

ΑΝΑΠΑΡΑΣΤΑΣΗ ΓΝΩΣΗΣ ΣΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ  
ΑΣΚΗΣΕΙΣ

1. Δίνονται οι ιδιότητες 'hasAmka' και 'isFatherOf' της κλάσης 'person'. Ποια/ές μπορούμε να δηλώσουμε ως FunctionalProperty και ποια/ες ως InverseFunctional στην OWL;

**ΑΠ:** Η hasAmka είναι **FunctionalProperty** και **InverseFunctional**, ενώ η isFatherOf είναι **InverseFunctional**.

2. Δίνονται οι ιδιότητες 'hasClassmate' και 'hasPredecessor'. Ποια/ές μπορούμε να δηλώσουμε ως TransitiveProperty και ποια/ες ως SymmetricProperty στην OWL;

**ΑΠ:** hasClassmate είναι **symmetric** και **transitive**  
hasPredecessor είναι **transitive**

3. Δίνονται οι κλάσεις 'animal' και 'car'. Να ορίσετε στην OWL ότι είναι ξένες μεταξύ τους, με δύο τρόπους.

**ΑΠ:**

**1<sup>ος</sup> τρόπος**

```
<owl:Class rdf:about="#animal">  
  <rdfs:subClassOf>  
    <owl:Class>  
      <owl:complementOf rdf:resource="#car"/>  
    </owl:Class>  
  </rdfs:subClassOf>  
</owl:Class>
```

**2<sup>ος</sup> τρόπος**

```
<owl:Class rdf:about="#animal">  
  <owl:disjointWith rdf:resource="#car"/>  
</owl:Class>
```

4. Δίνεται η παρακάτω περιγραφή:

```
<owl:Class rdf:about="#robot">  
  <rdfs:subClassOf>  
    <owl:Class>  
      <owl:complementOf rdf:resource="#bird"/>  
    </owl:Class>  
  </rdfs:subClassOf>  
</owl:Class>
```

Περιγράψτε τί αναπαριστά, σε φυσική γλώσσα. Εξηγήστε πώς επιτυγχάνεται αυτό.

ΑΠ:

Αναπαριστά το «οι κλάσεις robot και bird είναι ξένες μεταξύ τους».

Η robot δηλώνεται ως υποκλάση του συμπληρώματος της bird. Αυτό σημαίνει ότι η robot και η bird δεν έχουν κοινά στιγμιότυπα, άρα είναι ξένες μεταξύ τους.

5. Εκφράστε σε OWL την πρόταση: «Όλα τα μέλη μιας κτηματομεσιτικής εταιρίας πρέπει να κατέχουν τουλάχιστον ένα ακίνητο». Χρησιμοποιήστε ως ονόματα κλάσεων τα: 'realEstate' (κτηματομεσιτική εταιρία), 'estate' (ακίνητο) και ως ρόλο/ιδιότητα την 'possesses' (κατέχει). **ΕΞΗΓΗΣΤΕ.**

