> 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 xsd: <http://www.w3.org/2001/XMLSchema#>				
SELECT DISTINCT (?who AS ?PersonStaff) (?what AS ?With_What_Rank_Oversees) (?whome AS ?otherPersonStaff) (?z AS ?that_has_Rank) WHERE { ?who CoffeeShop:hasRank ?what. ?what CoffeeShop:oversees ?z. ?whome CoffeeShop:hasRank ?z } Order by ?who				
CoffeeShop:Barista_1	CoffeeShop:supervisor	CoffeeShop:Server_4	CoffeeShop:junior	
CoffeeShop:Barista_1	CoffeeShop:supervisor	CoffeeShop:Barista_2	CoffeeShop:head	
CoffeeShop:Barista_1	CoffeeShop:supervisor	CoffeeShop:Server_2	CoffeeShop:head	
CoffeeShop:Barista_2	CoffeeShop:head	CoffeeShop:Barista5	CoffeeShop:junior	
CoffeeShop:Barista_2	CoffeeShop:head	CoffeeShop:Barista_3	CoffeeShop:junior	
CoffeeShop:Barista_2	CoffeeShop:head	CoffeeShop:Barista_4	CoffeeShop:junior	
CoffeeShop:Barista_2	CoffeeShop:head	CoffeeShop:Server_3	CoffeeShop:junior	
CoffeeShop:Barista_2	CoffeeShop:head	CoffeeShop:Server_4	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Barista5	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Barista_3	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Barista_4	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Server_3	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Server_4	CoffeeShop:junior	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Barista_2	CoffeeShop:head	
CoffeeShop:Server_1	CoffeeShop:supervisor	CoffeeShop:Server_2	CoffeeShop:head	
CoffeeShop:Server_2	CoffeeShop:head	CoffeeShop:Barista5	CoffeeShop:junior	

2. “Ο αριθμός των παραγγελιών που σερβίρει κάθε υπάλληλος” και “Πόσες παραγγελίες σερβιρίστηκαν από κάθε υπάλληλο”. Τα ερωτήματα αυτά υλοποιήθηκαν ώστε να αναδείξουν την λειτουργία της αντίστροφης ιδιότητας servedBy-serves. Παρατηρούμε ορθά ότι το αποτέλεσμα των δύο ερωτημάτων είναι το ίδιο.

<pre> SELECT ?x (COUNT(?x) AS ?ordersServed) WHERE { ?x rdf:type CoffeeShop:Server . ?x CoffeeShop:serves ?y } GROUP BY ?x ORDER BY DESC(?ordersServed) </pre>	
Execute	
?x	?ordersServed
CoffeeShop:Server_3	6
CoffeeShop:Server_4	4
CoffeeShop:Server_2	3
CoffeeShop:Server_1	2

<pre> SELECT ?x (COUNT(?x) AS ?ordersServed) WHERE { ?y rdf:type CoffeeShop:Order . ?y CoffeeShop:servedBy ?x } GROUP BY ?x ORDER BY DESC(?ordersServed) </pre>	
Execute	
?x	?ordersServed
CoffeeShop:Server_3	6
CoffeeShop:Server_4	4
CoffeeShop:Server_2	3
CoffeeShop:Server_1	2

3. “Υπάρχει καταχωρημένο άτομο στην οντολογία που γεννήθηκε πριν τις 1-12-1960 ”. Για το ερώτημα αυτό χρησιμοποιήσαμε την δομή ASK της sparql που επιστρέφει boolean αποτέλεσμα.

SPARQL query:
<pre> PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> PREFIX CoffeeShop: <http://test1.project.rdfs/CoffeeShop#> ASK WHERE { ?any CoffeeShop:Person ?z. ?z CoffeeShop:birthday ?b. FILTER (?b < "1960-12-01"^^xsd:date) } </pre>
Result
False

CoffeeShop:Server_3	1998-11-16^^<internal:anonymous-constants>
CoffeeShop:Server_4	2001-09-01^^<internal:anonymous-constants>
CoffeeShop:Supplier_1	1973-01-30^^<internal:anonymous-constants>
CoffeeShop:Supplier_2	1969-08-22^^<internal:anonymous-constants>
CoffeeShop:Supplier_3	1964-09-04^^<internal:anonymous-constants>
CoffeeShop:Supplier_4	1974-10-21^^<internal:anonymous-constants>
CoffeeShop:Supplier_5	1997-12-08^^<internal:anonymous-constants>
CoffeeShop:Supplier_6	1987-12-02^^<internal:anonymous-constants>

4. Το παρακάτω ερώτημα αναδεικνύει την ισοδυναμία των ιδιοτήτων packed και supplies, που όπως περιμένουμε βγάζουν τα ίδια αποτελέσματα εφόσον είναι ισοδύναμες.

