

## Feuille de T. P. 2 : Interrogation avec SPARQL

### Exercice 1 : Interrogation d'une ontologie pizza avec SPARQL

Ouvrir Protégé. Charger votre ontologie (par exemple **pizza.owl**) dans le répertoire de travail.

Sélectionner dans le menu **Windows** puis dans l'onglet **Tabs** sélectionner **SPARQL query** saisir la requête et cliquer sur *Execute* en bas de la fenêtre.

Tester les requêtes suivantes avec SPARQL :

- Quelles sont tous les triplets de l'ontologie ?  
PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza\_1.owl#>  
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#>  
SELECT DISTINCT \*  
WHERE { ?x ?y ?z }
- Quels sont les concepts de l'ontologie ?  
PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza\_1.owl#>  
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#>  
SELECT DISTINCT ?x  
WHERE { ?x rdf:type owl:Class }
- Quelles sont les *ObjectProperties* ?  
PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza\_1.owl#>  
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#>  
SELECT DISTINCT ?x  
WHERE { ?x rdf:type owl:ObjectProperty }
- Quelles sont les *DatatypeProperties* ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT DISTINCT *
WHERE { ?x rdf:type owl:DatatypeProperty
}

```

— Quelles sont les sous-propriétés des *ObjectProperties* ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT DISTINCT ?p ?q
WHERE { ?p rdfs:subPropertyOf ?q .}

```

— Quelles sont les sous-classes de la classe **NamedPizza** ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>

SELECT DISTINCT *
WHERE { ?p rdfs:subClassOf test:NamedPizza}

```

— Quelles sont les instances du concept *Country* ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT ?p WHERE
{?p a test:Country.
}

```

— Quelles sont les instances des pizzas qui ont un diamètre ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

```

```

PREFIX owl:<http://www.w3.org/2002/07/owl#>
SELECT * WHERE
{?pi test:hasDiameter ?d .
}

```

- Quelles sont les instances des pizzas qui ont un diamètre supérieur à 20 cm

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT * WHERE
{?pi test:hasDiameter ?d .
FILTER (?d > 20)
}

```

- Quelles sont les pizzas nommées qui ont pour garniture du fromage ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT DISTINCT ?pizza
WHERE {
?pizza rdfs:subClassOf test:NamedPizza .
?pizza rdfs:subClassOf ?htopping .
?htopping owl:onProperty test:hasTopping .
?htopping owl:someValuesFrom ?topping .
?topping rdfs:subClassOf test:CheeseTopping .
}

```

- Quelle sont les garnitures de la PizzaMarseillaise ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT ?topping
WHERE
{
test:PizzaMarseillaise rdfs:subClassOf ?htopping.
?htopping owl:onProperty test:hasTopping.
}

```

```
?htopping owl:someValuesFrom ?topping.  
}
```

- Quelles sont les classes définies ?

```
PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>  
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#>  
SELECT DISTINCT ?p ?q  
WHERE { ?p owl:equivalentClass ?q .}
```

- Quelles sont les classes qui ont un nombre de sous-classes supérieur à 3 ?

```
PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>  
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#>  
SELECT DISTINCT (COUNT(?p) as ?nb)  
WHERE { ?p rdfs:subClassOf ?q.}  
GROUP BY ?q  
HAVING(?nb > 3)
```

- Quelles sont les sous-classes qui sont des sous-classes directes (pas de classe intermédiaire entre ces deux classes dans la hiérarchie) ?

```

PREFIX test: <file:///Users/odile/workspace-TPwebserm/pizza_1.owl#>
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#>
SELECT DISTINCT ?d ?s
WHERE { ?d rdfs:subClassOf ?s.
        FILTER NOT EXISTS { ?o rdfs:subClassOf ?s.
                            ?d rdfs:subClassOf ?o.
                            FILTER(?o != ?d)
        }
}

```

## Exercice 2 : Interrogation de plusieurs ontologies avec dbpedia

### Interrogation de dbpedia et foaf avec l'interface d'interrogation de dbpedia

Ouvrir un navigateur.