ΑΠ:

```
<owl:Class rdf:about="#realEstateMember">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#possesses"/>
      <owl:someValuesFrom rdf:resource="#estate"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Ορίζουμε την realEstateMember (μέλος κτηματομεσιτικής εταιρίας) ως υποκλάση μιας ανώνυμης κλάσης-περιορισμού, η οποία ορίζει τον περιορισμό someValuesFrom (τουλάχιστον μια τιμή από) στο πεδίο τιμών της ιδιότητας possesses (κατέχει), που είναι τύπου estate (ακίνητο).

6. Ποιες από τις παρακάτω εκφράσεις ανήκουν στο ABox (A) και ποιες στο TBox (T).
- TOMATO: Plant
  - AfricanPlant  $\subseteq$  Plant
  - $\exists$ eats.AfricanPlant  $\subseteq$  Vegetarian
  - (JOHN, GEORGE):  $\neg$ sibling
  - Mother  $\equiv$  Woman  $\cap$   $\exists$ child.T

ΑΠ:

- TOMATO: Plant  $\rightarrow$  ABOX
- AfricanPlant  $\subseteq$  Plant  $\rightarrow$  TBOX
- $\exists$ eats.AfricanPlant  $\subseteq$  Vegetarian  $\rightarrow$  TBOX
- (JOHN, GEORGE):  $\neg$ sibling  $\rightarrow$  ABOX
- Mother  $\equiv$  Woman  $\cap$   $\exists$ child.T  $\rightarrow$  TBOX

7. Χρησιμοποιώντας ως κλάσεις: Person (Πρόσωπο), Employer (Εργοδότης), Employee (Εργαζόμενος), Profession (Επάγγελμα) και ως ρόλους/ιδιότητες: isEmployerOf, isEmployeeOf, hasProfession, να εκφράσετε σε DL τις παρακάτω προτάσεις ΦΓ:

- «Δεν μπορεί κάποιος να είναι ταυτόχρονα και εργοδότης (Employer) και εργαζόμενος (Employee)»

ΑΠ:  $Employer \subseteq \neg Employee$

- «Όλοι οι εργαζόμενοι έχουν έναν εργοδότη και ένα επάγγελμα»

ΑΠ:  $(Employee \subseteq \leq 1 \text{ isEmployeeOf}) \cap (Employee \subseteq \leq 1 \text{ hasProfession})$

- «Η κλάση Employed ορίζεται ως η κλάση που περιέχει πρόσωπα που εργάζονται σε κάποιον εργοδότη»

ΑΠ:  $Employed \equiv Person \cap \exists \text{ isEmployeeOf.Employer}$

8. Για τις παρακάτω εκφράσεις DL, γράψτε τί σημαίνουν σε φυσική γλώσσα. Εκφράστε την πρώτη σε OWL.

- $Female \cap \forall \text{ hasChildren.Human} \cap (\geq 4 \text{ hasChildren})$

ΑΠ: Γυναίκες που έχουν περισσότερα από 4 παιδιά. (Ορισμός πολύτεκνης γυναίκας).

- $\exists \text{ eats.Plant} \subseteq \exists \text{ hasFriend.Vegetarian}$

ΑΠ: Όσοι τρώνε κάποια φυτά έχουν κάποιους φίλους χορτοφάγους.

- $HappyMan \equiv Human \cap \neg Female \cap (\exists \text{ married.Doctor}) \cap \leq 4 \text{ hasChild} \cap \geq 2 \text{ hasChild.Doctor}$

ΑΠ: Ευτυχισμένος άνδρας είναι ο άνδρας που έχει παντρευτεί κάποιο γιατρό και έχει το πολύ 4 παιδιά, εκ των οποίων τουλάχιστον δύο είναι γιατροί.

- $\text{Female} \cap \forall \text{hasChildren.Human} \cap (\geq 4 \text{ hasChildren})$  σε OWL:

ΑΠ:

```
<owl:Class rdf:about="#manyChildWoman">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#female"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasChildren"/>
      <owl:allValuesFrom rdf:resource="#Human"/>
    </owl:Restriction>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasChildren"/>
      <owl:minCardinality rdf:datatype="&xsd:nonNegativeInteger"> 4
    </owl:minCardinality>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

9. Αναπαραστήστε σε DL τις παρακάτω έννοιες, δεδομένων των πρωτογενών εννοιών: Female, Male και Company και των ρόλων: hasChild, hasSibling και worksFor.

- Uncle (Θείος)

ΑΠ:  $\text{Uncle} \equiv \text{Male} \cap \exists \text{hasSibling.} \geq 1 \text{ hasChild}$

- GrandMother (Γιαγιά)

ΑΠ:  $\text{Grandmother} \equiv \text{Female} \cap \exists \text{hasChild.} \geq 1 \text{ hasChild}$

- Γυναίκα που εργάζεται σε εταιρία και έχει περισσότερα από τρία παιδιά

ΑΠ:  $\text{Female} \cap \text{worksFor.Company} \cap \geq 3 \text{ hasChild}$

10. Για την παρακάτω έκφραση DL γράψτε την αντίστοιχη-ισοδύναμη έκφραση ΚΛΠΤ:

$\exists \text{eats.Plant} \subseteq \exists \text{hasFriend.Vegetarian}$

ΑΠ:  $\forall x \exists y (\text{plant}(y) \wedge \text{eats}(x,y)) \rightarrow \exists z \text{hasFriend}(x,z) \wedge \text{vegetarian}(z)$

11. Γράψτε κανόνα SWRL που ορίζει τη σχέση isUncleOf μεταξύ δύο προσώπων. Θεωρείστε την κλάση Person και τις ιδιότητες isBrotherOf και isParentOf.

