BRAZILIAN LAWS

Modeling the Brazilian legislation in Wikidata

This publication is part of a project in Wikidata called **WikiProject Brazilian Laws**[1], that have the goal to develop the Brazilian legislation in Wikidata. For that, is necessary to gather and scrape the information about the topic from different official sources, clean this data, model it in accordance with the properties and values on Wikidata and upload it. The data imported into Wikidata has the potential to reverberate in other projects, such as Wikisource and Wikipedias in multiple languages.

The data imported into Wikidata in this project so far embrace all the 28 thousand laws and decree-laws published into the official sources of the Brazilian government and parliament, all of them with federal jurisdiction: LeXML[2] and Palácio do Planalto[3] [4]. A tool with a broader scope for legislation types was also created to allow users to create Brazilian legislation items in Wikidata in a friendly manner; In total, these items represent a small fraction of the total items available in these sources, that are, in April 2021, next to 8.5 million Brazilian laws, decrees, provisory measures, court judgments etc in all sort of jurisdiction levels.

This project was developed and organized by the members of the **Wiki Movimento Brasil**[5] user group, affiliated with the Wikimedia Foundation, with the support of the **WikiCite**[6] project, in the format of a Grant.[7]

1 Wikidata:WikiProject Brazilian Laws (April 30, 2021). In *Wikidata*. Available at: https://www.wikidata.org/wiki/Wikidata:WikiProject_Brazilian_Laws/

2 LeXML Rede de Informação Legislativa e Jurídica (April 30, 2021). Available at: https://www.lexml.gov.br/

3 Legislação Federal Brasileira (April 30, 2021). Available at: https://legislacao.presidencia.gov.br/

4 Portal da Legislação (April 30, 2021). Available at : http://www4.planalto.gov.br/legislacao/

5 Wiki Movement Brazil User Group (April 30, 2021). Available at: https://meta.wikimedia.org/wiki/Wiki_Movement_Brazil_User_Group/

6 WikiCite (April 30, 2021). Available at: https://meta.wikimedia.org/wiki/WikiCite/

7 Wikicite/grant/Brazilian Laws: Modeling the Brazilian legislation in Wikidata (September 26, 2020). In *Wikimedia Meta-Wiki*. Available at: https://meta.wikimedia.org/wiki/Wikicite/grant/Brazilian_Laws:_Modeling_the_Brazilian_legislation_in_Wikidata/

This project was born in September of 2020, with a grant proposal to WikiCite, to solve a problem in the modeling of the Brazilian legislation items in Wikidata. This topic, Brazilian legislation, is a good example of a type of data that it is clearly of public interest, but it is presented in its sources in such a complex or human-dependable way that practically no analysis can be made using them.

The Brazilian legislation is gathered and presented mainly in two official websites: *Palácio do Planalto* (official website of the Presidency of Brazil) and the *LeXML Project*. The Palácio do Planalto website has a search system that allows a user to query a list of laws, decree-laws, provisory measures or the constitutions; For each legislation entity, there is a page with the complete text of the legislation, and a data sheet with various metadata. The LeXML Project website has a similiar system for searching an item, and although it presents a much better, machine-readable and accessible API, requires that the user be very familiar with the platform and know exactly what to look for.

Data about legislation in Brazil, and its contents are not subject to copyright. Section 5, Subsection XIV of the Constitution of the Federative Republic of Brazil states that "access to information is ensured to everyone". With these two principles in mind, and comparing them to the concept of open data being data that anyone could be able to access, use, analyze and share, the analysis of the tools available was that, although in the right track, they do not make Brazilian laws available either in an open or in a transparent manner, as they do not pass through the sieve of information being available in open formats in which they can be processed or used as desired.

Wikidata, as a database for open data in public domain, is a great platform to work with this type of data. In fact, some items were already there, mainly originated from articles in Wikipedia about the most famous laws in Brazil. In late September of 2020, there were 129 items on Wikidata, 104 uses of the template *citar lei* ("cite act" in English) and 5958 references to the Palácio do Planalto in Portuguese Wikipedia.

