



## > ANATOMIE D'UN ÉLÉMENT WIKIDATA — WIKIDATA.ORG

**libellé** — Douglas Adams (Q42) — **identifiant de l'élément**

**description** — écrivain anglais de science-fiction  
Douglas Noel Adams | Douglas Noël Adams — **alias**

**propriété** — **scolarité** — **valeur**

**rang** — **St John's College** — **qualificatifs**

**groupe de déclarations** — **références dépliées**

**références repliées**

**Déclarations**

**St John's College**

- date de fin: 1974
- principale spécialité d'étude: littérature anglaise
- diplôme: baccalauréat en arts
- date de début: 1971

**2 références**

**références dépliées**

- affirmé dans: Encyclopædia Britannica Online
- URL de la référence: <http://www.ndb.com/people/731/000023662/>
- langue originale: anglais
- date de consultation: 7 décembre 2013
- éditeur: NNDB
- titre: Douglas Adams (anglais)

**références repliées**

**Brentwood School**

- date de fin: 1970
- date de début: 1959

**0 références**

**Wikidata** est devenue l'infrastructure de base pour l'utilisation, le partage et la conservation collaborative des connaissances de références structurées. Dépôt central de données structurées comportant des millions de données statistiques et de données factuelles librement réutilisables (CC0) et contribuant au fonctionnement de milliers de sites internet et services.

### Type d'entités

- Élément
- Propriété
- Lexème
- Sens défini
- Forme lexicographique

### Type de données

- Chaîne
- Texte monolingue
- Identifiant externe
- URL
- Expression mathématique\*
- Notation musicale
- Quantité
- Date
- Coordonnées
- Fichier de Commons
- Forme géographique de Commons
- Donnée tabulaire de Commons

\* non requêttable

## > INTERFACE DE REQUÊTE SPARQL DES DONNÉES RDF — QUERY.WIKIDATA.ORG

Wikidata Query Service

Exemples Aide Davantage d'outils Constructeur de requête français

**zone de saisie de la requête**

```

1 #Chats
2 SELECT ?item ?itemLabel ?occupationLabel
3 WHERE
4 {
5   ?item wdt:P31 wd:Q146. # Ce doit être un chat
6   ?item wdt:P106 ?occupation. # Occupation du chat
7   SERVICE wikibase:label { bd:serviceParam wikibase:language "mul,fr". }
8 }

```

**interface de saisie de la requête**

**lancement et arrêt de la requête**

**nombre de résultats et temps d'exécution (60 000 ms max.)**

**type d'export**

**mode d'affichage des résultats (tableau, graphique, carte, etc.)**

**56 résultats en 778 ms**

**Code** **Télécharger** **Lien**

**résultats de la requête**

item	itemLabel	occupationLabel
<a href="#">wd:Q677525</a>	Orangey	acteur ou actrice
<a href="#">wd:Q893453</a>	Sam l'insubmersible	chat de navire
<a href="#">wd:Q1050083</a>	Catmando	personnalité politique

**SPARQL (SPARQL Protocol and RDF Query Language)** est un langage de requête standardisé du W3C. À l'aide d'un service de requête, il est possible d'interroger la base de connaissance Wikidata et la médiathèque Wikimedia Commons. Le point de terminaison (*endpoint*) SPARQL est accessible publiquement et dispose d'un client web SPARQL :

- Wikidata : [query.wikidata.org](http://query.wikidata.org)
- Wikimedia Commons : [commons-query.wikimedia.org](http://commons-query.wikimedia.org)
- Clients alternatifs : [qler.cs.uni-freiburg.de](http://qler.cs.uni-freiburg.de), [try.orbograph.com](http://try.orbograph.com), [wikidata.demo.openlinksw.com/sparql](http://wikidata.demo.openlinksw.com/sparql)

### Raccourcis clavier

- Auto-complétion : Ctrl+Espace
- Lancer la requête : Ctrl+Entrée
- Arrêter la requête : Ctrl+Échap

### Coloration syntaxique des requêtes

- **rouge** : commandes SPARQL
- **vert** : variables (débutent par ?)
- **bleu** : préfixes (wd, wdt, etc.)
- **marron** : chaînes de caractères
- **gris** : commentaires





# > GRAPHE DE TRIPLETS RDF



Wikidata comporte plus de 12000 propriétés (prédicats) tels que père (P22) et plus de 100 millions d'éléments (sujets) tels que Bach (Q1339) et l'un de ses fils (Q107277).