ΑΠ:  $\text{Person}(?p1) \wedge \text{Person}(?p2) \wedge \text{Person}(?p3) \wedge \text{isParentOf}(?p3,?p2) \wedge \text{isBrotherOf}(?p1,?p3) \rightarrow \text{isUncleOf}(?p1,?p2)$

12. Δίνεται ο παρακάτω κανόνας SWRL. Εκφράστε τον σε φυσική γλώσσα.

$(\text{hasChild} \geq 1)(?x) \rightarrow \text{Parent}(?x)$

ΑΠ: Όποιος έχει τουλάχιστον ένα παιδί είναι γονέας.

13. Εκφράστε ως κανόνα SWRL την παρακάτω πρόταση:

«Δημοσίευση μοναδικού συγγραφέα είναι αυτή που έχει μόνο ένα συγγραφέα»

Θεωρείστε τις κλάσεις Publication, SingleAuthorPublication και την ιδιότητα has Author.

ΑΠ:  $Publication(?p) \wedge (hasAuthor = 1)(?p) \rightarrow SingleAuthorPublication(?p)$

14. Θεωρείστε ότι υπάρχουν οι ιδιότητες hasAge και isFatherOf της κλάσης Person. Γράψτε ένα ερώτημα SQWRL που να επιστρέφει τους πατέρες με ηλικία πάνω από 45 χρόνων.

ΑΠ:  $Person(?p) \wedge hasAge(?p,?age) \wedge isFatherOf(?p,?x) \wedge swrlb:greaterThan(?age,45) \rightarrow sqwrl:select(?p).$

15. Θεωρείστε ότι υπάρχουν οι ιδιότητες hasAge, hasCity και isEmployeeOf της κλάσης Person. Γράψτε ερώτημα SPARQL που επιστρέφει τους εργαζόμενους με ηλικία πάνω από 50 χρόνων και την πόλη τους, αν υπάρχει.

ΑΠ:

```
SELECT ?e ?city
WHERE {?e isEmployeeOf ?x.
      ?e hasAge ?age.
      OPTIONAL {?e hasCity ?city}
      FILTER (?age > 50).
}
```

16. Δίνεται η παρακάτω έννοια:

$\neg(\exists R.(C \cap \neg D) \cup \forall R.(\neg C \cup \neg D))$

α) Να παραχθεί η ΑΚΜ (Αρνητική Κανονική Μορφή)

ΑΠ:

$\neg(\exists R.(C \cap \neg D) \cup \forall R.(\neg C \cup \neg D)) \equiv \forall R. \neg(C \cap \neg D) \cap \exists R. \neg(\neg C \cup \neg D) \equiv \forall R. (\neg C \cup D) \cap \exists R. (C \cap D)$

β) Να αποδειχθεί με τη μέθοδο tableau αν είναι ικανοποιήσιμη ή όχι

ΑΠ:

$S_0 = \{\forall R. (\neg C \cup D) \cap \exists R. (C \cap D)\}$   
 $S_0 \rightarrow_{\cap} S_1 = S_0 \cup \{\forall R. (\neg C \cup D), \exists R. (C \cap D)\}$   
 $S_1 \rightarrow_{\cap} S_2 = S_1 \cup \{\exists R. C, \exists R. D\}$   
 $S_2 \rightarrow_{\cup} S_{31} = S_2 \cup \{\forall R. \neg C\}$   
 $S_2 \rightarrow_{\cup} S_{32} = S_2 \cup \{\forall R. D\}$

Δεν υπάρχει σύγκρουση (clash)!

Άρα η πρόταση είναι ικανοποιήσιμη.