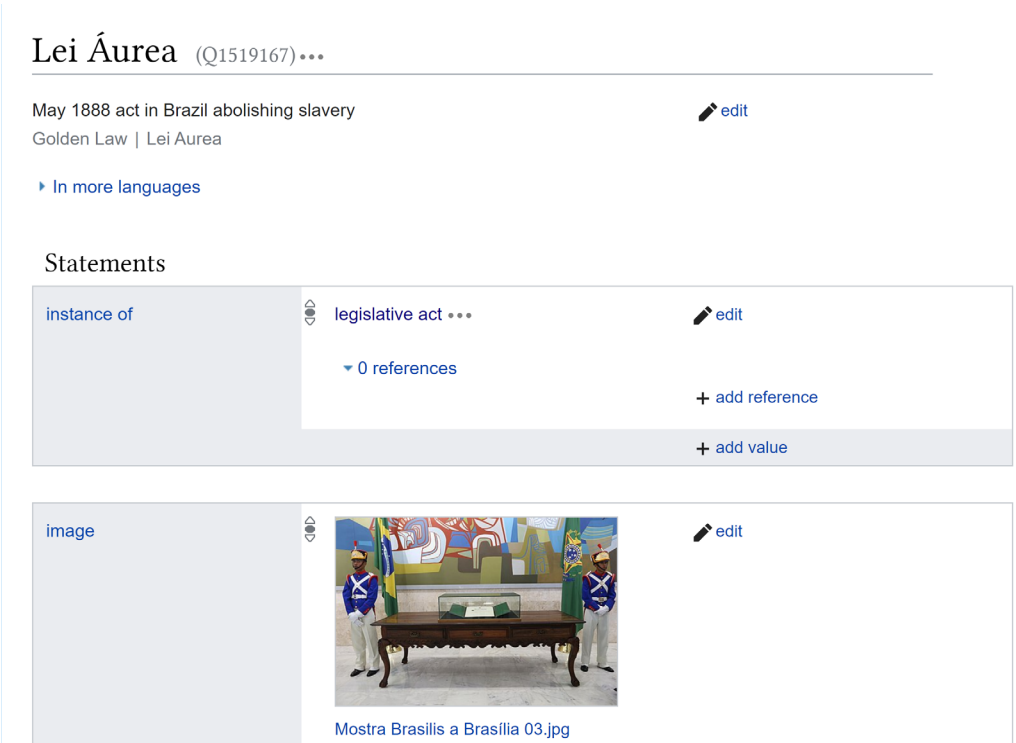
The solution proposed in this project involved set up a process to model and import the entire framework of the Brazilian legislation to Wikidata. Wikidata ticks all the boxes for transparent data: It is accessible, has a large set of tools that allow its use in any project, can provide relations and be agregated and queried for analysis and can be shared in multiple open formats, such as json, csv and html. Also have a well documented API, which make the data not only human-readable, but machine-readable as well. With this data available on Wikidata, users can, for example, use it as automated references on Wikipedia articles using *Cite Q* template, organize efforts to transcribe its contents to WikiSource and query and analyse legislation data that is currently not possible in any official platform.

Along with the process of modeling and importing the data, we proposed to create a user friendly tool that one can curate and then create a Brazilian legislation item on Wikidata by pasting its URL. This tool is stored in Toolforge, a hosting enviroment for tools focused on the Wikimedia movement. To improve the communities capacity building, we also proposed a technical training, called *Wikidata Lab*, detailing the project and its methodology. The proposed activities, in list format were:

- Create a project page on wiki to organize the activites;
- Create a schema crosswalk between Wikidata and the official websites;
- Model the metadata for each type of Brazilian legislation act on federal level;
- Create a dynamic lexicon of distinct terms of categorization of the Brazilian legislation;
- Curate the Brazilian legislation that is already in Wikidata;
- Load all the legislation on Palácio do Planalto and LeXML websites in Wikidata;
- Develop an interface on Toolforge to reconciliate the metadata from the Palácio do Planalto and LexML websites and to import into Wikidata;
- Create in parallel the documentation for all steps;
- Promote a Wikidata Lab to improve community capacity.

The first step to model a subject or an item in Wikidata is to understand its *ontology*. In information science and in semanthic web, the term ontology means the formal representation of the properties of an entity and how they are related with each other and with other entities in a machine-processable manner. Instances, classes, properties, relations and restrictions are the elements of ontology that stand out the most when working with Wikidata.

To understand the ontology, it is necessary to investigate the properties and values for existing items in the Wikidata database. To do this, one have to search for known existing items; In general, Wikipedia articles are a good place to start, as they have their interwiki linking dealt in Wikidata, so if there is an article, it is highly probable that an item on Wikidata exists as well. In our case, laws of great repercussion, like the *Law of Free Birth*, the *Brazilian Civil Rights Framework for the Internet* and the *Ban on gambling in Brazil* are some of the items that we investigated. Other legislation of other countries were analysed as well.



Screenshot of Wikitada page of the item Q1519167 ("Golden Law", in English)

The items within the scope of this project were defined to be laws and decree-laws, or "leis" and "decretos-leis" in Portuguese. Law, in the Brazilian legal context, is a written document edited by a competent authority in accordance with some legislative procedure to convey legal norms. To become a law, this document is presented as a bill to the legislative branch of the government either by the legislative branch itself, by the executive branch or by a popular initiative; If the legislative branch approves, the bill is sent to the executive branch for promulgation. Only then it becomes a law. A decree-law, in the other hand, still in the Brazilian context, it was a document edited, approved and promulgated by the executive branch, without discussion on the legislative branch, that had force of law and required urgency or relevant public interest to be emited. This type of legislation act was extinct by the Constitution of 1988, but it was heavily used throughout the history of Brazil, especially during the dictatorships endured by the country. So, in order to model the information, we defined the scope of our instances to be, *a priori*, only **laws (Q820655)** and **decree-laws (Q2571972)**.

The second step to model the ontology is to define the properties and relations for each of the instances. In order to do that, it is important to understand the state of the art of the type of items we want to model. In our case, there were other WikiProjects focused on legislation of other countries, that had their modeling already done, the main ones being the WikiProject US legislation, the WikiProject Japan/Law and the WikiProject France/Législation. With the knowledge of which properties are already available on Wikidata, the next thing to do is to list and correspond all the properties on the databases we want to import with the properties on Wikidata. This is called a "schema crosswalk", and is meant to help us navigate throughout the different databases structures, or schemas, and map all the correspondencies of the metadata fields among them. Not all properties will have a correspondent on Wikidata; If a property is structured enough, or if it represents a type of identifier, it can have its creation proposed and if the community approves, it can be created. We proposed two properties in this project: LeXML Brazil ID and law digest. In the next pages we present the schema crosswalk for this project.

