



WEB SÉMANTIQUE ET ONTOLOGIES

WEB DES DONNÉES

DONNÉES LIÉES (LINKED DATA)

4 – INTERROGER LES DONNÉES AVEC SPARQL

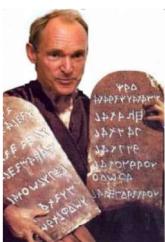
Philippe GENOUD – Danielle ZIEBELIN – Mohamed Wadhah Mabrouk

LIG

Prenom.Nom@imag.fr



Querying Linked Data with SPARQL



Linked Data: 3rd Principle

When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).



Most apps use only a subset of the stack

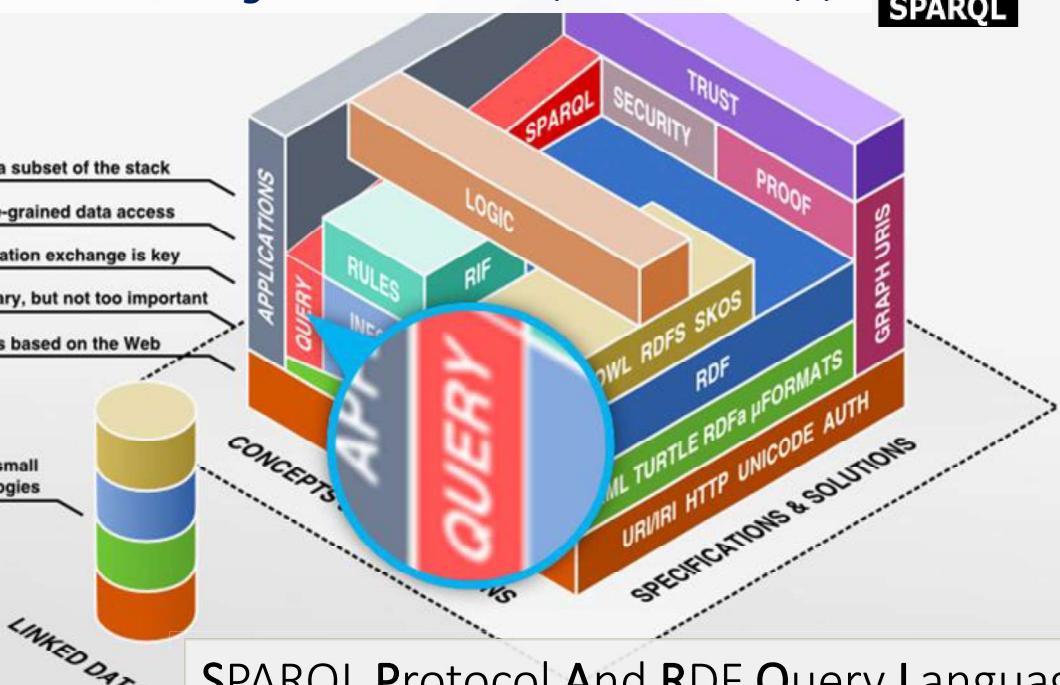
Querying allows fine-grained data access

Standardized information exchange is key

Formats are necessary, but not too important

The Semantic Web is based on the Web

Linked Data uses a small selection of technologies



Exercice SPARQL

- "DBpedia est un projet universitaire et communautaire d'exploration et extraction automatiques de données dérivées de Wikipédia. Son principe est de proposer une version structurée et sous forme de données normalisées au format du web sémantique des contenus encyclopédiques de chaque fiche encyclopédique. DBpedia vise aussi à relier à Wikipédia (et inversement) des ensembles d'autres données ouvertes provenant du Web des données." (<http://fr.wikipedia.org/wiki/DBpedia>).
- L'url `http://dbpedia.org/sparql` correspond à un point d'accès SPARQL (SPARQL endpoint) aux données de DBPedia. Vous allez utiliser celui-ci pour explorer les données RDF de DBpedia.

Dbpedia sparql endpoint

Interface web du point d'accès SPARQL de DBPedia : <https://dbpedia.org/sparql>

SPARQL Query Editor [About](#) [Tables](#) ▾

Conductor Facet Browser Permalink

Extensions: [cxml](#) [save to dav](#) [sponge](#) **User: SPARQL**

Default Data Set Name (Graph IRI)

Query Text

```
select distinct ?Concept where {[] a ?Concept} LIMIT 100
```

Results Format ▾

Execute Query [Reset](#)

Execution timeout milliseconds

Options

- Strict checking of void variables
- Strict checking of variable names used in multiple clauses but not logically connected to each other
- Suppress errors on wrong geometries and errors on geometrical operators (failed operations will return NULL)
- Log debug info at the end of output (has no effect on some queries and output formats)
- Generate SPARQL compilation report (instead of executing the query)