# > ANATOMIE D'UNE REQUÊTE SPARQL



Commande	Description	Exemple	
1 #title:	Titre de la requête.	#title:Titre du tableau	OPTIONNEL
#defaultView:	Type d'affichage des résultats : Graph, Dimensions, Timeline, Tree, TreeMap, BubbleChart, ImageGrid, ScatterChart, BarChart, AreaChart, LineChart, Map.	#defaultView:BarChart #defaultView:Map{"layer": "?typeLabel"}	
3 PREFIX	Abrège les adresses IRI. La plupart des préfixes courants sont pris en charge par le moteur sans qu'il soit nécessaire de les spécifier.	PREFIX bnf-onto: <http://data.bnf.fr/ontology/bnf-onto/>	
4 SELECT	Type de requête SPARQL le plus courant permettant de définir les colonnes du tableau de solutions attendu en retour.	SELECT ?humain ?humainLabel ?humainDescription ?humainAltLabel	OBLIGATOIRE
5 WHERE {...}	Spécifie les critères de recherche énumérés sous forme de séries de motifs de triplets afin de définir les lignes souhaitées.	WHERE {?sujet ?prédicat ?objet.} WHERE {?humain wd:P31 wdt:Q5.}	
6 GROUP BY	Sépare les résultats en plusieurs groupes. Mêmes variables que dans SELECT.	SELECT ?pays ?population GROUP BY ASC ?population ?pays	OPTIONNEL
7 HAVING	Filtre les valeurs agrégées. Requiert GROUP BY.	HAVING (?date>2) HAVING (AVG (?score) >= 10)	
8 ORDER BY	Ordonne les résultats de la requête suivant un ou plusieurs des éléments à afficher.	ORDER BY (?PIB) ORDER BY DESC (?PIB) ORDER BY (?item1) (?item2)	
9 LIMIT	Limite le nombre d'éléments à afficher.	LIMIT 100	
10 OFFSET	Décale les éléments à afficher du nombre indiqué.	OFFSET 10	

# > SYMBOLES



Description	Exemple : les enfants du couple Bach	
.	?enfant wdt:P22 wd:Q1339.	
;	?enfant wdt:P22 wd:Q1339; ?enfant wdt:P25 wd:Q57487.	?enfant wdt:P22 wd:Q1339; wdt:P25 wd:Q57487.
^	?enfant wdt:P22 wd:Q1339; ?enfant wdt:P25 wd:Q57487; ?enfant wdt:P106 wd:Q36834, ?enfant wdt:P106 wd:Q486748.	?enfant wdt:P22 wd:Q1339; wdt:P25 wd:Q57487; wdt:P106 wd:Q36834, wd:Q486748.
[...]	wd:Q1339 wdt:P40 [ wdt:P40 ?petitEnfant ].	
#	#commentaire d'une ligne maximum	

# > EXEMPLES DE REQUÊTES



## Requêtes de base

**Obtenir la liste de tous les chats célèbres qui sont sur Wikidata**

```
SELECT *
WHERE { ?chat wdt:P31 wd:Q146. }
```

**Compter le nombre de chats célèbres**

```
SELECT (COUNT(*) AS ?nombre)
WHERE { ?chat wdt:P31 wd:Q146. }
```

**Afficher les libellés en français ou en anglais**

```
SELECT ?chat ?chatLabel
WHERE {
  ?chat wdt:P31 wd:Q146.
  SERVICE wikibase:label { bd:serviceParam
    wikibase:language "fr,en". }
}
```