LeXML	Palácio do Planalto	Wikidata	Description
autoridade facet-autoridade	-	approved by (P790)	organization that issued the legislation
tipoDocumento facet-tipoDocumento	-	instance of (P31)	type of legislation
localidade facet-localidade	-	country (P17); applies to jurisdiction (P1001)	geographic unit that the legislation has jurisdiction. country (P17) can be implied
date	Data de Publicação	publication date (P577)	date the legislation was published in an official journal (such as the Diário Oficial da União (Q3712237)) and came into force)
description	Ementa	law digest (P9376)	short text that summarizes a law, part of the preamble
identifier	-	-	8-9 digits long number which is also used as an identifier. There is no documentation about where it is used
subject	Assunto	main subject (P921)	main topics of the legislation
title	Page's title (h1 tag)	labels, title (P1476)	official name of the legislation
urn	-	LexML Brazil ID (P9119)	uniform resource identifier that follows the Uniform Resource Name (Q76497) scheme.
-	Data de assinatura	point in time (P585) (qualifier of approved by (P790))	date in which the legislation was signed or put into effect
-	Situação	-	current state of the legislation (if it was repealed, amended, overruled etc)
-	Chefe de Governo	signatory (P1891) (qualifier of approved by (P790))	person who issued the legislation. Can be infered to be the sitting president at the time
-	Origem	legislated by (P467)	institution in which the legislation was originated

LeXML	Palácio do Planalto	Wikidata	Description
-	Fonte	published in (P1433)	official journal in which the legislation was published. Qualifiers stated as (P1932), page(s) (P304) and point in time (P585) can be infered from this value
-	Link	full work available at URL (P953)	link to the full text of the legislation
-	Referenda	-	institutions or group of citizens who proposed the creation of the legislation
-	Alteração	amended by (P2567), repealed by (P2568), repeals (P3148), overrules (P4006)	other legislations that altered this legislation
-	Correlação	-	other legislations to which the current one relates
-	Veto	-	whether the legislation has been totally or partially vetoed, and which parts
-	Classificação de direito	facet of (P1269)(?)	area of law (Q1756157) related to this legislation
-	Observação	-	miscellaneous footnotes
-	-	place of publication (P291)	place of publication. Can be inferred from applies to jurisdiction (P1001) and publication date (P577)
-	-	language of work or name (P407)	language of the legislation. Can be assumed to be Brazilian Portuguese (Q750553)
-	-	legal citation of this text (P1031)	legal citation of the legislation. Infered from title (P1476)
-	-	on focus list of Wikimedia project (P5008)	wikiproject that this law is of particular interest. e.g. WikiProject Brazilian Laws (Q105091640)

There are two main official platforms to access Brazilian legislation: the LexML Project and the Palácio do Planalto website. Both sites provide a search interface so one can query items in a limited set of parameters. LeXML also provides an API and has controlled dictionary for its metadata, but neither platform offers the option to download its contents. All the metadata about legislation and other official acts in Brazil, including their full text, are not subject to copyright.

The schema crosswalk of these entities shows which metadata each website provides; None of them contains all the metadata available for a legislation. So, as an activity of this project, both websites were scrapped using Python scripts to fetch as much information about the legislation they hold as possible. Latter on, we compiled all this information into a spreadsheet, wikidatified and uploaded it to Wikidata.

In the next pages, we illustrate the kind of metadata present in both websites, using Law No. 13709 of August 14, 2018 (Q105691828) as an example. Next, we go further into the details of the scripts used in each step of the process of scrapping them.

The table below is a direct representation of the metadata shown in its LeXML page:

Localidade	Brasil
Autoridade	Federal
Título	Lei nº 13.709, de 14 de Agosto de 2018
Data	14/08/2018
Apelido	Lei Geral de Proteção de Dados Pessoais (LGPDP)
Apelido	LEI-13709-2018-08-14 , LEI GERAL DE PROTEÇÃO DE DADOS
Ementa	Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet).
Nome Uniforme	urn:lex:br:federal:lei:2018-08-14;13709
Mais detalhes	Senado Federal
Mais detalhes	Câmara dos Deputados
Projeto de Origem	[Projeto de Lei (CD) nº 4060/2012 > Projeto de Lei da Câmara nº 53/2018 > Veto nº 33/2018 : Lei nº 13.709 de 14/08/2018]