Copyright © 2023 [OpenLink Software](#)
[Virtuoso](#) version 08.03.3329 (42780c0b7c) on Linux (x86_64-generic-linux-glibc25) Single Server Edition (61 GB total memory, 43 GB memory in use)

Question 1

- Question 1: Dans la zone texte prévue à cet effet tapez la requête SPARQL suivante :

```
SELECT ?m ?a WHERE {  
  ?m a dbo:Mountain;  
  dbo:locatedInArea dbr:Nepal;  
  dbo:elevation ?a.  
}
```

- D'après vous que fait cette requête?
- à quoi correspond le a dans le pattern ?m a dbo:Mountain; ? A quelle URI se substitue-t-il ?
- A quelles URIs correspondent les préfixes dbo:, dbr: ?
- Exécuter la requête et observez le résultat obtenu. Correspond-t-il à ce que vous aviez prévu ?

Question 1: solution 1)

- Question 1: Dans la zone texte prévue à cet effet tapez la requête SPARQL suivante :

```
SELECT ?m ?a WHERE {  
  ?m a dbo:Mountain;  
  dbo:locatedInArea dbr:Nepal;  
  dbo:elevation ?a.  
}
```

- D'après vous que fait cette requête?

Cette requête SPARQL recherche dans le graphe DBpedia toutes les ressources de type Montagne situées au Népal qui ont une altitude.

Le résultat produit est l'ensemble des bindings qui répondent à cette question, où ?m est l'URI des ressources trouvées et ?a est l'altitude de celles-ci.

Question 1: solution 2)

- à quoi correspond le **a** dans le pattern ?m **a** dbo:Mountain; ? A quelle URI se substitue-t-il ?

a est un raccourci que proposent Turtle et SPARQL en lieu est place de la propriété type define dans le vocabulaire RDF.

l'URI de celle-ci est <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> que l'on abrège souvent en rdf:type ou rdf: est le préfixe correspondant à <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

Question 1: solution 4)

- A quelles URIs correspondent les préfixes dbo:, dbr: ?

Les prefixes dbo: et dbr: sont implicites dans l'interface web du SPARQL-endpoint de DBpedia. La liste ci-dessous présente à quelles URIs il correspondent

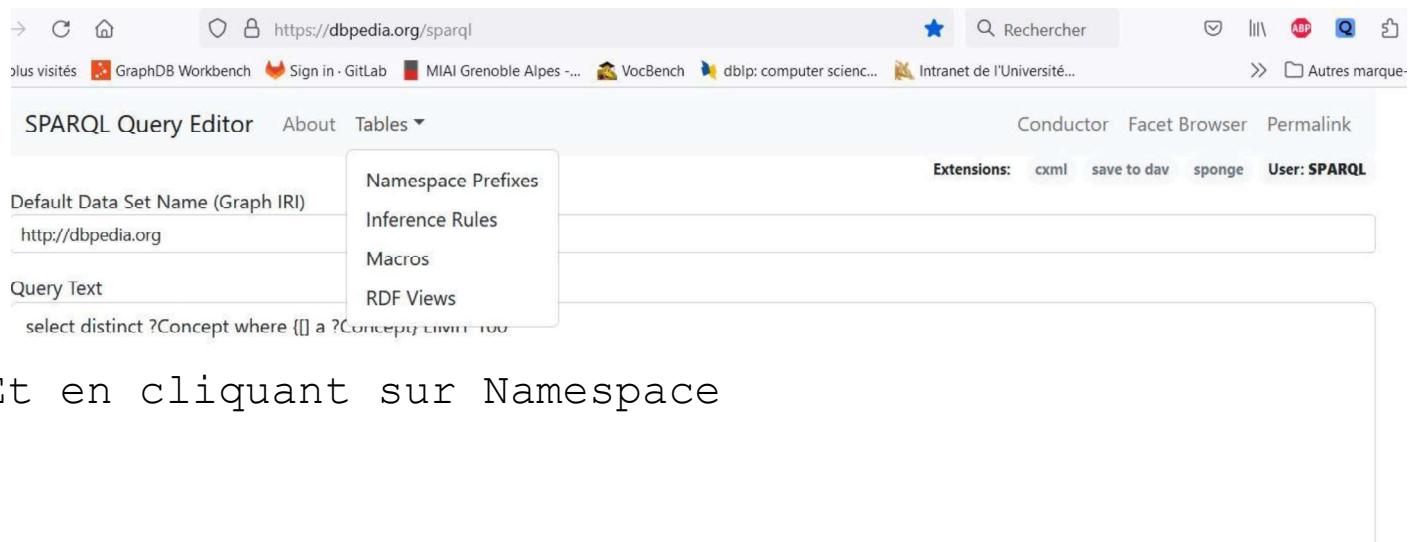
prefixe URI dbo: c'est le préfixe de l'ontologie (le vocabulaire : types et propriétés) de dbpedia <http://dbpedia.org/ontology/>

prefixe URI dbr: c'est le préfixe pour les ressource de dbpedia <http://dbpedia.org/resource/>

On peut trouver la liste des préfixes utilisés implicitement par le SPARQL endpoint de DBpedia en cliquant sur le lien de navigation correspondant dans l'interface web.

