

# TP : Écriture de requêtes SPARQL sur DBpedia puis sur un graphe RDF local avec Apache Jena

Hala Skaf-Molli  
Master 1 Informatique  
2025-2026

## Objectifs

- Apprendre à **formuler** des requêtes SPARQL correctes.
- Interroger un **endpoint public** (DBpedia) puis un **endpoint local** (Fuseki).
- Comprendre la fédération de requêtes entre données ouvertes et graphe privé.

## Étape A : Requêtes sur DBpedia

Vous allez utiliser le serveur (endpoint) SPARQL de DBpedia. : <https://dbpedia.org/sparql>. Vous pouvez utiliser directement les préfixes dbr, dbo, rdf, etc. : de nombreux préfixes sont déjà définis sur DBpedia. Pour écrire une requête, il est utile de connaître les informations disponibles pour une ressource. Les URI des ressources sont construites selon le modèle :

<http://dbpedia.org/resource/Nantes>  
[http://dbpedia.org/resource/Christine\\_and\\_the\\_Queens](http://dbpedia.org/resource/Christine_and_the_Queens)

Il est préférable d'écrire vos requêtes SPARQL dans un fichier, puis exécuter les exécuter sur l'endpoint SPARQL de DBpedia.

Essayez la requête SPARQL suivante :

```
SELECT DISTINCT ?p
WHERE {
  dbr:Nantes ?p ?o
}
```

Cette requête affiche toutes les propriétés pour lesquelles Nantes est le sujet. **Écrivez et testez vos propres requêtes** pour répondre aux questions suivantes :

1. Trouver le nom et la date de naissance de 10 écrivains (classe `dbo:Writer`).
2. Lister les écrivains **nés en France**.
3. Obtenir les 5 villes françaises les plus peuplées (nom de la ville et population).
4. Afficher la capitale et la langue officielle de trois pays de votre choix.

## Étape B : Mise en place d'Apache Jena Fuseki

**Apache Jena** est un framework open source (Java) pour le Web sémantique. Il propose :

- **ARQ** : moteur d'exécution SPARQL 1.1 (SELECT, CONSTRUCT, ASK, DESCRIBE, UPDATE).
- **TDB2** : base de stockage de triplets RDF sur disque (persistance).

- **Fuseki** : serveur HTTP qui expose un *endpoint* SPARQL et une interface Web.
- 1. Téléchargez Apache Jena Fuseki <https://jena.apache.org/download/> (The binary distribution of the Fuseki server).
- 2. Décompressez l'archive et lancez le serveur :  
`./fuseki-server`
- 3. Accédez à <http://localhost:3030/>, créez un dataset nommé **authors**, puis importez le fichier **authors.ttl** fourni.

## Étape C : Requêtes locales

À l'aide de l'interface SPARQL de Fuseki, **récrivez vos propres requêtes** pour :

1. Lister tous les auteurs et leur date de naissance, triés par date croissante.
2. Trouver les œuvres publiées avant 1900.
3. Compter le nombre d'œuvres par auteur.
4. Créer une requête **CONSTRUCT** qui produit un graphe des seuls livres de Victor Hugo.

## Étape D : Requêtes fédérées

Chaque auteur possède un lien **schema:sameAs** vers DBpedia.

**Écrivez une requête fédérée** qui, pour chaque auteur de votre graphe local, récupère depuis DBpedia :

- sa ville de naissance (**dbo:birthPlace**),
- et sa nationalité (**dbo:birthPlace/dbo:country** ou équivalent).

## Livrables déposer sur madoc

- Un fichier **requetes.rq** contenant **toutes vos requêtes SPARQL**.