LeXML API result for the query searching this item is presented bellow. Notice the tree structure, in which each metadata, for each result, is presented as a string of text inside a tag element.


```
<srw:searchRetrieveResponse xmlns:srw_dc="info:srw/schema/1/dc-schema" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:srw="http://www.loc.gov/zing/srw/" xmlns:xsi="http://www.w3.org/2001/XMLSchema">
	<srw:version>1.1</srw:version>
	<srw:numberOfRecords>1</srw:numberOfRecords>
	<srw:records>
		<srw:record>
			<srw:recordPacking>XML</srw:recordPacking>
			<srw:recordSchema>info:srw/schema/1/dc-v1.1</srw:recordSchema>
			<srw:recordData>
				<srw_dc:dc xsi:schemaLocation="info:srw/schema/1/dc-schema http://www.loc.gov/z3950/agency/zing/srw/dc-schema.xsd">
					<tipoDocumento>Lei</tipoDocumento>
					<facet-tipoDocumento>Legislação::Lei</facet-tipoDocumento>
					<dc:date>2018-08-14</dc:date>
					<urn>urn:lex:br:federal:lei:2018-08-14;13709</urn>
					<localidade>Brasil</localidade>
					<facet-localidade>Brasil</facet-localidade>
					<autoridade>Federal</autoridade>
					<facet-autoridade>Federal</facet-autoridade>
					<dc:title>Lei nº 13.709, de 14 de Agosto de 2018</dc:title>
					<dc:title>Lei nº 13.709 de 14/08/2018</dc:title>
					<dc:description>Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet).</dc:description>
					<dc:description>Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet).</dc:description>
					<dc:subject>CRITERIOS , TRATAMENTO , PROTEÇÃO , SEGURANÇA , SIGILO , DADOS PESSOAIS , PESSOA FISICA , PESSOA JURIDICA . ALTERAÇÃO , MARCO REGULATORIO , INTERNET , DIREITOS , USUARIO , ARMAZENAGEM , DADOS PESSOAIS , REGISTRO , EXCLUSÃO .</dc:subject>
					<dc:type>html</dc:type>
					<dc:identifier>000967073</dc:identifier>
				</srw_dc:dc>
			</srw:recordData>
		</srw:record>
	</srw:records>
	<srw:echoedSearchRetrieveRequest>
		<srw:version>1.1</srw:version>
		<srw:query>urn ="urn lex br federal lei 2018 08 14 13709"</srw:query>
		<srw:startRecord>1</srw:startRecord>
		<srw:maximumRecords>1</srw:maximumRecords>
		<srw:recordPacking>xml</srw:recordPacking>
		<srw:recordSchema>dc</srw:recordSchema>
	</srw:echoedSearchRetrieveRequest>
</srw:searchRetrieveResponse>
```

The same legislation has a record page in Palácio do Planalto, with other metadata:

Data de assinatura:	14 de Agosto de 2018
Ementa:	DISPÕE SOBRE A PROTEÇÃO DE DADOS PESSOAIS E ALTERA A LEI Nº 12.965, DE 23 DE ABRIL DE 2014 (MARCO CIVIL DA INTERNET). Vigência Veto Parcial
Situação:	Não consta revogação expressa
Chefe de Governo:	MICHEL TEMER
Origem:	Executivo
Data de Publicação:	15 de Agosto de 2018
Fonte:	D.O.U de 15/08/2018, pág. nº 59
Link:	Texto integral
Referenda:	---
Alteração:	MPV 869, DE 27/12/2018: ALTERA ARTS. 3º, 4º, 5º, 11, 20, 26, 27, 29; ACRESCE ARTS. 55-A, 55-B, 55-C, 55-D, 55-E, 55-F, 55-G, 55-H, 55-I, 55-J, 55-K. 58-A, 58-B LEI 13.853, DE 08/07/2019: ALTERA A EMENTA, ART. 1º, 3º, 4º, 5º, 7º, 11, 18, 23, 26, 27, 29, 41, 52, 55-A, 55-B, 55-C, 55-D, 55-E, 55-F, 55-G, 55-H, 55-I, 55-J, 55-K, 55-L, 58-A, 58-B E 65. REVOGA §§ 1º E 2º DO ART. 7º MPV 959, DE 29/04/2020: ALTERA ART. 65 Vigência LEI 14.010, DE 10/06/2020: ACRESCE INCISO I-A AO ART. 6
Correlação:	
Veto:	Mensagem de veto: MSG 451, DE 14/08/2018 - DOU DE 15/08/2018, P. 75: VETO PARCIAL - PARTE VETADA: INCISO II DO § 1º DO ART. 26; ART. 28; INCISOS VII, VIII E IX DO ART. 52; ARTS. 55 AO 59.
Assunto:	CRITERIOS , TRATAMENTO , PROTEÇÃO , SEGURANÇA , SIGILO , DADOS PESSOAIS , PESSOA FISICA , PESSOA JURIDICA . ALTERAÇÃO , MARCO REGULATORIO , INTERNET , DIREITOS , USUARIO , ARMAZENAGEM , DADOS PESSOAIS , REGISTRO , EXCLUSÃO .
Classificação de direito:	DIREITOS E GARANTIAS FUNDAMENTAIS .
Observação:	---