Question 1: solution 3)

On peut trouver la liste des préfixes utilisés implicitement par le SPARQL endpoint de DBpedia en cliquant sur le lien de navigation correspondant dans l'interface web.



The screenshot shows a web browser window with the URL <https://dbpedia.org/sparql>. The page is a SPARQL Query Editor. On the left, there is a sidebar with the following options: Default Data Set Name (Graph IRI) set to <http://dbpedia.org>, Query Text containing the SPARQL query `select distinct ?Concept where {} a ?Concept`, and a list of links: Namespace Prefixes, Inference Rules, Macros, and RDF Views. The 'Namespace Prefixes' link is highlighted with a red box. At the top of the page, there is a navigation bar with links to GraphDB Workbench, Sign in · GitLab, MIAI Grenoble Alpes, VocBench, dblp: computer scienc..., and Intranet de l'Université... There are also buttons for Rechercher, Conductor, Facet Browser, Permalink, and User: SPARQL.

Et en cliquant sur Namespace

Question 1: solution 3)

Extrait des namespaces

SPARQL Query Editor

Namespace Prefixes

Prefix	URI
a	http://www.w3.org/2005/Atom
address	http://schemas.talis.com/2005/address/schema#
admin	http://webns.net/mvcb/
as	https://www.w3.org/ns/activitystreams#
atom	http://atomowl.org/ontologies/atomrdf#
aws	http://soap.amazon.com/
b3s	http://b3s.openlinksw.com/
batch	http://schemas.google.com/gdata/batch
bibo	http://purl.org/ontology/bibo/
bif	http://www.openlinksw.com/schemas/bif#
bugzilla	http://www.openlinksw.com/schemas/bugzilla#
c	http://www.w3.org/2002/12/cal/icaltzd#
cb	http://www.crunchbase.com/
cc	http://web.resource.org/cc/
content	http://purl.org/rss/1.0/modules/content/
cv	http://purl.org/captSolo/resume-rdf/0.2/cv#
cvbase	http://purl.org/captSolo/resume-rdf/0.2/base#

Question 1: solution 3)

Extrait des namespaces

Préfixe	Namespace	Description
dbo:	http://dbpedia.org/ontology/	Ontologie DBpedia (concepts, propriétés)
dbr:	http://dbpedia.org/resource/	Ressources concrètes (instances Wikipedia)
dbp:	http://dbpedia.org/property/	Propriétés extraites directement de Wikipedia infoboxes
yago:	http://dbpedia.org/class/yago/	Classes issues du projet YAGO
foaf:	http://xmlns.com/foaf/0.1/	Données sur les personnes
rdfs:	http://www.w3.org/2000/01/rdf-schema#	Schéma RDF standard
rdf:	http://www.w3.org/1999/02/22-rdf-syntax-ns#	Syntaxe RDF
owl:	http://www.w3.org/2002/07/owl#	Ontology Web Language
xsd:	http://www.w3.org/2001/XMLSchema#	Types de données

Question 1: solution 3)

On peut aussi les trouver à l'aide du service web prefix.cc (<https://prefix.cc/>)

La requête SPARQL complète incluant les prefixes serait :

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-
syntax-ns#>
SELECT ?m ?a WHERE {
  ?m rdf:type dbo:Mountain;
  dbo:locatedInArea dbr:Nepal;
  dbo:elevation ?a.
}
```

Question 1: solution 4)

- Exécutez la requête et observez le résultat obtenu. Correspond-t-il à ce que vous aviez prévu ?

