

# TD Web sémantique 4-IF-WS (2016–2017)

Aimene Belfodil, Sylvie Calabretto, Mehdi Kaytoue

## 1 Découverte

On considère la page Web suivante, qui décrit un épisode d'une série TV sur le site IMDB : <http://www.imdb.com/title/tt3811812/>

- On utilisera le validateur RDFa <http://www.w3.org/2012/pyRdfa/Validator.html> et microdata <http://www.w3.org/2012/pyMicrodata/> afin de voir si IMDB est, selon le W3C, un bon élève : trouvez vous des données ? Dans quel format ? pouvez vous les ré utiliser ?
- On testera avec des pages représentant des médias d'autres sites comme *youtube*, *books.google.com*, *expedia*, *booking.com*, etc.
- On comprendra les vocabulaires via <http://prefix.cc/> et <http://schema.org/>
- On "jouera" avec <https://rdfa.info/play/> pour comprendre comment intégrer du RDF dans du code HTML, et on créera une page Web qui nous décrit pour la tester. Les moteurs de recherche évolués liront ce code pendant leur phase d'indexation !
- Une autre solution consiste à annoter une page Web existante avec des outils comme l'outil d'aide au balisage de Google ([www.google.com/webmasters/markup-helper/](http://www.google.com/webmasters/markup-helper/))

## 2 SPARQL

### 2.1 Syntaxes

On considère le jeu de données *etudiants.n3*. Ouvrir et comprendre ce fichier. Dessiner le graphe RDF correspondant. Convertir dans d'autres syntaxes (RDF/XML, RDF/JSON, triples, ...) avec des outils du Web donnés en cours ([www.w3.org/RDF/Validator/](http://www.w3.org/RDF/Validator/), [www.easyrdf.org/converter](http://www.easyrdf.org/converter), ...).

### 2.2 Trouver les réponses à la requête

On considère le vocabulaire *insa* avec les prédicats suivants et leur signification intuitive :

- <http://insa-lyon.fr#inscrit> le sujet est inscrit à l'objet
- <http://insa-lyon.fr#binome> le sujet a comme binôme l'objet
- <http://insa-lyon.fr#nom> le sujet a pour nom l'objet
- <http://insa-lyon.fr#formation> le sujet est dans la formation objet
- <http://insa-lyon.fr#departement> le sujet est dans le département objet

Pour chaque requête, trouver la(les) réponse(s) possible(s). On dessinera le graphe requête que l'on cherchera à appareiller avec le graphe données.

- |   |  |
|---|--|
| — PREFIX insa: < <a href="http://insa-lyon.fr/insa#">http://insa-lyon.fr/insa#</a> ><br>PREFIX ue: < <a href="http://insa-lyon.fr/ue#">http://insa-lyon.fr/ue#</a> ><br>SELECT * WHERE<br>{<br>?e insa:inscrit ue:alia .<br>}                           | — PREFIX insa: < <a href="http://insa-lyon.fr/insa#">http://insa-lyon.fr/insa#</a> ><br>PREFIX ue: < <a href="http://insa-lyon.fr/ue#">http://insa-lyon.fr/ue#</a> ><br>SELECT ?e WHERE {<br>{ { ?e insa:binome ?e2 . }<br>UNION<br>{ ?e2 insa:binome ?e . } }<br>?e2 insa:inscrit ue:alia .<br>}  |
| — PREFIX insa: < <a href="http://insa-lyon.fr/insa#">http://insa-lyon.fr/insa#</a> ><br>PREFIX ue: < <a href="http://insa-lyon.fr/ue#">http://insa-lyon.fr/ue#</a> ><br>SELECT ?e WHERE<br>{<br>?e insa:binome ?e2 .<br>?e2 insa:inscrit ue:alia .<br>} | — PREFIX insa: < <a href="http://insa-lyon.fr/insa#">http://insa-lyon.fr/insa#</a> ><br>PREFIX ue: < <a href="http://insa-lyon.fr/ue#">http://insa-lyon.fr/ue#</a> ><br>PREFIX foaf: < <a href="http://xmlns.com/foaf/spec/">http://xmlns.com/foaf/spec/</a> ><br>SELECT ?p ?e2 WHERE {<br>?e ?p ?e2; foaf:name ?n .<br>FILTER(?n = "Basile")<br>} |

## 2.3 Trouver la requête

Écrire en SPARQL les requêtes suivantes :

- Quel(s) étudiants ont Alice pour binôme ?
- Quels étudiants sont inscrits à une UE de la formation <http://insa-lyon.fr/formation#4if> ?
- Quels sont les étudiants en binôme inscrit à la même UE ?
- Quelles sont les UE d'une même formation qui ont des départements différents ?
- Quels étudiants partagent un même binôme ? Même question si on considère qu'il faut rendre symétrique la relation binôme.

## 2.4 Utilisation de Twinkle

Twinkle fournit une interface simple et un moteur SPARQL. Il permet de charger un fichier RDF ou encore de se connecter à un point d'accès comme celui de DBPEDIA. Tester les requêtes des deux questions précédentes et vérifier vos réponses !

# 3 DBpedia

## 3.1 Exploration

- Chercher des personnes, des pays, des villes, ... Que peut-on dire de ces concepts ? Les descriptions sont-elles toujours cohérentes ?
- Pour cela, taper un URI dans le navigateur Web, et naviguer.
- On pourra aussi chercher des vocabulaires <http://prefix.cc/>

## 3.2 Requêtes

1. Quels sont les pays de plus de 15 millions d'habitants dont on peut avoir le nom en anglais ?
2. Quels sont les pays qui ont 'Republic' dans leur nom anglais et établit avant 1920 ?
3. Quels sont les pays qui utilisent l'euro et où l'on parle français ? Et de moins de 5000 habitants ?
4. Le fleuve Amazone est-il plus grand que le Nil ?
5. Donner le film qui a le second revenu le plus élevé donné en dollars  
(`<HTTP ://DBPEDIA.ORG/DATATYPE/USDOLLAR>` )
6. Donner pour chaque dirigé (director) par Steven Spielberg la liste des acteurs et producteurs si possible.  
(on peut utiliser un `GROUP BY` et la fonction d'agrégation `GROUP_CONCAT(...)`. On filtrera le résultat pour ne garder que les 5 films les plus récents d'une part, les plus anciens d'autre part.
7. Lister tous les couples de noms de présidents français tels que le second est un successeur (pas forcément direct) du premier. Pour chaque couple, on donnera la distance du chemin qui les sépare (`DBPPROP :SUCCESSOR+ , DBPPROP :SUCCESSOR*`)

## Divers

- TWINKLE : outil graphique multi-fonctions <http://www.ldodds.com/projects/twinkle/>
- JENA : Outil multi-fonctions <http://jena.apache.org/download>
- Tutorial SPARQL <http://jena.apache.org/tutorials/sparql.html>
- DBpedia access point <http://dbpedia.org/snorql/>
- DBpedia access point <http://dbpedia.org/sparql/>