**Lister les chats avec ou sans date de décès**

```
SELECT ?chat ?décès
WHERE {
  ?chat wdt:P31 wd:Q146.
  OPTIONAL { ?chat wdt:P570 ?décès. }
}
```

## Regroupement, filtrage et tri

**Afficher les 10 premiers résultats**

```
SELECT ?chat
WHERE { ?chat wdt:P31 wd:Q146. }
LIMIT 10
```

**Lister les noms des chats par ordre anti-alphabétique**

```
SELECT ?chat
WHERE { ?chat wdt:P31 wd:Q146. }
ORDER BY DESC (?chat)
```

**Afficher le domaine d'activité des chats**

```
SELECT ?chat ?chat ?occupation
WHERE {
  ?chat wdt:P31 wd:Q146.
  ?chat wdt:P106 ?occupation.
}
ORDER BY ?occupation
```

**Lister les chats nés avant l'an 2000**

```
SELECT ?chat
WHERE {
  ?chat wdt:P31 wd:Q146.
  ?chat wdt:P569 ?naissance.
  FILTER ( ?naissance <
    "2000-01-01T00:00:00Z"^^xsd:dateTime )
}
```

**Lister des pays où sont nés 5 chats ou plus**

```
SELECT ?pays
WHERE {
  ?chat wdt:P31 wd:Q146.
  ?chat wdt:P19 ?naissance.
  ?naissance wdt:P17 ?pays.
}
GROUP BY ?pays
HAVING (COUNT(?chat) >=5)
```

**Lister tous les chats (« chat domestique » et ses sous-classes)**

```
SELECT ?chat ?chat
WHERE {
  ?chat wdt:P31/wdt:P279* wd:Q146.
}
```

## Affichage des résultats

**Afficher les portraits des chats**

```
#title:Chats, avec des images
#defaultView:ImageGrid
SELECT ?chat ?portrait
WHERE {
  ?chat wdt:P31 wd:Q146;
  wdt:P18 ?portrait.
}
```

**Afficher la carte des lieux de naissance des chats**

```
#defaultView:Map
SELECT ?chat ?coordonnées
WHERE {
  ?chat wdt:P31 wd:Q146;
  wdt:P19 ?naissance.
  ?naissance wdt:P625 ?coordonnées.
}
```

**Insérer une image manquante**

```
OPTIONAL { ?chat wdt:P18 ?img. }
BIND ( IF ( BOUND (?img), ?img,
  <http://commons.wikimedia.org/wiki/Special:FilePath/File:No-image-available.jpg>
) AS ?image )
```



Pierre-Yves Beaudouin  
Novembre 2024  
Dernière version sur <https://www.wiki/AQVR>

Niveau de difficulté :  
Facile  
Difficile





# CLAUSE SELECT



Commande	Description	Exemple
*	Affiche toutes les valeurs présentes dans <b>WHERE</b>	<code>SELECT * WHERE { ... }</code>
<b>DISTINCT</b>	Élimine les doublons	<ul style="list-style-type: none"> <li><code>SELECT DISTINCT ?item</code></li> <li><code>SELECT... (COUNT(DISTINCT ?a) AS ?aa)</code></li> </ul>
<b>COUNT, SUM, AVG, MIN, MAX, SAMPLE, GROUP_CONCAT</b>	Fonctions d'agrégation + <b>GROUP BY</b> pour les autres variables	<ul style="list-style-type: none"> <li><code>SELECT...(AVG(YEAR(?death)) AS ?deathYear)</code></li> <li><code>SELECT...(GROUP_CONCAT(DISTINCT ?occupationLabel; SEPARATOR=";") AS ?occupations)</code></li> </ul>
<b>AS</b>	Renomme une variable	<code>SELECT (MAX(?population) AS ?populationMax)</code>