```
SELECT ?m ?a WHERE {
  ?m a dbo:Mountain;
  dbo:locatedInArea dbr:Nepal;
  dbo:elevation ?a.
}
```

SPARQL | HTML5 table

m	a
http://dbpedia.org/resource/Pumori	7161.0
http://dbpedia.org/resource/Purbung	6465.0
http://dbpedia.org/resource/Putha_Hiunchuli	7246.0
http://dbpedia.org/resource/Sandakphu	3636.0
http://dbpedia.org/resource/Ombigaichan	6340.0
http://dbpedia.org/resource/Baruntse	7162.0
http://dbpedia.org/resource/Hiunchuli	6441.0
http://dbpedia.org/resource/Paungda_Danda	4755.0
http://dbpedia.org/resource/Saipal	7031.0

Question 2 et 3:

- 1) A quoi correspond l'url <http://dbpedia.org/resource/Saipal> qui apparaît dans la colonne m du tableau résultat ?
- 2) Cliquez sur ce lien. A quoi correspond le résultat obtenu ?
- 3) Le lien était bien <https://dbpedia.org/resource/Saipal> (vous pouvez le vérifier en tapant directement cette url dans la barre d'adresse de votre navigateur), pourtant l'url de la page affichée est <https://dbpedia.org/page/Saipal>.
 - Que s'est-il passé ? Pouvez-vous expliquer ce comportement de votre navigateur ?
 - Que faudrait-il modifier dans la requête HTTP pour obtenir en RDF les données de DBpedia concernant Saipal ?
 - Récupérez ces données au format Turtle. Expliquez comment vous avez procédé.
- Donnez une requête SPARQL qui permet d'obtenir les URIs et l'altitude des dix plus hauts sommets du Népal.

Question 2: solution 1)

- a) A quoi correspond l'url <http://dbpedia.org/resource/Saipal> qui apparait dans la colonne m du tableau résultat ?

A chaque page Wikipedia dont l'URL de la forme <https://en.wikipedia.org/wiki/XXXX> correspond une ressource DBpedia dont l'URI est <http://dbpedia.org/resource/XXXX> où XXXX désigne le sujet de la page. A la page consacrée à Saipal (<https://en.wikipedia.org/wiki/Saipal>) correspond la ressource <http://dbpedia.org/resource/Saipal>.

Question 2: solution 2)

- 2) Cliquez sur ce lien. A quoi correspond le résultat obtenu ?

La page affichée dans le navigateur représente sous une forme lisible par un humain (texte HTML), l'ensemble des triplets dont la ressource <http://dbpedia.org/resource/Saipal> est le sujet.

Lorsque l'objet du triplet est une ressource DBpedia, le lien hypertexte permet de naviguer vers la description de celle-ci, en d'autres termes de parcourir le graphe RDF.

Question 2: solution 2)



[Browse using](#) [Formats](#)

[Faceted Browser](#) [Sparql Endpoint](#)

About: [Saipal](#)

An Entity of Type: [lieu](#), from Named Graph: <http://dbpedia.org>, within Data Space: [dbpedia.org](#)

Saipal is a mountain in the Himalayas of north-west Bajhang district in Nepal. Api, Nampa and Saipal are a trio of high mountains located in northwestern Nepal. Together they form a small range of sharp, icy peaks, rising from a long, steep, snowy crest.

Property	Value
dbo:abstract	<ul style="list-style-type: none">Saipal is a mountain in the Himalayas of north-west Bajhang district in Nepal. Api, Nampa and Saipal are a trio of high mountains located in northwestern Nepal. Together they form a small range of sharp, icy peaks, rising from a long, steep, snowy crest. (en)
dbo:elevation	<ul style="list-style-type: none">7031.000000 (xsd:double)
dbo:firstAscentYear	<ul style="list-style-type: none">1960-01-01 (xsd:gYear)
dbo:locatedInArea	<ul style="list-style-type: none">dbr:Sudurpashchim_Provincedbr:Nepal
dbo:prominence	<ul style="list-style-type: none">1824.000000 (xsd:double)
dbo:wikiPageID	<ul style="list-style-type: none">12850731 (xsd:integer)
dbo:wikiPageLength	<ul style="list-style-type: none">1476 (xsd:nonNegativeInteger)

Question 2: solution 3)

- 3) Le lien était bien <https://dbpedia.org/resource/Saipal> (vous pouvez le vérifier en tapant directement cette url dans la barre d'adresse de votre navigateur), pourtant l'url de la page affichée est <https://dbpedia.org/page/Saipal>.
 - Que s'est-il passé ? Pouvez-vous expliquer ce comportement de votre navigateur ?

L'URL affiché dans la barre de navigation est :

<http://dbpedia.org/page/Saipal>

Dans l'URI de la ressource, resource a été remplacé par page. Elle diffère de celle de la ressource car il y a eu négociation de contenu entre le navigateur et le serveur web de dbpedia. Le navigateur a envoyé un en-tête HTTP Accept-Content : text/html. Le serveur a répondu en faisant une redirection vers l'URL <http://dbpedia.org/page/Saipal> pour obtenir une page HTML décrivant la ressource.

Question 2: solution 3)

- Que faudrait-il modifier dans la requête HTTP pour obtenir en RDF les données de DBpedia concernant Saipal ?

Pour récupérer une description en Turtle il suffit de faire une requête HTTP de type GET avec l'url de la ressource

(http://dbpedia.org/resource/Grenoble_Alpes_University) mais en précisant dans l'en tête Accept le type MIME text/turtle.

- Récupérez ces données au format Turtle. Expliquez comment vous avez procédé.

Pour récupérer la réponse en Turtle, plusieurs moyens sont possibles :

Utiliser l'onglet Network des outils de développement du navigateur et renvoyer la requête en modifiant l'en-tête (header) Accept

Utiliser un outil ligne de commande wget ou curl

Question 3: solution

- Donnez une requête SPARQL qui permet d'obtenir les URIs et l'altitude des dix plus hauts sommets du Népal.

EN SPARQL les graph patterns d'une requête SELECT génèrent une collection non ordonnée de solutions (solution sequence). Cette séquence peut être modifiée en utilisant les modificateurs de séquence définis dans le langage SPARQL. Pour savoir quels modificateurs sont disponibles et comment les utiliser, reportez vous à la documentation de référence de SPARQL.

Pour ordonner les résultats par ordre décroissant on utilise le modificateur ORDER BY et le modificateur LIMIT pour restreindre à 10 le nombre de solutions.

