**Skript Videoguide 2 Modul 2 (KG)**

**Aufgabe**:

In dieser Übung beschäftigen wir uns damit SPARQL-Abfragen in der Datenbank WikiData anzuwenden. Wir schauen uns die Struktur einiger beispielhaften Abfragen und deren Ergebnisse in der Wikidata-Query an. Anschließend erstellen wir mithilfe des Query-Builders Eine eigene einfache Abfragen.

Folge dem Screencast und nutze die vorgestellten Ressourcen gerne um weitere Abfragen durchzuführen.

**Skript**

INTRO

**[Einleitung]**

Hallo und herzlich willkommen zu diesem Videoscreencast über Wikidata. Wir werden uns mit den Grundlagen von Wikidata beschäftigen und erfahren, wie wir Informationen speichern und abrufen können. Lass uns also loslegen!

WIKIDATA Home <https://www.wikidata.org/wiki/Wikidata:Main_Page>

**[Was ist Wikidata?]**

Was ist eigentlich Wikidata? Wikidata ist eine frei zugängliche Wissensdatenbank, die von der Wikimedia Foundation entwickelt wurde. Sie wurde ins Leben gerufen, um strukturierte Daten zu sammeln und zu speichern, die in anderen Wikimedia-Projekten wie Wikipedia verwendet werden können. Anders als Wikipedia, das sich auf textbasierte Informationen konzentriert, ist Wikidata darauf ausgerichtet, Fakten und Beziehungen zwischen verschiedenen Entitäten zu erfassen.

**[Welchen Zweck soll Wikidata erfüllen?]**

Wikidata hat mehrere Ziele. Es soll eine zentrale Quelle für strukturierte Daten sein, die von anderen Projekten genutzt werden können. Es ermöglicht die Verknüpfung von Informationen in verschiedenen Sprachen und Projekten. Darüber hinaus sollen durch die Nutzung von Wikidata maschinenlesbare Informationen bereitgestellt werden, die von KI-Systemen, Bots und anderen Anwendungen genutzt werden können.

WIKIDATA Introduction <https://www.wikidata.org/wiki/Wikidata:Introduction>

**[In welcher Form sind Informationen gespeichert?]**

Die Daten werden in sogenannten "Items" und "Properties" organisiert. Ein Item repräsentiert eine Entität wie eine Person, einen Ort oder ein Konzept. Jedes Item hat eine eindeutige ID, die als QID bezeichnet wird. Properties sind Eigenschaften oder Merkmale, die zu einem Item gehören, z. B. das Geburtsdatum einer Person oder die Fläche eines Landes die als P-IDs hinterlegt sind.

**[Erkläre die Item-IDs]**

Aber lasst uns noch genauer auf die Item-IDs eingehen.

ALBERT EINSTEIN <https://www.wikidata.org/wiki/Q937>

Entitäten haben eine ID die mit einem Q gefolgt von einer Zahl ausgestattet ist. Diese IDs dienen dazu, einzelne Items eindeutig zu identifizieren und sie von anderen zu unterscheiden. Zum Beispiel hat Albert Einstein die Q-ID Q937. Diese IDs sind wichtig, wenn wir spezifische Informationen über ein bestimmtes Item abrufen möchten.

Date of birth <https://www.wikidata.org/wiki/Property:P569>

P-IDs hingegen werden für Properties verwendet, die die Eigenschaften oder Merkmale der Items darstellen. Eine P-ID besteht aus einem "P" gefolgt von einer Zahl. P569 steht zum Beispiel für das Property "Geburtsdatum". Die P-IDs kennzeichnen die verschiedenen Eigenschaften, die den Items zugeordnet sind. Sie ermöglichen es, Informationen über die Entitäten strukturiert zu speichern und abzurufen.

Property Verzeichnis + Searchbar auf Webseite <https://www.wikidata.org/wiki/Special:Search?profile=advanced&fulltext=Search&ns120=1>

Um herauszufinden, wofür eine bestimmte Q-ID oder P-ID steht, kannst du das offizielle Wikidata-Property-Verzeichnis und die Wikidata-Website selbst nutzen. Im Wikidata-Verzeichnis kannst du nach spezifischen Items oder Properties suchen und weitere Informationen über sie erhalten. Die Wikidata-Website bietet eine umfangreiche Dokumentation und eine Vielzahl von Hilfeseiten, die dir dabei helfen, die Bedeutung und Verwendung der verschiedenen IDs besser zu verstehen.