Because Palácio do Planalto's website does not have an API, we explore the source-code for the record page of the legislation:

```
<html lang="pt-br" class="h-100">
<head>...</head>
<body class="d-flex flex-column h-100">
[...]
<div class="card mb-4 border-0 ">
        <div class="card-body">
                <ul class="list-group list-group-flush form-detalhe mt-4">
                        <li class="list-group-item border-0 p-0">
                                <div class="row">
                                        <div class="col-sm-2 label p-2">
                                                <h2>Data de assinatura:</h2>
                                        </div>
                                        <div class="col-sm bg-conteudo bg-secondary p-2 text-justify">
                                                14 de Agosto de 2018
                                        </div>
                                </div>
                        </li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">...</li>
                        <li class="list-group-item border-0 p-0">
                                <div class="row">
                                        <div class="col-sm-2 label p-2">
                                                <h2>Observação:</h2>
                                        </div>
                                        <div class="col-sm bg-conteudo bg-secondary p-2 text-justify">
                                                ---
                                        </div>
                                </div>
                        </li>
                </ul>
        </div>
</div>
[...]
</body>
</html>
```

The LeXML website has an API, acronym of "Application Programming Interface", a set of routines and patterns established to make machine requests in the websites functionalities, such as search pages, show their metada etc. A website with an API is a very good start to fetch metadata. If your target website has an API, you can follow the methodology adopted fot the LeXML website and scrape the desired data, using the same or different tools and languages. Here, we used Python, with some helpfull libraries. All the source code for this script is available in GitHub and is thoroughly explained in the Scripts section of the WikiProject page. The steps implemented and described in detail next are:

* Chose an item to test against the API;
* Study how the information is presented and which metadata you want to scrape;
* Study the API parameters (this is important to generalize and fetch a batch of items);
* Implement a script function to access, read, scrape and write the information you want;
* Run the script.

Chose an item to test against the API

It is good to chose an item that you have some familiarity, but also that have a good amount of the properties the website presents. Examples of the API itself are quite helpfull, as they are usually thought to present the majority of the information available. One of the items selected was the Law No. 13709 of August 14, 2018, illustrated earlier.

How the informations is presented

As shown in page 11, the API result is a XML (Extensible Markup Language) page with the information nested into named tags. Depending on the format that the information is returned, different libraries have to be used in a script to properly use it. In our case, a package of the lxml Python library was used.

Once you have the XML retrieved information set into a variable, you can navigate inside this tree and retrieve all the metadata present. In our case, all the desired metadata are available inside the named tags "srw_dc:dc", that have "srw:record" → "srw:recordData" → "srw_dc:dc" as their path inside the "srw:records" list of distinct results. Is the content of "srw_dc:dc" that we analyse to determine which metadata we want to scrape. In our case, the LeXML project has a series of controled vocabulary for authorities, localities, keywords and document types, so it was not of our interest to gather information about the "friendly" label to the controlled terms, hence why we excluded those tags from the list.

The API parameters

When available, the documentation of the API is the place to go to understand the API parameters. Usually, it is possible to find at least one example of the API working properly. The LeXML API parameters we set to fetch the metadata of one or multiple items were very intuitive, as the good practices of building APIs recommend.

The API url is a concatenation of the base URL of the API and the parameters with their values. The base URL for the LeXML API is "https://www.lexml.gov.br/busca/SRU" and the parameters we were able to set are "operation", "query", "maximumRecords" and "startRecord". The first parameter, "operation", declares the API function we will be using; The next parameter, "query", declares the terms we will be searching; The parameter "maximumRecords" is self explanatory, it declares the number of results per page and the last parameter, "startRecord", declares the number of the record in which to begin showing the results.

If we were planning to retrieve only one result from the API, the "maximumRecords" parameter would be equal to 1, and the "startRecord" wouldn't be necessary. Otherwise, if we were planning to scrape all the results, "maximumRecords" would be 500 (that is the limit the API imposes) and, if the number of results were more than 500, after the the first request to the API, "startRecord" would be set to be multiples of 500.

Implement and run the script

The structure of each website is different, hence why the scripts will always be different from one to another. But the logic set here can be applied.

In this script, we want to scrape a lot of information, around 28K items, so, because the API limits its results in batches of 500, we need to *create a loop* to access the result page of every 500 batch by modifying the "startRecord" parameter; Inside this loop, for each of the 500 results, we need to access every tag inside the "srw_dc:dc" branch and store the tag name and the value into a variable. Finally, we write this variable into a file. This file can be a csv, a txt file or whatever format fits you best.

Whithin the scripts, one can create other functions, not related directly to the API, that facilitate the workflow. This is done in this script mainly to format values or print them. The subjects metadata, for example, is a list of keywords with a consistent method of separation (a dot "." or a comma "," character); When reading the content, the code interprets the value as one string of text, so we add some extra lines of code to split the content into a Python list of the keywords.

The Palácio do Planalto website has a Search mechanism that allows users to list the Brazilian legislation issued since the beginning of the 19th century. This Search mechanism does not rely on the URL to perform a query, so our script for the LeXML website's API could not be reused here. The scrapping method had to be another, entirely new, using different libraries and tools.

The website actively tries to block any machine from accessing its content. That is a scenario you might encounter when dealing with websites not built thinking in the automated usage or scrapping (hence the abscence of an API). That obstacle made the scripts built to scrape this website far more slower than the LeXML. To contour this problem, we used other Python libraries and tools. All the source code for this script is available in GitHub and is thoroughly explained in the Scripts section of the WikiProject page.