```
select ?m ?a where {  
  ?m rdf:type dbo:Mountain;  
        dbo:locatedInArea dbr:Nepal;  
        dbo:elevation ?a.  
} ORDER by DESC(?a) LIMIT 10
```

Question 3: solution

< SPARQL | HTML5 table

m	a
http://dbpedia.org/resource/Kangchenjunga	8586.0
http://dbpedia.org/resource/Lhotse	8516.0
http://dbpedia.org/resource/Makalu	8481.0
http://dbpedia.org/resource/Lhotse_Middle	8410.0
http://dbpedia.org/resource/Lhotse_Shar	8383.0
http://dbpedia.org/resource/Cho_Oyu	8188.0
http://dbpedia.org/resource/Dhaulagiri	8167.0
http://dbpedia.org/resource/Manaslu	8163.0
http://dbpedia.org/resource/Annapurna	8091.0
http://dbpedia.org/resource/Gyachung_Kang	7952.0

Question 4

- Complétez la requête précédente pour obtenir en plus de l'altitude, le nom de chaque sommet défini par la valeur de la propriété rdfs:label.
- Qu'observez-vous ? Combien de sommets différents obtenez-vous ? Pourquoi ?
- Modifiez la requête pour faire en sorte de n'obtenir que les 10 plus hauts sommets du Népal avec uniquement leur nom en anglais.

Question 4 : solution

- Complétez la requête précédente pour obtenir en plus de l'altitude, le nom de chaque sommet défini par la valeur de la propriété rdfs:label.

Pour obtenir les labels des sommets, il faut dans la requête précédente rejouter le pattern `?m rdfs:label ?l`. Ce qui donne la requête ci-dessous :

```
select ?m ?a ?l where {
  ?m rdf:type dbo:Mountain;
  dbo:locatedInArea dbr:Nepal;
  dbo:elevation ?a;
  rdfs:label ?l.
} ORDER by DESC(?a) LIMIT 10
```

Question 4 : solution

SPARQL | HTML5 table

m	a	i
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@en
http://dbpedia.org/resource/Kangchenjunga	8586.0	"كانغشينجونغا"@ar
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kanchenjunga"@ca
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kančendženga"@cs
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchendzönga"@de
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Κανγκτοενγκιούνγκα"@el
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@eo
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kanchenjunga"@eu
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kanchenjunga"@es
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@fr

Question 4 : solution

- Qu'observez-vous ? Combien de sommets différents obtenez-vous ? Pourquoi ?

Cependant on n'obtient qu'un sommet : le Kangchenjuncha

En effet pour chaque sommet plusieurs labels sont disponibles dans des langues différentes, un même sommet apparaît donc dans autant de binding qu'il a de labels différents.

Question 4 : solution

- Modifiez la requête pour faire en sorte de n'obtenir que les 10 plus hauts sommets du Népal avec uniquement leur nom en anglais.

Un filtre est une clause que l'on peut insérer dans une requête SPARQL pour, comme son nom l'indique, filtrer les résultats. Dans un filtre une expression booléenne est définie et seuls les résultats pour lesquels l'expression est vraie sont conservés. Hint : fonctions filtres sur les chaînes.

```
select ?m ?a ?l where {
  ?m rdf:type dbo:Mountain;
  dbo:locatedInArea dbr:Nepal;
  dbo:elevation ?a;
  rdfs:label ?l.
  FILTER langMatches( lang(?l) , "EN" )
} ORDER by DESC(?a) LIMIT 10
```

Web Sémantique – Ontologies © Ph. Genoud, D. Ziebelin

Question 4 : solution

SPARQL | HTML5 table

m	a	l
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@en
http://dbpedia.org/resource/Lhotse	8516.0	"Lhotse"@en
http://dbpedia.org/resource/Makalu	8481.0	"Makalu"@en
http://dbpedia.org/resource/Lhotse_Middle	8410.0	"Lhotse Middle"@en
http://dbpedia.org/resource/Lhotse_Shar	8383.0	"Lhotse Shar"@en
http://dbpedia.org/resource/Cho_Oyu	8188.0	"Cho Oyu"@en
http://dbpedia.org/resource/Dhaulagiri	8167.0	"Dhaulagiri"@en
http://dbpedia.org/resource/Manaslu	8163.0	"Manaslu"@en
http://dbpedia.org/resource/Annapurna	8091.0	"Annapurna"@en
http://dbpedia.org/resource/Gyachung_Kang	7952.0	"Gyachung Kang"@en

Question 5 :

- La requête précédente donne-t-elle tous les sommets du Népal de plus de 8000m ?
- a) Modifiez la requête de façon à obtenir dans l'ordre du plus haut au plus bas, l'altitude et le nom nom anglais de tous les sommets népalais de plus de 8000m .
- b) Combien de sommets trouvez vous ? Ecrivez un requête qui vous donne directement ce résultat (le nombre de sommets népalais de plus de 8000 m).

Question 5 : solution 1)

- La requête précédente donne-t-elle tous les sommets du Népal de plus de 8000m ?
1) Modifiez la requête de façon à obtenir dans l'ordre du plus haut au plus bas, l'altitude et le nom anglais de tous les sommets népalais de plus de 8000m .

Pour obtenir tous les sommets de plus de 8000 (qui sont neuf), il faut dans la requête précédente rejouter un filtre sur l'altitude et supprimer le modifieur limitant aux 10 premiers résultats. Ce qui donne la requête ci-dessous :