WIKIDATA QUERY SERVICE <https://query.wikidata.org/>

**[Stelle den Wikidata Query Service mit Hilfe von 2 Beispielen vor]**

Nun werden wir uns den Wikidata Query Service anschauen, mit dem wir Daten aus Wikidata abfragen können. Der Wikidata Query Service verwendet die Abfragesprache SPARQL. Lasst uns einige Beispiele für SPARQL-Abfragen betrachten.

**[Beispiel 1]**

Das erste Beispiel stellt eine sehr simple Abfrage da. Die Abfrage zielt darauf ab, Informationen über Katzen aus der Wikidata-Datenbank abzurufen. Die Abfrage gibt die Variablen "?item" und "?itemLabel" zurück.

#Katzen

SELECT ?item ?itemLabel

WHERE

{

?item wdt:P31 wd:Q146. # Muss von einer Katze sein

SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO\_LANGUAGE],en". } # Hilft dabei, das Label in deiner Sprache zu erhalten, falls nicht verfügbar, wird die englische Sprache verwendet

}

Im Block "WHERE" suchen wir nach Items, die die Eigenschaft "P31" (ist eine/r/s) mit dem Wert "Q146" (Katze) haben. Dadurch stellen wir sicher, dass die abgerufenen Items tatsächlich Katzen sind.

Der Block "SERVICE wikibase:label" ermöglicht es uns, die Labels oder Bezeichnungen der Ergebnisse in der gewünschten Sprache zu erhalten. Mit "bd:serviceParam wikibase:language "[AUTO\_LANGUAGE],en"" wird angegeben, dass wir die Labels in der Sprache "[AUTO\_LANGUAGE]" (automatisch ermittelt) und Englisch ("en") erhalten möchten.

Durch die Abfrage erhalten wir eine Liste von Katzen-Items und ihren dazugehörigen Labels.

**[Beispiel 2]**

Als Beispiel 2 betrachten wir eine SPARQL-Abfrage, die dazu dient, eine Karte mit Krankenhäusern anzuzeigen. Diese Abfrage ist bereits etwas komplexer. Sie nutzt die Wikidata-Datenbank, um die geografischen Koordinaten von Krankenhäusern abzurufen und auf einer Karte darzustellen.

Hier ist die Beschreibung der Abfrage:

#Karte mit Krankenhäusern #added 2017-08 #defaultView:Map SELECT DISTINCT \* WHERE { ?item wdt:P31/wdt:P279\* wd:Q16917; wdt:P625 ?geo . }

Nach einigen auskommentierten Zeilen beginnt die eigentliche Abfrage mit dem Schlüsselwort "SELECT DISTINCT \*". Dies bedeutet, dass alle Ergebnisvariablen zurückgegeben werden sollen. Das Sternchen (\*) steht hier als Platzhalter für alle Variablen.

Der Block "WHERE" gibt die Bedingungen an, die die Ergebnisse erfüllen müssen. In diesem Fall suchen wir nach Items (?item), die sowohl die Eigenschaft "P31" (ist ein/eine) als auch die Eigenschaft "P625" (geografische Koordinaten) besitzen. Die Besonderheit hierbei ist, dass wir die Eigenschaft "P31" mit dem Operator "P279\*" (subclass of) kombinieren. Dies bedeutet, dass wir nicht nur nach direkten Instanzen suchen, sondern auch nach Unterinstanzen der Klasse "Q16917" (Krankenhaus).

Die Variable "?geo" wird verwendet, um die geografischen Koordinaten des Items abzurufen.

Zusammengefasst sucht diese Abfrage nach Krankenhäusern in der Wikidata-Datenbank und ruft ihre geografischen Koordinaten ab, um sie auf einer Karte darzustellen. Durch die Verwendung des Standardansichtsformats "Map" wird gewährleistet, dass die Ergebnisse als Karte visualisiert werden.

**[Beispiel 3]**

Umso umfangreicher die Abfrage, desto spezifischer werden die Ergebnisse.

Bei dieser SPARQL-Eingabe handelt es sich beispielsweise um eine Abfrage, die darauf abzielt, eine Karte mit den Geburts- und Sterbeorten von Wirtschaftswissenschaftlern zu erstellen. Die punktuell dargestellten Orte werden je nach hinterlegtem Zeitalter mit verschiedenen Farben eingefärbt.

#Karte mit den Geburts- und Sterbeorten von Wirtschaftswissenschaftlern unterschiedlich gefärbt nach Ära

#defaultView:Map

SELECT DISTINCT ?person ?name ?birthplace ?birthyear ?coord ?layer WHERE {

{?person wdt:P106 wd:Q188094} UNION {?person wdt:P101 wd:Q8134}

?person wdt:P570 ?dod;

wdt:P19 ?place .