Pour accéder à l'interface d'interrogation de dbpedia utiliser l'URL :

<https://dbpedia.org/sparql>

- 1) Rechercher le nom du fondateur du World Wide Web. Pour cela utiliser l'ontologie dbpedia et l'ontologie foaf :

- Aller sur la page : [http://dbpedia.org/resource/World\\_Wide\\_Web\\_Foundation](http://dbpedia.org/resource/World_Wide_Web_Foundation)

- Chercher le nom du fondateur : C'est une URI, cliquer sur l'URI, trouver le nom et utiliser ces informations pour formuler la requête avec SPARQL dans l'interface d'interrogation de dbpedia. Pour cela utiliser :

- Le préfixe pour l'ontologie dbpedia :  
*PREFIX dbpedia - owl : < http : //dbpedia.org/ontology/ >*
- Le préfixe pour les ressources de l'ontologie dbpedia :  
*PREFIX dbpedia - r : < http : //dbpedia.org/resource/ >*
- Le préfixe pour l'ontologie foaf :  
*PREFIX foaf : < http : //xmlns.com/foaf/0.1/ >*

Requête 1 :

```
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dbpedia-r: <http://dbpedia.org/resource/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
select ?name where {
dbpedia-r:World_Wide_Web_Foundation dbpedia-owl:foundedBy/foaf:name ?name
}
```

### Interrogation de dbpedia et pizza ensemble

Ouvrir un navigateur.

Pour accéder à l'interface d'interrogation utiliser l'URL :

<http://www.lsis.org/groplan/Papini/pizza.html>

- 2) Rechercher des informations sur les pays qui sont des instances de *Country* dans l'ontologie des pizzas. Pour cela utiliser l'ontologie dbpedia et l'ontologie pizza :

- Reprendre l'URL de l'ontologie des pizzas et chercher des informations sous forme de triplets en utilisant *owl:sameAs*
- pour chercher des informations dans l'ontologie dbpedia utiliser : *SERVICE <https://dbpedia.org/sparql>*
- pour exécuter la requête cliquer sur le bouton (triangle) en haut à droite de l'interface d'interrogation.
- Chercher les IRI (dans dbpedia) des capitales des pays qui sont des instances de *Country*.

Requête 2.1 :

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT distinct ?piz ?countryDBpedia ?capital
WHERE {
?piz a pizza:Country; owl:sameAs ?countryDBpedia.
SERVICE <https://dbpedia.org/sparql>
{
?countryDBpedia dbo:capital ?capital
}
}
```

Requête supplémentaire : donne les 5 premiers triplets d'information sur dbpedia

Requête 2.1.1 :

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT *
WHERE {
    ?piz a pizza:Country; owl:sameAs ?countryDBpedia.
    SERVICE <https://dbpedia.org/sparql>
    {
        ?countryDBpedia ?pred ?obj
    }
    } limit 5
```

— Chercher le nom des capitales des pays qui sont des instances de *Country*

Requête 2.2 :

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT distinct ?piz ?countryDBpedia ?capital
WHERE {
    ?piz a pizza:Country; owl:sameAs ?countryDBpedia.
    SERVICE <https://dbpedia.org/sparql>
    {
        ?countryDBpedia dbo:capital ?capital
    }
    }
```

- Chercher le résumé en français sur les capitales des pays qui sont des instances de *Country*. Requête 2.3 :

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT ?description_FR ?capital_FR WHERE {

    ?piz a pizza:Country; owl:sameAs ?country.

    SERVICE <https://dbpedia.org/sparql>
    {
        ?country dbo:capital/dbo:abstract ?description.
        FILTER (lang(?description) = 'fr')
        BIND(STR(?description) AS ?description_FR)

        ?country dbo:capital/rdfs:label ?capital
        FILTER (lang(?capital) = 'fr')
        BIND(STR(?capital) AS ?capital_FR)
    }
}
```



### Exercice 3 : Interrogation avec WIKIDATA

#### Interrogation de wikidata avec l'interface d'interrogation de wiki- data

Ouvrir un navigateur.

Pour accéder à l'interface d'interrogation , utiliser l'URL :

<https://query.wikidata.org>

Pour trouver les identifiants wikidata des classes et des propriétés :

Ouvrir un navigateur

Aller sur wikipedia et saisir le nom de la classe : sur la colonne de gauche cliquer sur **Elément Wikidata**, à côté du nom de la classe s'affiche l'identifiant wikidata (Pour pizza Q177), descendre et trouver l'identifiant de la propriété (Pour has part P527)

Préfixe pour les classes : wd

Préfixe pour les propriétés : wdt

Les prefixes WIKIDATA

```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wds: <http://www.wikidata.org/entity/statement/>
PREFIX wdv: <http://www.wikidata.org/value/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX wikibase: <http://wikiba.se/ontology#>
PREFIX p: <http://www.wikidata.org/prop/>
PREFIX ps: <http://www.wikidata.org/prop/statement/>
PREFIX pq: <http://www.wikidata.org/prop/qualifier/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bd: <http://www.bigdata.com/rdf#>
```

Requête avec WIKIDATA : De quoi est constituée une pizza ?

```
SELECT ?t
WHERE
{
    wd:Q177 wdt:P527 ?t.
    SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
}
```

## Interrogation de pizza, dbpedia et wikidata ensemble

Ouvrir un navigateur.