```
select ?m ?a ?l where {  
  ?m rdf:type dbo:Mountain;  
  dbo:locatedInArea dbr:Nepal;  
  dbo:elevation ?a;  
  rdfs:label ?l.  
  FILTER langMatches( lang(?l) , "EN" )  
  FILTER (?a >= 8000)  
} ORDER by DESC(?a)
```

Question 5 : solution 1)

SPARQL | HTML5 table

m	a	i
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@en
http://dbpedia.org/resource/Lhotse	8516.0	"Lhotse"@en
http://dbpedia.org/resource/Makalu	8481.0	"Makalu"@en
http://dbpedia.org/resource/Lhotse_Middle	8410.0	"Lhotse Middle"@en
http://dbpedia.org/resource/Lhotse_Shар	8383.0	"Lhotse Shar"@en
http://dbpedia.org/resource/Cho_Oyu	8188.0	"Cho Oyu"@en
http://dbpedia.org/resource/Dhaulagiri	8167.0	"Dhaulagiri"@en
http://dbpedia.org/resource/Manaslu	8163.0	"Manaslu"@en
http://dbpedia.org/resource/Annapurna	8091.0	"Annapurna"@en

Question 5 : solution 2)

b) Combien de sommets trouvez vous ? Ecrivez un requête qui vous donne directement ce résultat (le nombre de sommets népalais de plus de 8000 m).

hint: utilisez la fonction d'aggrégation COUNT.

Pour obtenir le nombre de sommets de plus de 8000 (qui sont neuf), il faut utiliser le filtre COUNT de la manière suivante :

```
select (COUNT(?m) AS ?nb8000) where {
    ?m rdf:type dbo:Mountain;
        dbo:locatedInArea dbr:Nepal;
        dbo:elevation ?a.
    FILTER (?a >= 8000)
}
```

SPARQL HTML5 table	
nb8000	
9	

Question 6

- Si tous les sommets népalais de plus de 8000 m ont un nom en anglais, DBpedia ne contient le nom en russe que de certains d'entre eux.
- Pouvez-vous dire combien et lesquels ? Indiquez la ou les requêtes qui vous a permis de répondre à ces questions.
- Ecrivez une requête qui pour ces sommets donne simultanément leur URI, leur altitude, leur nom en anglais et leur nom en russe et les trie selon leur altitude (l'ordre décroissant).
- Inversement donnez une requête qui donne la liste des sommets de plus de 8000 qui n'ont pas de nom (label) en russe
- Donnez une requête qui donne la liste des tous les sommets de plus de 8000 avec leur altitude, leur nom en anglais et **si il existe** leur nom en russe.

Question 6 : solution

- Si tous les sommets népalais de plus de 8000m ont un nom en anglais, DBpedia ne contient le nom en russe que de certains d'entre eux.
- Pouvez-vous dire combien et lesquels ? Indiquez la ou les requêtes qui vous a permis de répondre à ces questions.

Pour obtenir le nombre de sommets de plus de 8000 du Népal dont DBpedia connaît le nom en russe, on peut utiliser la requête suivante :

```
SELECT (COUNT(?m) AS ?nb8000) WHERE {
```

```
?m rdf:type dbo:Mountain;
  dbo:locatedInArea dbr:Nepal;
  dbo:elevation ?a;
  rdfs:label ?l
  FILTER langMatches( lang(?l) , "RU" )
  FILTER (?a >= 8000)
```

```
}
```

SPARQL HTML5 table	
nb8000	
5	

Question 6 : solution

- Pouvez-vous dire lesquels ?

Pour connaître quels sont ces sommets on pourrait utiliser la requête suivante :