Snap SPARQL Query:	Snap SPARQL Query:																										
<pre>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 xsd: <http://www.w3.org.2001/XMLSchema> PREFIX CofShop: <http://test1.project.rdfs/CoffeeShop#> SELECT ?x ?z WHERE { ?x CofShop:supplies ?z #?x CofShop:packed ?z }</pre>	<pre>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 xsd: <http://www.w3.org.2001/XMLSchema> PREFIX CofShop: <http://test1.project.rdfs/CoffeeShop#> SELECT ?x ?z WHERE { #?x CofShop:supplies ?z ?x CofShop:packed ?z }</pre>																										
<table><thead><tr><th>?x</th><th>?z</th></tr></thead><tbody><tr><td>CofShop:Supplier_1</td><td>CofShop:Bottle_Juice</td></tr><tr><td>CofShop:Supplier_1</td><td>CofShop:Ground_coffee</td></tr><tr><td>CofShop:Supplier_1</td><td>CofShop:Oranges</td></tr><tr><td>CofShop:Supplier_1</td><td>CofShop:Bottle_Tea</td></tr><tr><td>CofShop:Supplier_1</td><td>CofShop:Bag_Tea</td></tr><tr><td>CofShop:Supplier_2</td><td>CofShop:Beans_Coffee</td></tr><tr><td>CofShop:Supplier_3</td><td>CofShop:Bottle_Wine</td></tr><tr><td>CofShop:Supplier_5</td><td>CofShop:Bottle_Beer</td></tr><tr><td>CofShop:Supplier_5</td><td>CofShop:Bottle_Tequila</td></tr><tr><td>CofShop:Supplier_5</td><td>CofShop:Bottle_Whisky</td></tr><tr><td>CofShop:Supplier_6</td><td>CofShop:Ice</td></tr><tr><td>CofShop:Supplier_6</td><td>CofShop:Milk</td></tr></tbody></table>	?x	?z	CofShop:Supplier_1	CofShop:Bottle_Juice	CofShop:Supplier_1	CofShop:Ground_coffee	CofShop:Supplier_1	CofShop:Oranges	CofShop:Supplier_1	CofShop:Bottle_Tea	CofShop:Supplier_1	CofShop:Bag_Tea	CofShop:Supplier_2	CofShop:Beans_Coffee	CofShop:Supplier_3	CofShop:Bottle_Wine	CofShop:Supplier_5	CofShop:Bottle_Beer	CofShop:Supplier_5	CofShop:Bottle_Tequila	CofShop:Supplier_5	CofShop:Bottle_Whisky	CofShop:Supplier_6	CofShop:Ice	CofShop:Supplier_6	CofShop:Milk	
?x	?z																										
CofShop:Supplier_1	CofShop:Bottle_Juice																										
CofShop:Supplier_1	CofShop:Ground_coffee																										
CofShop:Supplier_1	CofShop:Oranges																										
CofShop:Supplier_1	CofShop:Bottle_Tea																										
CofShop:Supplier_1	CofShop:Bag_Tea																										
CofShop:Supplier_2	CofShop:Beans_Coffee																										
CofShop:Supplier_3	CofShop:Bottle_Wine																										
CofShop:Supplier_5	CofShop:Bottle_Beer																										
CofShop:Supplier_5	CofShop:Bottle_Tequila																										
CofShop:Supplier_5	CofShop:Bottle_Whisky																										
CofShop:Supplier_6	CofShop:Ice																										
CofShop:Supplier_6	CofShop:Milk																										
12 results																											

5. Το τελευταίο ερώτημα αναδεικνύει τη σημασιολογία της κλάσης Cocktail. Σε ερωτήματα που αφορούν για παράδειγμα την κλάση Juice εμφανίζονται και στιγμιότυπα της κλάσης Cocktail, καθώς είναι υποκλάση της Juice. Επίσης σε ερωτήματα που αφορούν την ιδιότητα made With σε στιγμιότυπα Cocktail εμφανίζονται οπωσδήποτε Supply τύπου Rum και Juice.

Snap SPARQL Query:

```
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 xsd: <http://www.w3.org.2001/XMLSchema>
PREFIX CofShop: <http://test1.project.rdfs/CoffeeShop#>

SELECT ?x
WHERE {
    ?z rdf:type CofShop:Cocktail.
    ?z CofShop:madeWith ?x
}
```

Execute

?x

CofShop:Bottle_Juice

CofShop:Bottle_Rum

Snap SPARQL Query:

```
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 xsd: <http://www.w3.org.2001/XMLSchema>
PREFIX CofShop: <http://test1.project.rdfs/CoffeeShop#>

SELECT DISTINCT ?x
WHERE {
    ?x rdf:type CofShop:Juice.
    #?z rdf:type CofShop:Rum
}
```

Execute

?x

CofShop:Apple_Juice

CofShop:Cocktail1

CofShop:Pineapple_Juice

CofShop:Mixed_Juice

CofShop:Peach_Juice