Pour accéder à l'interface d'interrogation utiliser l'URL :

<http://www.lsis.org/groplan/Papini/pizza.html>

— Reprendre l'URL de l'ontologie des pizzas et chercher des informations sous forme de triplets en utilisant *owl : sameAs*

— pour chercher des informations dans l'ontologie dbpedia utiliser :

*SERVICE* < <https://dbpedia.org/sparql> >

— pour chercher des informations dans la base de données wikidata utiliser :

*SERVICE* < <https://query.wikidata.org/bigdata/namespace/wdq/sparql> >

Requête : les 3 pizza, wikidata, dbpedia MARCHE accès SERVICE WIKI-DATA .....

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

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

```
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
```

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

```
PREFIX dbo: <http://dbpedia.org/ontology/>
```

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
SELECT distinct *
```

```
WHERE {
```

```
  pizza:America owl:sameAs ?countryWikidata; owl:sameAs ?countryDBpedia.
```

```
  SERVICE <https://query.wikidata.org/bigdata/namespace/wdq/sparql>{  
    ?countryWikidata wdt:P41 ?images.  
  }
```

```
  SERVICE <https://dbpedia.org/sparql>  
  {?countryDBpedia ?d ?e.  
  }
```

```
}
```

```
limit 10
```

sameAs avec Wikidata

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>

SELECT distinct *
WHERE {
  pizza:America owl:sameAs ?countryDBpedia.
  pizza:America owl:sameAs ?countryWikidata.

  # commentaire
  SERVICE <https://query.wikidata.org/bigdata/namespace/wdq/sparql>{
    ?countryWikidata wdt:P41 ?images.
  }
  SERVICE <https://dbpedia.org/sparql>
  {?countryDBpedia owl:sameAs ?countryWikidata.
  }

}

limit 10
```

Rechercher pour l'individu Sfax (instance de Region) de l'ontologie pizza, la description en langue arabe sur l'ontologie dbepdia et l'image sur la base de données wikidata

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pizza: <file:///Users/odile/workspace-TPwebsem/pizza_1.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```

SELECT distinct ?description_ar ?images
WHERE {
pizza:Sfax owl:sameAs ?SfaxWikidata; owl:sameAs ?SfaxDBpedia.

    SERVICE <https://query.wikidata.org/bigdata/namespace/wdq/sparql>{
    ?SfaxWikidata wdt:P18 ?images.
    }

    SERVICE <https://dbpedia.org/sparql>{
?SfaxDBpedia dbo:abstract ?description.
FILTER (lang(?description) = 'ar')
BIND(STR(?description) AS ?description_ar)
    }
}

```

#### Pour en savoir plus sur les requêtes avec WIKIDATA :

[https://www.mediawiki.org/wiki/Wikidata\\_Query\\_Service/User\\_Manual/fr](https://www.mediawiki.org/wiki/Wikidata_Query_Service/User_Manual/fr)

[https://www.wikidata.org/wiki/Wikidata:SPARQL\\_query\\_service/](https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service/)

[A\\_gentle\\_introduction\\_to\\_the\\_Wikidata\\_Query\\_Service/fr](A_gentle_introduction_to_the_Wikidata_Query_Service/fr)

[https://www.wikidata.org/wiki/Wikidata:SPARQL\\_tutorial](https://www.wikidata.org/wiki/Wikidata:SPARQL_tutorial)

Et aussi un nouveau tutoriel sur wikidata :

<https://wdqs-tutorial.toolforge.org/>

#### Exercice 4 : règles SWRL et Interrogation avec SQWRL

A partir charger l'ontologie **family.owl**) dans vore répertoire de travail. Ouvrir Protégé. Ouvrir l'ontologie **family.owl**) dans le répertoire de travail. Vérifier que L'URI de l'ontologie chargée est bien l'ontologie qui se trouve dans votre répertoire de travail :

*file : ///Users/ chemin jusqu'au fichier /nom\_fichier.owl*

L'ontologie comporte :

- **les concepts suivants** : Personne, Homme, Femme, GrandMere, Adulte, Auteur, Publication
- **les rôles suivants** : aEnfant, aFille, aFrere, aAge, aOncle, aAgeConducteur, aNumeroTel, aNumeroTelInt, coopereAvec, aSalaireEnEuros, aSalaireEnDollars

**Tester les règles en SWRL** : Ecrire et tester les règles suivantes

- 1) Les personnes qui ont un enfant de sexe féminin sont des personnes qui ont une fille.