```
SELECT ?m ?a ?l WHERE {
  ?m rdf:type dbo:Mountain;
    dbo:locatedInArea dbr:Nepal;
    dbo:elevation ?a;
    rdfs:label ?l
  FILTER langMatches( lang(?l) , "RU" )
  FILTER (?a >= 8000)
}
```

Question 6 : solution

SPARQL | HTML5 table

m	a	l
http://dbpedia.org/resource/Lhotse	8516.0	"Лхонзе"@ru
http://dbpedia.org/resource/Makalu	8481.0	"Макалу"@ru
http://dbpedia.org/resource/Manaslu	8163.0	"Манаслу"@ru
http://dbpedia.org/resource/Cho_Oyu	8188.0	"Чо-Ойю"@ru
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Канченджанга"@ru

Question 6 : solution

- Ecrivez une requête qui pour ces sommets donne simultanément leur URI, leur altitude, leur nom en anglais et leur nom en russe et les trie selon leur altitude (l'ordre décroissant).

Pour obtenir les noms anglais et russe des sommets de plus de 8000 du Népal dont DBpedia connaît le nom en russe, on peut utiliser la requête suivante :

```
SELECT ?m ?a ?l1 ?l2 WHERE {  
  ?m rdf:type dbo:Mountain;  
  dbo:locatedInArea dbr:Nepal;  
  dbo:elevation ?a;  
  rdfs:label ?l1, ?l2 .  
  FILTER (langMatches( lang(?l1) , "EN" ) &&  
 langMatches( lang(?l2) , "RU" ))  
  FILTER (?a >= 8000)  
} ORDER BY DESC(?a)
```

Question 6 : solution

SPARQL | HTML5 table

m	a	l1	l2
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@en	"Канченджанга"@ru
http://dbpedia.org/resource/Lhotse	8516.0	"Lhotse"@en	"Лхоцзе"@ru
http://dbpedia.org/resource/Makalu	8481.0	"Makalu"@en	"Макалу"@ru
http://dbpedia.org/resource/Cho_Oyu	8188.0	"Cho Oyu"@en	"Чо-Ойю"@ru
http://dbpedia.org/resource/Manaslu	8163.0	"Manaslu"@en	"Манаслу"@ru

Question 6 : solution

- Inversement donnez une requête qui donne la liste des sommets de plus de 8000 qui n'ont pas de nom (label) en russe

Utilisez la négation et la possibilité d'effectuer un filtrage en testant l'absence d'un pattern. Negation dans SPARQL.

Pour obtenir les noms anglais des sommets de plus de 8000 du Népal pour lesquels DBpedia n'a pas de nom en russe, on peut utiliser la requête suivante :

```
SELECT ?m ?a ?l1 WHERE {
  ?m rdf:type dbo:Mountain;
    dbo:locatedInArea dbr:Nepal;
    dbo:elevation ?a;
    rdfs:label ?l1 .
  FILTER (langMatches( lang(?l1) , "EN" ))
  FILTER (NOT EXISTS { ?m rdfs:label ?l2. FILTER
  (langMatches( lang(?l2) , "RU" )) })
  FILTER (?a >= 8000)
} ORDER BY DESC(?a)
```

Question 6 : solution

SPARQL | HTML5 table

m	a	l1
http://dbpedia.org/resource/Lhotse_Middle	8410.0	"Lhotse Middle"@en
http://dbpedia.org/resource/Lhotse_Shar	8383.0	"Lhotse Shar"@en
http://dbpedia.org/resource/Dhaulagiri	8167.0	"Dhaulagiri"@en
http://dbpedia.org/resource/Annapurna	8091.0	"Annapurna"@en

Question 6 : solution

- Donnez une requête qui donne la liste des tous les sommets de plus de 8000 avec leur altitude, leur nom en anglais et si il existe leur nom en russe.

Pour obtenir les noms anglais et éventuellement russes des sommets de plus de 8000 du Népal on peut utiliser la requête suivante :