# CLAUSE WHERE



Commande	Description	Exemple
<b>VALUES</b>	Limite la recherche à certains éléments	<code>VALUES ?item {wd:Q1339 wd:Q1268}.</code>

## Exemples de filtres

<code>FILTER ( ?age &gt; 40 ).</code> <code>FILTER(30 &lt;= ?age &amp;&amp; ?age &lt; 40).</code>	<b>Humains sans illustration</b> <code>FILTER NOT EXISTS { ?humain wdt:P18 ?photo . }</code>
<b>Élément Wikidata avec identifiant BnF de Zola</b> <code>FILTER CONTAINS(?idbnf, "119297200") .</code>	<b>Pas d'article sur la Wikipédia en français</b> <code>FILTER NOT EXISTS { ?article schema:about ?item . ?article schema:isPartOf &lt;https://fr.wikipedia.org/&gt; . }</code>
<b>Exclusion d'éléments spécifiques</b> <code>FILTER ( ?item NOT IN ( wd:Q1,wd:Q2 ) ) .</code>	<b>Propriété P18 non renseignée</b> <code>FILTER ( NOT EXISTS { ?item wdt:P18 [] } )</code>
<b>Vérifie si des résultats commencent par Mr.</b> <code>FILTER ( STRSTARTS ( ?humanLabel, "Mr." ) ) .</code>	<b>Filtre + Expression régulière</b> <code>FILTER ( !REGEX ( STR ( ?item ), "[\\.\q]" ) ) .</code>
<b>Conserve les libellés en anglais</b> <code>FILTER ( LANG ( ?humanLabel ) = "en" ) .</code>	

## Opérateurs de filtre

<code>+, -, *, /</code>	Opérateur arithmétique	<ul style="list-style-type: none"> <li><code>BIND( YEAR(?mort) - YEAR(?né) AS ?age ) .</code></li> <li><code>(AVG(?mort - ?né) AS ?age).</code></li> </ul>
<code>&lt;, &gt;, =, !=, &lt;=, &gt;=</code>	Comparaison	<ul style="list-style-type: none"> <li><code>"abc" &lt; "abd"</code></li> <li><code>"2016-01-01"^^xsd:dateTime &gt; "2015-12-31"^^xsd:dateTime</code></li> <li><code>wd:Q4653 != wd:Q283111</code></li> </ul>
<code>&amp;&amp;,   , !</code>	Opérateur logique (ET, OU, NON)	<ul style="list-style-type: none"> <li><code>FILTER (?age &gt; 10 &amp;&amp; ?age &lt; 100).</code></li> <li><code>FILTER ( ?mort "1969-01-01T00:00:00"^^xsd:dateTime &amp;&amp; ?mort &lt; "1970-01-01T00:00:00"^^xsd:dateTime ) .</code></li> </ul>
<code>xsd:integer</code> <code>xsd:string</code>	Conversion d'une chaîne en entier ou vice-versa	<ul style="list-style-type: none"> <li><code>ORDER BY xsd:integer(?identifiant)</code></li> <li><code>wikibase:timePrecision "9"^^xsd:integer</code></li> </ul>
<code>BOUND</code> <code>isURI(?x)</code> <code>isBlank(?x)</code> <code>isLiteral(?x)</code>	Test	<ul style="list-style-type: none"> <li><code>FILTER ( BOUND ( ?ended ) ) .</code></li> <li><code>BIND ( IF ( BOUND(?rgb),?rgb,"CCCCC") AS ?rgb ) .</code></li> </ul>