```
Personne(?x) ^ aEnfant(?x, ?y) ^ Femme(?y) -> aFille(?x, ?y)
```

- 2) Les personnes de sexe féminin qui ont un petit-enfant sont des grand-mères.

```
Personne(?x) ^ ex:Femme(?x) ^ aEnfant(?x, ?y) ^ aEnfant(?y, ?z)
-> GrandMere(?x)
```

- 3) Les frères des parents d'une personne sont ses oncles.

```
aEnfant(?y, ?x) ^ aFrere(?y, ?z) -> aOncle(?x, ?z)
```

- 4) Les personnes âgées de plus de 18 ans sont des adultes.

```
Personne(?x) ^ aAge(?x, ?a) ^ swrlb:greaterThan(?a, 17) -> Adulte(?x)
```

- 5) Les personnes âgées de plus de 18 ans et de moins de 80 ans ont l'âge d'être conducteur.

```
Personne(?x) ^ aAge(?x, ?a) ^ swrlb:greaterThan(?a, 17)
^ swrlb:lessThan(?a, 80) -> aAgeConducteur(?x, true)
```

- 6) Les personnes qui ont un numéro de téléphone qui commence par + ont des numéros de téléphone internationaux

```
Personne(?x) ^ aNumeroTel(?x, ?n) ^ swrlb:startsWith(?n, "+")
-> aNumeroTelInt(?x, true)
```

- 7) Deux auteurs de publications coopèrent.

```
Publication(?p) ^ aAuteur(?p, ?x) ^ aAuteur(?p, ?y)
^ owl:differentFrom(?x, ?y) -> coopereAvec(?x, ?y)
```

- 8) Si le salaire en euros d'une personne est de  $S$  alors il est de  $S \times 1.20$  en dollars.

```
Personne(?p) ^ aSalaireEnEuros(?p, ?e) ^ swrlb:multiply(?d, ?e, 1.2)
-> aSalaireEnDollars(?p, ?d)
```

**Interroger avec SQWRL** : Ecrire et tester les requêtes suivantes

1) Quel est l'âge de la personne la plus âgées ?

```
Person(?p) ^ hasAge(?p, ?a) -> sqwrl:max(?a)
```

2) Quel est nombre de personnes de moins de 18 ans ?

```
Person(?p) ^ hasAge(?p, ?a) ^ swrlb:lessThan(?a, 18) -> sqwrl:count(?p)
```

3) Par personne, quel est le nombre de personnes qui coopèrent avec cette personne ?

```
Person(?p) ^ Person(?q) ^ coopereWith(?p, ?q) -> sqwrl:select(?p) ^ sqwrl:count(?q)
```

4) Quels sont les personnes et les salaires en euros par ordre croissant des salaires ?

```
Person(?p) ^ hasSalaryInEuros(?p, ?s) -> sqwrl:select(?p, ?s) ^ sqwrl:orderBy(?s)
```

5) Quels sont les personnes et les salaires en livre par ordre décroissant des salaires ?  
Préalablement écrire la règle :

```
Person(?p) ^ hasSalaryInEuros(?p, ?euros) ^ swrlb:multiply(?pounds, ?euros, 0.89) ->  
hasSalaryInPounds(?p, ?pounds)
```

Ensuite écrire la requête :

```
hasSalaryInPounds(?p, ?pounds) -> sqwrl:select(?p, ?pounds) ^ sqwrl:orderByDescending(?pounds)
```

6) Quelles sont les deux premières personnes qui ont l'âge d'être conducteur ?

```
Person(?p) ^ hasSalaryInEuros(?p, ?s) -> sqwrl:select(?p, ?s) ^ sqwrl:orderBy(?s)
```