The information we really wanted was the metadata about the the legislation on the website planalto.gov.br, that stores the full text of the legislation and a record with its metadata. The record page's URL uses a hash parameter, and the legislacao.presidencia.gov.br was the only place where this parameter was somehow listed. The steps implemented and described in detail next are:

* Open a Selenium WebDriver;

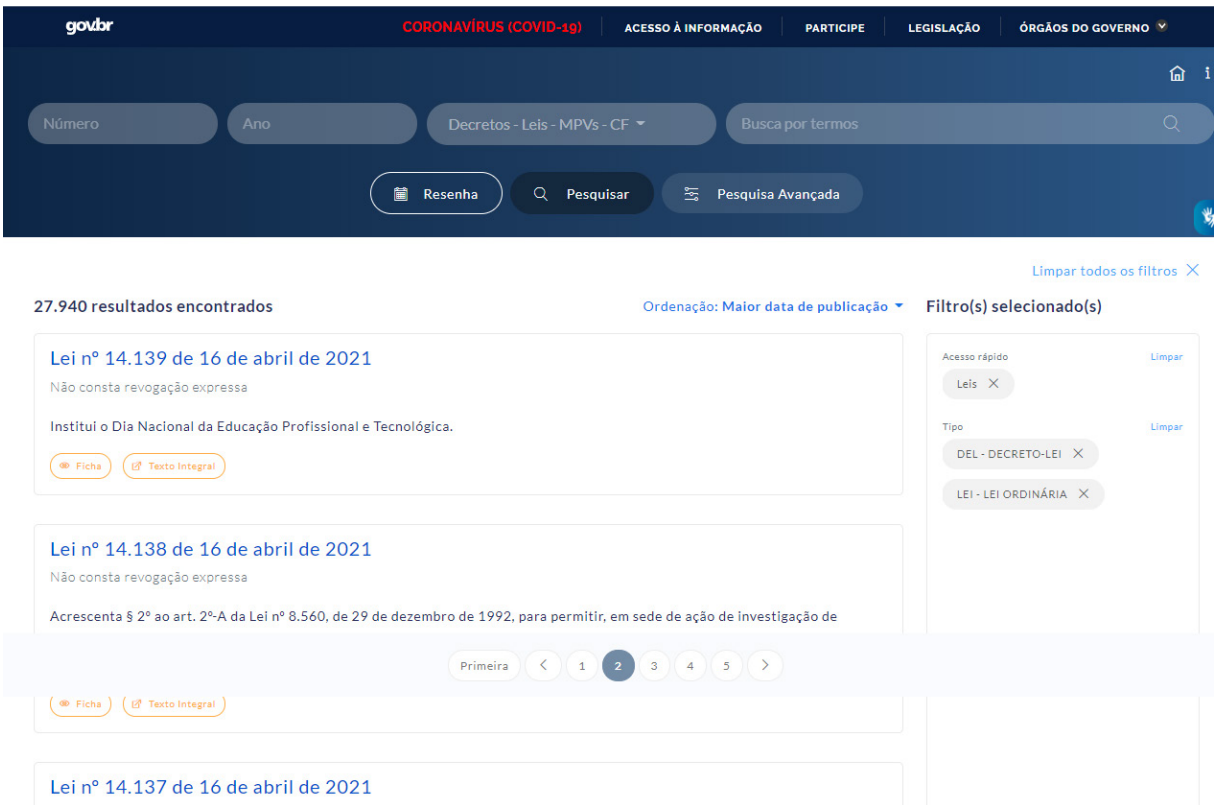
* Go to the page of the Search mechanism and manually define the query parameters;

* Implement a script function to navigate through the source code, extract and write the relevant information and then run the script.

Open a Selenium WebDriver

Selenium is a portable framework for testing web applications, and has been used to perform "human-like" actions in websites for a long time. It is good to chose an item that you have some familiarity, but also that have a good amount of the properties the website presents. Examples of the API itself are quite helpfull, as they are usually thought to present the majority of the information available. One of the items selected was the Law No. 13709 of August 14, 2018, illustrated earlier.

Screenshot of Legislação Federal Brasileira website



Screenshot of Legislação Federal Brasileira website with filters

Define the query parameters, implement and run the script

The Selenium WebDriver has functions to fetch and navigate through the elements of a webpage. One can find all elements based on the HTML tag of the elements, or their attributes, classes, etc. These functions simulate the human actions on the website, but if the structure of the elements are too complex, or if the website crashes the script due to delays in the loading, you can manually execute some actions and then let the script run. This was our case. The website always crashed after some thousand items scrapped, and we had to stop the script, adjust the query, and then run it again, so we could start scrapping from where it crashed, and not from the beginning.

On this website, the path was to click on "Pesquisa avançada" (*Advanced Search*) and filter laws and decre-laws. For this script, as for the others, we used an web-based interactive computational enviroment called Jupyter Notebook, so we could execute each part of the code at a time. That was necessary especially due to the manual intervention on the scrapping of the pages. The second part of the script is more machine-centered.

Once the results were loaded, the script would navigate through the elements of the page, get the name of the law (always inside a *h4* tag), the link to the full text in the Palácio do Planalto website and the link to the record page of the legislation there (both being the *href* attribute of an *a* tag inside a *unordered list* tag (*ul*) for each of the 10 elements shown in each page).