# CHEMINS DE PROPRIÉTÉS



/	Séquence (sous-classes)	<code>?a wdt:P31 ?b. ?b wdt:P279 ?c.</code>	<code>= ?a wdt:P31/wdt:P279 ?c.</code>
*	Zéro ou plusieurs propriétés	Bach + enfants : <code>wd:Q1339 wdt:P40* ?enfant.</code>	
+	Une ou plusieurs propriétés	Descendance : <code>wd:Q1339 wdt:P40+ ?enfant.</code>	
?	Zéro ou une propriété	Bach + personnes dont le père est Bach : <code>?enfant wdt:P22 ?parent. wd:Q1339.</code>	
	Chemin alternatif	<code>{?enfant wdt:P22 ?parent. } UNION {?enfant wdt:P25 ?parent. }</code>	<code>?enfant wdt:P22 wdt:P25 ?parent.</code>
(...)	Groupe de chemins	Enfants du coupe : <code>?enfant (wdt:P22 wdt:P25)+ wd:Q1339.</code>	
^	Chemin inverse	De l'objet au sujet : <code>wd:Q1339 wdt:P22/^wdt:P22 ?fratrie.</code>	

# MODÈLE DE DONNÉES RDF

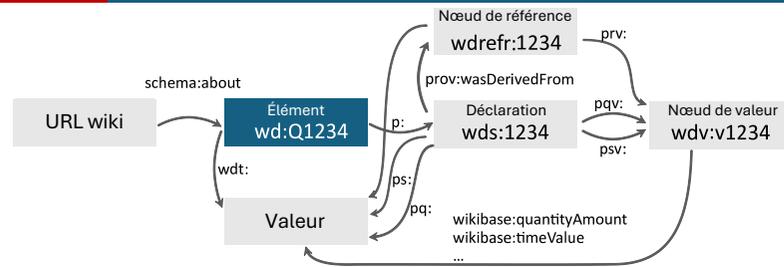


Schéma original par Thomas Shafee, CC-BY-4.0

# COMBINAISON DE GRAPHS



Commande	Description et exemple
A.B.	Réunir les solutions de A et de B en combinant les valeurs de variables communes. <code>?enfant wdt:P22 wd:Q1339. #enfants de Bach</code> <code>?enfant wdt:P21 wd:Q6581072. #de sexe féminin</code>
A. OPTIONAL {B.}	<b>Valeurs optionnelles</b> : réunir les résultats de A et B par leurs valeurs en commun. Garder toutes les solutions de A qu'il y ait une solution dans B ou non. <code>?enfant wdt:P22 wd:Q1339.</code> <code>OPTIONAL { ?enfant wdt:P569 ?naissance. }</code>
{A.} UNION {B.}	<b>Union</b> : réunir toutes les solutions de A et celles de B. <code>{ ?enfant wdt:P569 ?naissance. } #avec date de naissance</code> <code>UNION { ?enfant wdt:P570 ?décès. } #avec date de décès</code>
A. MINUS {B.}	<b>Différence</b> : trouver les solutions de A et celles de B et ne garder que les solutions de A qui ne sont pas compatibles avec celles de B. <code>?enfant wdt:P22 wd:Q1339. #enfants de Bach</code> <code>MINUS { ?enfant wdt:P570 ?décès. } #sans date de décès</code>
A. FILTER NOT EXISTS {B.}	Méthode quasi-équivalente à la précédente, mais généralement plus lente. <code>?enfant wdt:P22 wd:Q1339.</code> <code>FILTER NOT EXISTS { ?enfant wdt:P570 ?décès. }</code>