```
SELECT ?m ?a ?l1 ?l2 WHERE {
  ?m rdf:type dbo:Mountain;
    dbo:locatedInArea dbr:Nepal;
    dbo:elevation ?a;
    rdfs:label ?l1 .
  FILTER (?a >= 8000)
  FILTER (langMatches( lang(?l1) , "EN" ))
  OPTIONAL {
    ?m rdfs:label ?l2.
    FILTER (langMatches( lang(?l2) , "RU" ))
  }
} ORDER BY DESC(?a)
```

Question 6 : solution

SPARQL | HTML5 table

m	a	l1	l2
http://dbpedia.org/resource/Kangchenjunga	8586.0	"Kangchenjunga"@en	"Кангченджанга"@ru
http://dbpedia.org/resource/Lhotse	8516.0	"Lhotse"@en	"Лхотзе"@ru
http://dbpedia.org/resource/Makalu	8481.0	"Makalu"@en	"Макалу"@ru
http://dbpedia.org/resource/Lhotse_Middle	8410.0	"Lhotse Middle"@en	
http://dbpedia.org/resource/Lhotse_Shар	8383.0	"Lhotse Shar"@en	
http://dbpedia.org/resource/Cho_Oyu	8188.0	"Cho Oyu"@en	"Чо-Ойю"@ru
http://dbpedia.org/resource/Dhaulagiri	8167.0	"Dhaulagiri"@en	
http://dbpedia.org/resource/Manaslu	8163.0	"Manaslu"@en	"Манаслу"@ru
http://dbpedia.org/resource/Annapurna	8091.0	"Annapurna"@en	

Question 7

- Donnez une requête qui vous permet de trouver les sommets dont la première ascension a été effectuée par Lionel Terray, l'année de cette première ascension et si DBpedia dispose des données le nom du ou des alpinistes avec qui Lionel Terray a accompli cette première.

Vous devriez obtenir un résultat ressemblant à celui obtenu dans l'image ci-dessous

SPARQL | HTML5 table

m	nom	altitude	pnom	année
http://dbpedia.org/resource/Makalu	"Makalu"@en	8481.0	"Jean Couzy"@en	"1955"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Chomo_Lonzo	"Chomo Lonzo"@en	7804.0	"Jean Couzy"@en	"1954"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Kumbhakarna_Mountain	"Kumbhakarna Mountain"@en	7710.0		"0027"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Huantsán	"Huantsán"@en	6369.0		"1952"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Chacraraju	"Chacraraju"@en	6108.0		"1956"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Fitz_Roy	"Fitz Roy"@en	3405.0		"1952"^^< http://www.w3.org/2001/XMLSchema#gYear >

Question 7 : solution

```
SELECT ?m ?nom ?altitude ?pnom ?année WHERE {  
?m rdf:type dbo:Mountain;  
  dbo:firstAscentPerson dbr:Lionel_Terray;  
  dbo:elevation ?altitude ;  
  rdfs:label ?nom.  
  FILTER langMatches( lang(?nom), "EN" )  
  OPTIONAL {  
    ?m dbo:firstAscentPerson ?p.  
    ?p rdfs:label ?pnom.  
    FILTER langMatches( lang(?pnom), "EN" )  
    FILTER (dbr:Lionel_Terray != ?p)  
  }  
  OPTIONAL {  
    ?m dbo:firstAscentYear ?année.  
  }  
} ORDER BY DESC (?altitude)
```

Question 7 a): solution

SPARQL | HTML5 table

m	nom	altitude	pnom	année
http://dbpedia.org/resource/Makalu	"Makalu"@en	8481.0	"Jean Couzy"@en	"1955"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Chomo_Lonzo	"Chomo Lonzo"@en	7804.0	"Jean Couzy"@en	"1954"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Kumbhakarna_Mountain	"Kumbhakarna Mountain"@en	7710.0		"0027"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Huantsán	"Huantsán"@en	6369.0		"1952"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Chacraraju	"Chacraraju"@en	6108.0		"1956"^^< http://www.w3.org/2001/XMLSchema#gYear >
http://dbpedia.org/resource/Fitz_Roy	"Fitz Roy"@en	3105.0		"1952"^^< http://www.w3.org/2001/XMLSchema#gYear >

Question 7 b):

Donnez une requête qui vous permet de trouver les sommets dont la première ascension a été effectuée par Lionel Terray, l'année de cette première ascension et si DBpedia dispose des données le nom **et le pays de naissance** du ou des alpinistes avec qui Lionel Terray a accompli cette première.

sommet	annee	compagnonCordee	pays
"Makalu"@en	"1955"^^< http://www.w3.org/2001/XMLSchema#gYear >	"Jean Couzy"@en	"France"@en
"Chomo Lonzo"@en	"1954"^^< http://www.w3.org/2001/XMLSchema#gYear >	"Jean Couzy"@en	"France"@en
"Huantsán"@en	"1952"^^< http://www.w3.org/2001/XMLSchema#gYear >		

Question 7

```
SELECT ?m ?nom ?altitude ?année ?pnom ?nomPays
WHERE {
  ?m rdf:type dbo:Mountain;
    dbo:firstAscentPerson dbr:Lionel_Terray;
    dbo:elevation ?altitude ;
    rdfs:label ?nom.
    FILTER langMatches( lang(?nom), "EN" )
  OPTIONAL {
    ?m dbo:firstAscentPerson ?p.
    ?p rdfs:label ?pnom;
      dbo:birthPlace ?lieuNaissance.
    ?lieuNaissance dbo:country ?pays.
    ?pays rdfs:label ?nomPays.
    FILTER (dbr:Lionel_Terray != ?p)
    FILTER langMatches( lang(?pnom), "EN" )
    FILTER langMatches( lang(?nomPays), "EN" )
  }
  OPTIONAL {
    ?m dbo:firstAscentYear ?année.
  }
} ORDER BY DESC(?altitude )
```