?place wdt:P625 ?coord

OPTIONAL { ?person wdt:P569 ?dob }

BIND(YEAR(?dob) as ?birthyear)

BIND(IF( (?birthyear < 1700), "Pre-1700", IF((?birthyear < 1751), "1700-1750", IF((?birthyear < 1801), "1751-1800", IF((?birthyear < 1851), "1801-1850", IF((?birthyear < 1901), "1851-1900", IF((?birthyear < 1951), "1901-1950", "Post-1950") ) ) ) )) AS ?layer )

?person rdfs:label ?name FILTER (lang(?name) = "en")

?place rdfs:label ?birthplace FILTER (lang(?birthplace) = "en")

} ORDER BY ?birthyear

**[Erstelle eine eigene Abfrage mit einem geografischen Thema]**

Auf diese Abfrage werden wir jetzt aber nicht mehr genauer eingehen, sondern selbst einmal versuchen eine eigene SPARQL-Abfrage zu erstellen. Wir möchten uns eine Liste aller Hauptstädte ausgeben lassen, die in Europa liegen.

Abfrage nach allen Hauptstädten in Europa:

```

SELECT ?capital ?capitalLabel

WHERE {

?capital wdt:P31 wd:Q5119.

?capital wdt:P1376 ?country.

?country wdt:P30 wd:Q46.

?capital wdt:P17 ?country.

SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO\_LANGUAGE],en". }

}

```

Die Abfrage beginnt mit dem Schlüsselwort "SELECT", gefolgt von den Variablen "?capital" und "?capitalLabel". Hier geben wir an, welche Ergebnisvariablen wir aus der Abfrage zurückerhalten möchten. In diesem Fall wollen wir sowohl die Variable "?capital" als auch die Variable "?capitalLabel" abrufen. Die Variable "?capital" repräsentiert die Hauptstadt, während die Variable "?capitalLabel" den Namen der Hauptstadt enthält.

Der nächste Teil der Abfrage ist das Schlüsselwort "WHERE". Hier geben wir an, welche Bedingungen die Ergebnisse erfüllen müssen. Alle Bedingungen werden innerhalb einer geschweiften Klammer Zusammengefasst.

Die erste Bedingung lautet "?capital wdt:P31 wd:Q5119". Hier suchen wir nach Items, bei denen das Property "P31" (Instanz von) den Wert "Q5119" (Hauptstadt) hat. Das bedeutet, dass wir nur Items auswählen, die als Hauptstadt klassifiziert sind.

Die zweite Bedingung "?capital wdt:P1376 ?country" gibt an, dass das Item "?capital" eine Beziehung zum Property "P1376" (Land) hat, das in der Variable "?country" gespeichert wird. Diese Bedingung stellt sicher, dass die Hauptstadt mit einem Land verknüpft ist.

Die dritte Bedingung "?country wdt:P30 wd:Q46" besagt, dass das Land in der Variable "?country" das Property "P30" (ist Teil von) den Wert "Q46" (Europa) haben muss. Dies stellt sicher, dass das Land zum Kontinent Europa gehört.

Die vierte Bedingung "?capital wdt:P17 ?country" stellt sicher, dass die Hauptstadt mit dem Land in der Variable "?country" verknüpft ist. Das Property "P17" (Land) wird verwendet, um die Beziehung herzustellen.

Schließlich verwenden wir das Schlüsselwort "SERVICE" gefolgt von "wikibase:label", um die Labels oder Bezeichnungen der Ergebnisse abzurufen. Mit "bd:serviceParam wikibase:language "[AUTO\_LANGUAGE],en"" geben wir an, dass wir die Labels in der Sprache "[AUTO\_LANGUAGE]" (automatisch ermittelt) und Englisch ("en") erhalten möchten.

Probiert die Abfrage selbst aus und schaut, welche Hauptstädte Europas ihr als Ergebnis erhaltet!

**[Fasse das Gelernte zusammen und schließe das Video ab]**

Ihr habt nun einen ersten Überblick über Wikidata und den Wikidata Query Service erhalten. Wir haben gelernt, wie Informationen in Items und Properties organisiert sind und wie wir SPARQL-Abfragen verwenden können, um gezielte Informationen abzurufen.

Wikidata bietet eine unglaubliche Menge an Wissen und die Möglichkeit, dieses Wissen auf verschiedene Weise zu nutzen.

Ich hoffe, dass euch dieses Video geholfen hat, die Grundlagen von Wikidata zu verstehen. Vielen Dank fürs Zuschauen.