# FNCTIONS



# QUALIFICATIF, RÉFÉRENCE, RANG



Commande	Description et exemple		
<b>DATATYPE</b>	<code>FILTER(DATATYPE(?object) = xsd:dateTime).</code>	Généralité	
<b>STR</b> (value)	Convertit une valeur en chaîne de caractères ou une chaîne en		
<b>IRI</b> (string)	adresse IRI : <code>BIND(STR(?item) AS ?string).</code>		
<b>LANG</b> (string)	Renvoie le code langue des libellés, descriptions, alias et textes : <code>BIND(LANG(?nativename) AS ?language).</code>		
<b>BIND</b>	<b>BIND</b> permet d'assigner la valeur d'une expression à une variable : <ul style="list-style-type: none"> <li><code>BIND(?décès - ?naissance AS ?age).</code></li> <li><code>BIND(IF(BOUND(?pseudo), ?pseudo, ?ecrivainLabel) AS ?libellé).</code></li> <li><code>BIND(COALESCE(?pseudo, ?ecrivainLabel) AS ?libellé).</code></li> </ul>		
<b>BOUND</b>			
<b>IF</b>			
<b>IN</b>	Vérifie si la valeur figure ou non dans une liste :		Chaîne de caractère
<b>NOT IN</b>	<code>FILTER(?city IN (wd:Q84, wd:Q90)).</code>		
<b>isBLANK</b> (variable)	Vérifie si la valeur est une "valeur inconnue" : <code>FILTER isBLANK(?gender).</code>		
<b>STRLEN</b> (string)	Taille d'une chaîne de caractère : <code>STRLEN(?name)</code> où ?name est "ABC" renvoie 3. Voir aussi <b>SUBSTR</b> (string, beginposition, stringlength).		
<b>UCASE</b> (string)	Mise en majuscule. Voir aussi <b>LCASE</b> .		
<b>STRSTARTS</b> (string, comparestring)	Vérifie si la chaîne de caractères débute par une certaine valeur : <code>STRSTARTS("ABCDEFGH", "FGH")</code> renvoie "faux". Voir aussi <b>STREND</b> , <b>CONTAINS</b> , <b>STRBEFOR</b> et <b>STRAFTER</b> .		
<b>ENCODE_FOR_URI</b> (string)	<code>ENCODE_FOR_URI("ABC DĚF")</code> renvoie "ABC%20D%C3%8BF". Voir aussi <code>wikibase:decodeUri</code> .		
<b>CONCAT</b> (string1, string2)	<code>CONCAT("ABC", "XYZ")</code> renvoie "ABCXYZ".		
<b>LANGMATCHES</b> (language tag, language range)	<code>FILTER LANGMATCHES ( LANG(?label), "fr").</code>		
<b>REGEX</b> (string, pattern, flag)	Vérifie si la chaîne de caractère correspond à l'expression régulière : <code>FILTER REGEX (?itemLabel, "(ow itz)\$").</code>		
<b>REPLACE</b> (string, pattern, replacement, flag)	<code>REPLACE("ABCDEFGH", "[AEIOU]", "")</code> enlève toutes les voyelles.	Nombre	
<b>ABS</b> (nombre)	Valeur absolue : <code>ABS(-1)</code> renvoie 1.		
<b>ROUND</b> (nombre)	Arrondi : <code>ROUND(1.4)</code> renvoie 1. Voir aussi <b>CEIL</b> et <b>FLOOR</b> .		
<b>RAND</b>	Renvoie une valeur aléatoire entre 0 et 1. <code>SELECT ?image (MD5(CONCAT(STR(?item), STR(RAND())))) AS ?random</code> <b>WHERE</b> {...} <b>ORDER BY</b> ?random		
<b>COUNT</b>	Compte le nombre d'éléments. <code>COUNT(*)</code> comptabilise tous les résultats.		
<b>MIN, MAX, SUM, AVG</b>	Renvoie la valeur minimum, maximum, la somme ou la moyenne.		
<b>NOW</b> ()	Date et heure du jour.		
<b>YEAR</b> (date)	Extrait l'année d'une date. Autres commandes utiles dans le traitement de dates : <b>MONTH</b> , <b>DAY</b> , <b>HOURS</b> , <b>MINUTES</b> et <b>SECONDS</b> .		
<b>geof:distance</b>	Distance entre deux points, en kilomètres.		Date
<b>geof:longitude</b>	Longitude d'une coordonnée géographique. Voir aussi <b>geof:latitude</b> , <b>wikibase:geoLongitude</b> , <b>wikibase:geoLatitude</b> , <b>wikibase:center</b> , <b>wikibase:radius</b> , <b>wikibase:globe</b> , <b>wikibase:distance</b> , <b>wikibase:cornerWest</b> ...		
<b>AUTO_COORDINATES</b>	Renvoie votre localisation.		
<b>SAMPLE</b>	Renvoie qu'une valeur : <code>SAMPLE (?date) AS ?date + GROUP BY</code>	Coordonnée	
<b>GROUP_CONCAT</b>	Concaténation : <code>(GROUP_CONCAT(?métierLabel; SEPARATOR="; ") AS ?métiers).</code>		
		Agrège	