Troubleshooting

Sometimes, even with the implementation of pauses between actions on the script, the page takes a little longer to load due to a oscilation on the internet signal, or other reason that might break the script. For these cases, we stop the script, check for the results file and, based on the title of the legislation, we fetch a date parameter to adjust the query from the last item written. When running the script once again, we adjust the parameters manually and continue to scrape the information from where we were. Some duplicate checking might be necessary.

As said before, the website, although presenting a query based URL, requires a hash parameter named "*ato*". This, in practice, makes it impossible to, even with some of the metadata we already knew from LeXML (namelly tipe of item, its number and its year), access the record page for the legislation item, as the information the hash encodes is unidirectional.

With this information scraped using the Legislação scrapper, we can finally begin to scrape all the metadata available in the record page. For that we still use the Selenium methodology, as this is the same website and will require that a CAPTCHA be filled. The process used here is very similar to the LeXML script, with the difference that here we use Selenium.

The structure of the record page is very simple: All metadata name is inside a *h2* tag, and its value is the content of the next *div* tag after this *h2* tag; The name of the legislation item is always inside a *h1* tag. All the metadata values are formatted and stored in a custom variable, that is latter printed into a file. All the source code for this script is available in GitHub and is thoroughly explained in the Scripts section of the WikiProject page. To reproduce this type of script for your website, you need to study how the information is organized in it and then build the loops and other functions to extract the information you want.

Reconcile the results from the scrapers

LeXML and Palácio do Planalto legislation websites do not have a common identifier. However, the metadata scrapped in one need to be matched into the other. In our project, the parameters that were commons to both databases were the number and the date of the legislation item, both present in its title. This match was made by concatenating such values in a *DD-MM-YYYY;number* format in Google Sheets and comparing the results using the *VLOOKUP* formula. This didn't always work due to discrepancies between the title of the legislation items in both databases. In these cases, we did the matching manually. Once the matching was done, the spreadsheets of metadata from each website could be merged into one for the wikidatification.

Wikidatify the metadata

Once we obtained a spreadsheet with textual metadata values, we can begin a process of translation of the metadata to a Wikidata machine-readable format, called *wikidatification*. This process is nothing but the reconciliation between textual values and Wikidata entities (QIDS). This can be done in many ways. One can use tools and add-ons to find correspondent items in Wikidata such as OpenRefine or Google Sheets + Wikipédia and Wikidata Tools add-on. Items with no correspondence were marked as so and latter researched and created on Wikidata.

Perhaps the most complicated metadata to be wikidatified is the keyword of a legislation. Because of the inconsistency and wide range of values, we decided to do only the most used terms and implement the reconciliation of keywords into the activities of the project. The tab that shows the reconciliation table is available in the project page on Wikidata under the name of "Lexicon".

With all the relevant metadata wikidatified, QuickStatements commands were created in the spreadsheet and then uploaded to Wikidata using the QuickStatements tool. Created by Magnus Manske, QuickStatements is a helpfull tool to import information into Wikidata using simple commands. The commands involve the wikidatified values of the properties defined in the Ontology section of this document and are available in the Lexicon tab on the project page.

Workflow of the data. From left to right: Sources databases, spreadsheet and wikidatification softwares and Wikidata.

The Wikidata Query Service (WDQS) is a powerful tool that allows us to query and view information about items on Wikidata. This helps us to understand the impact that this project has had, as well as to have some insights into the legal history of Brazil.

Before the beggining of this project and the importation of the Brazilian legislation into Wikidata, there were less than 200 of this type of items registered on Wikidata. To give some perspective on this number, the country with the most law items on Wikidata is the United Kingdom, with over 134 thousand laws, followed by Ireland, with over 37 thousand. As of April, 2021, Brazil jumped from the bottom of the top 20 countries with the most laws on Wikidata, to being the third, with almost 28 thousand laws.

Many analysis are possible with this tool and the data imported into Wikidata. In the chart below, we can see the evolution of the Brazilian legislation in Wikidata and the impact of the activities of this project in its completness there. The proportion between laws and decree-laws and other types of legislation is so high that it is not even possible to clearly see their values, all due to the upload of these items into Wikidata in late February, 2021.

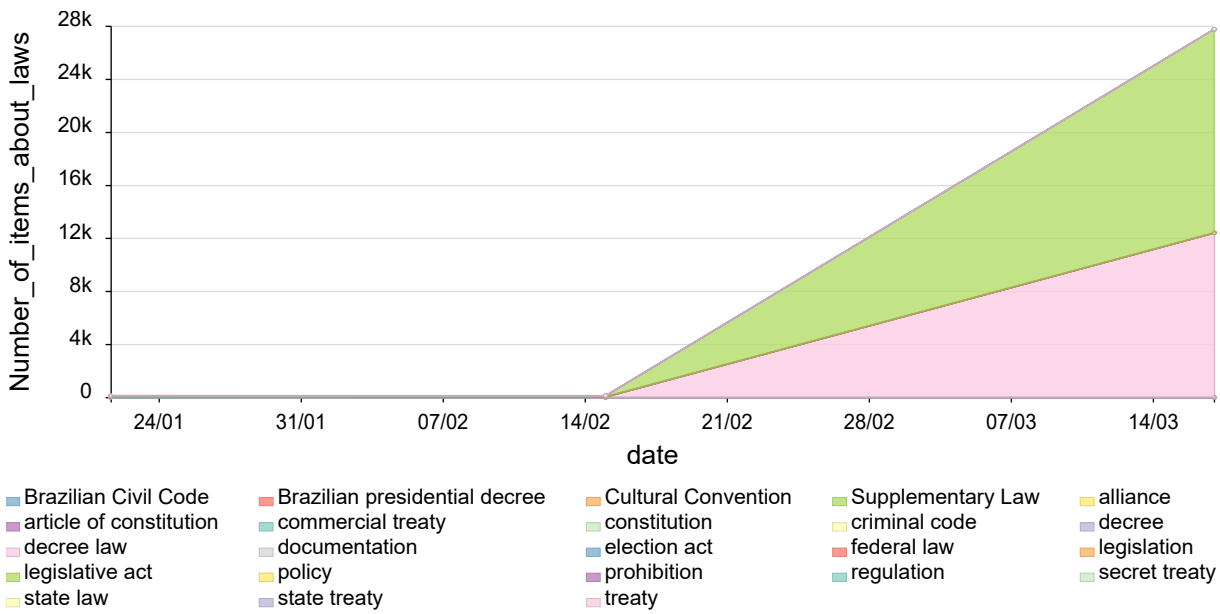
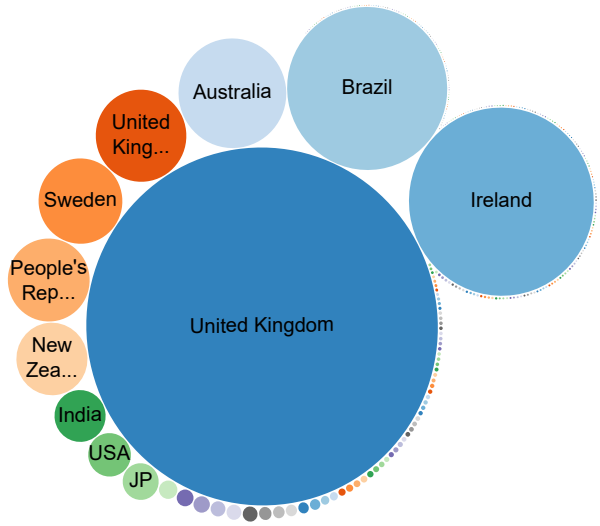
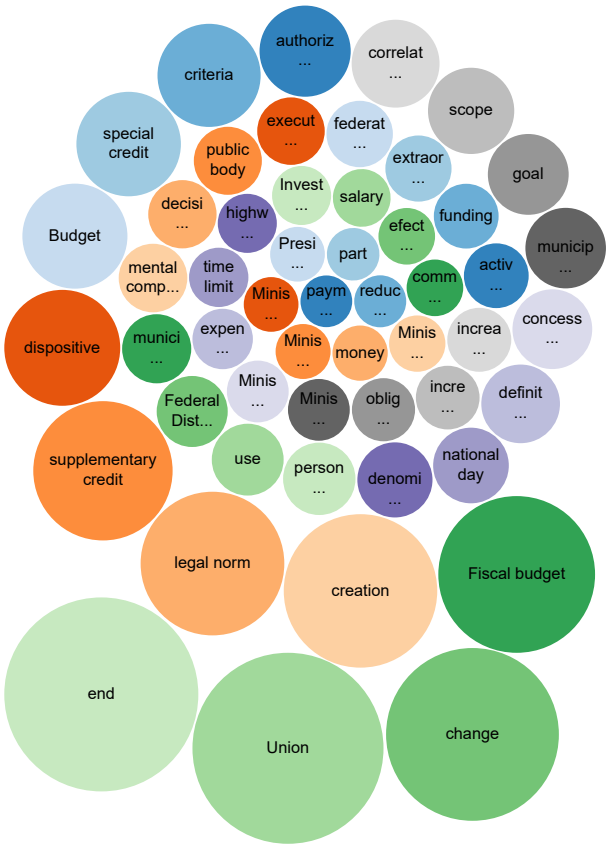


Chart of the number of Brazilian legislation items on Wikidata (Jan-Mar, 2021)



Bubble chart showing the number of legislation items per country in Wikidata (Apr, 2021)



Bubble chart showing the most used keywords of Brazilian legislation items in Wikidata (Apr, 2021)

Analysing the keywords of the Brazilian legislation it is possible to build a profile of topics with most concern in the country at an especific year or decade. For instance, is there a correlation between new legislation citing credit-related terms and periods of financial crisis? What are the main focus of legislation in the beginning and ending of a legislature? Questions and queries like these could bring further insights over historical periods and events; That information, if not present on Wikidata, wouldn't be possible to see with the tools available today.

To learn how to create charts like these, you can search for SPARQL documentation on the internet of follow the Wikidata Query Service Tutorial, a detailed guide on the Wikidata Query Service, developed by Wikimedia Israel. Help visualize the data contributed into Wikidata is fundamental to understand the data itself, specially considering that there is no tool or website publicly available where one could query and visualize this public information.

Organization

Grupo de usuários
WMB
Wiki Movimento Brasil

Support

{ } wikicite