Donnée	Exemple	
<b>Qualificatif</b>	<code>wd:Q42 p:P69 ?déclaration.</code> <code>?déclaration ps:P69 ?éducation.</code> <code>?déclaration pq:P580 ?début.</code>	<code>wd:Q42 p:P69 [ps:P69 ?éducation; pq:P580 ?qualificatif].</code>
<b>Référence</b>	<code>wd:Q42 p:P69 ?déclaration.</code> <code>?déclaration ps:P69 ?éducation.</code> <code>?déclaration prov:wasDerivedFrom ?noeudref.</code> <code>?noeudref pr:P248 ?référence.</code>	<code>wd:Q42 p:P69 [ps:P69 ?éducation; prov:wasDerivedFrom [pr:P248 ?référence ]].</code>
<b>Rang</b>	Meilleur	<code>SELECT * WHERE { wd:Q692 wdt:P569 ?valeur. }</code>
	Préféré	<code>SELECT * WHERE { wd:Q692 p:P569 ?déclaration . ?déclaration ps:P569 ?valeur . ?déclaration wikibase:rank wikibase:PreferredRank. }</code>
	Normal	... <code>?déclaration wikibase:rank wikibase:NormalRank.</code>
	Obsolète	... <code>?déclaration wikibase:rank wikibase:DeprecatedRank.</code>
	Non obsolète	... <code>MINUS { ?déclaration wikibase:rank wikibase:DeprecatedRank. }</code>
<b>Interwiki</b>	Un wiki donné	<code>?article schema:about ?item.</code> <code>?article schema:inLanguage "fr".</code> <code>?article schema:isPartOf &lt;https://fr.wikipedia.org/&gt;.</code>
	Nombre d'interwikis	<code>SELECT ?linkcount WHERE { ?humain wikibase:sitelinks ?linkcount. } GROUP BY ?linkcount ORDER BY DESC(?linkcount)</code>
<b>Badge</b>	<code>?label schema:isPartOf ?wiki.</code> <code>?label wikibase:badge ?badge.</code> <code>?label schema:about ?item.</code>	



## VOIR AUSSI

<p><b>Erreurs fréquentes :</b></p> <ul style="list-style-type: none"> <li>Temps d'exécution dépassé : filtrez pour diminuer le nombre de résultats, n'utilisez pas le service Label</li> <li>Ponctuation : chaque phrase nécessite un point, parenthèse manquante, parenthèses à la place d'accolades.</li> <li>Problème d'agrégation : complétez <b>SELECT</b> ou <b>GROUP BY</b></li> <li>Résultat manquant : utilisez <b>OPTIONAL</b></li> <li>Doubleton : utilisez <b>SELECT DISTINCT</b></li> </ul>	<p><b>Requêtes imbriquées :</b> utiliser les résultats d'une requête pour formuler une autre requête</p> <pre>SELECT... WHERE {   { SELECT...     WHERE {...}   }   ...} </pre>
<p><b>Outil de visualisation d'ontologies :</b> <a href="https://angryloki.github.io/wikidata-graph-builder">https://angryloki.github.io/wikidata-graph-builder</a></p> <p><b>API MediaWiki :</b> <code>SERVICE wikibase:mwapi</code> permet d'appeler l'API dans le client SPARQL.</p>	<p><b>Requêtes fédérées :</b> interroger simultanément plusieurs points d'accès SPARQL à l'aide de la clause <b>SERVICE</b>.</p> <pre>SELECT... {... SERVICE ...{...} ...}</pre>

