

INTRODUCING

# KEGGutils

Filippo Maria Castelli  
Master Student in Physics  
Applied Physics Curriculum

[filippocastelli42@gmail.com](mailto:filippocastelli42@gmail.com)



[flippocastelli/KEGGutils](https://github.com/flippocastelli/KEGGutils)



# WHAT IS KEGGUTILS ?

AND HOW CAN WE USE IT



LET'S SUPPOSE WE WANT TO  
WORK WITH DATA COMING  
FROM **KEGG**

- ▶ WE NEED A RELIABLE AND QUICK  
WAY TO INTERFACE KEGG'S **REST  
API**
- ▶ WE NEED TO ORGANIZE SUCH DATA  
INTO A **WORKABLE FORMAT**



KEGG

Search

Help

## KEGG: Kyoto Encyclopedia of Genes and Genomes

KEGG is a database resource for understanding high-level functions and utilities of the biological system, such as the cell, the organism and the ecosystem, from molecular-level information, especially large-scale molecular datasets generated by genome sequencing and other high-throughput experimental technologies. See [Release notes](#) (April 1, 2019) for new and updated features.

### Main entry point to the KEGG web service

**KEGG2**

[KEGG Table of Contents](#) [\[Update notes\]](#) [\[Release history\]](#)

### Data-oriented entry points

<b>KEGG PATHWAY</b>	KEGG pathway maps
<b>KEGG BRITE</b>	BRITE hierarchies and tables
<b>KEGG MODULE</b>	KEGG modules
<b>KEGG ORTHOLOGY</b>	KO functional orthologs <a href="#">[Annotation]</a>
<b>KEGG GENOME</b>	Genomes <a href="#">[Pathogen]</a> <a href="#">[Virus]</a> <a href="#">[Plant]</a>
<b>KEGG GENES</b>	Genes and proteins <a href="#">[SeqData]</a>
<b>KEGG COMPOUND</b>	Small molecules
<b>KEGG GLYCAN</b>	Glycans
<b>KEGG REACTION</b>	Biochemical reactions <a href="#">[RModule]</a>
<b>KEGG ENZYME</b>	Enzyme nomenclature
<b>KEGG NETWORK</b>	Disease-related network elements
<b>KEGG DISEASE</b>	Human diseases <a href="#">[Cancer]</a>
<b>KEGG DRUG</b>	Drugs <a href="#">[New drug approvals]</a>
<b>KEGG MEDICUS</b>	Health information resource <a href="#">[Drug labels search]</a>

### Classification

Pathway  
Brite  
Brite table  
Module  
KO (Function)  
Organism  
Compound  
Network  
Disease (ICD)  
Drug (ATC)  
Drug (Target)



KEGG'S API EXPOSES A FEW **COMMANDS** VIA **GET** REQUESTS

- ▶ THEY ALL COME IN THE FORM

`http://rest.kegg.jp/<operation>/<argument>`

- ▶ POSSIBLE OPERATIONS ARE

- ▶ **INFO** DISPLAYS INFORMATIONS ON A SELECTED DATABASE
- ▶ **LIST** OBTAINS A LIST OF ENTRY IDENTIFIERS
- ▶ **FIND** FINDS ENTRIES MATCHING A KEYWORD
- ▶ **GET** RETRIEVES A GIVEN DATABASE ENTRY
- ▶ **CONV** CONVERTS KEGG IDENTIFIERS TO/FROM OUTSIDE IDENTIFIERS
- ▶ **LINK** FINDS RELATED ENTRIES WITH CROSS-REFERENCES
- ▶ **DDI** FINDS ADVERSE DRUG-DRUG INTERACTIONS

AND THE RESPONSES CAN BE

- ▶ RAW TEXT
- ▶ IMAGES (.GIF OR .PNG)
- ▶ XML FILES
- ▶ CONFIGURATION FILES
- ▶ JSON FILES

MANUALLY HANDLING HTTP REQUESTS AND RESPONSES  
SIGNIFICANTLY SLOWS DOWN EVERY KIND OF WORKFLOW AS  
ONE WOULD HAVE TO:

- ▶ MANUALLY COMPOSE AN HTTP URL
- ▶ SEND THE REQUEST
- ▶ WAIT FOR THE RESPONSE, CATCH AND DECODE IT
- ▶ FIND A WAY TO PARSE PLAIN TEXT, XML OR JSON
- ▶ ORGANIZE OBTAINED DATA IN A MEANINGFUL FORMAT



KEGGUTILS DOES ALL OF THAT FOR YOU, AND MORE.

KEGGUTILS AIMS TO OFFER A DROP-IN SOLUTION

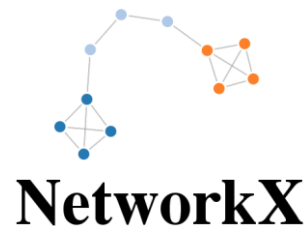
IT'S ENTIRELY  
WRITTEN IN PYTHON



ALL THE HTTP  
WORKLOAD IS  
POWERED BY  
REQUESTS



AUTOMATIC DATA  
ORGANIZATION IN A  
CUSTOM NETWORKX  
COMPATIBLE  
FORMAT



HOW DO WE **INSTALL** IT?

- ▶ **KEGGUTILS** IS AVAILABLE AS A **PYPI** PACKAGE AND IT'S INSTALLABLE WITH A ONE-LINE **PIP** COMMAND

```
> pip install KEGGutils
```





ALTERNATIVELY

YOU CAN CLONE THE OFFICIAL REPO FROM [GITHUB](#)

```
> git clone https://github.com/filippocastelli/KEGGutils
```

**GitHub**



**git**

IN THE [REPO](#) YOU WILL FIND

- ▶ EXTENSIVE [TUTORIALS](#) ON HOW TO USE KEGGUTILS
- ▶ ALL THE EXTERNAL [REFERENCES](#) YOU MAY WANT
- ▶ AND, OF COURSE, [KEGGUTILS](#)'S CODE

# A QUICK OVERVIEW OF SOME FEATURES

PART 1:  
KEGG REST API  
INTERFACE

LET'S START **SIMPLE**: HOW TO GET **INFOS** ON A **DATABASE**

```
In[1]: import KEGGutils as kg
```

```
In[2]: kg.__version__
```

```
Out[2]: '0.3.2'
```

NOW THAT WE'RE SURE THE **LATEST** VERSION IS INSTALLED, LET'S TRY TO USE **KEGGUTILS** TO INTERFACE THE API'S **INFO COMMAND**

```
In [3]: kg.keggapi_info("hsa")
```

```
INFO:root:Infos on hsa from KEGG:
```

```
T01001 Homo sapiens (human) KEGG Genes Database
```

```
hsa Release 90.0+/04-03, Apr 19
```

```
Kanehisa Laboratories
```

```
38,683 entries
```

```
linked db pathway [...]
```

ALL THE OTHER **API COMMANDS** ARE **INTERFACED**, BUT  
THAT'S JUST **SCRAPING THE SURFACE**

WE CAN TRY TO GET **SPECIFICATIONS** ON A PARTICULAR KEGG ENTRY  
USING THE **KEGGAPI\_GET()** INTERFACE

```
In[1]: kg.keggapi_get("hsa:10458")
```

Infos on hsa:10458 from KEGG:

ENTRY	10458	CDS	T01001
NAME	BAIAP2, BAP2, FLAF3, IRSP53		
DEFINITION	(RefSeq) BAI1 associated protein 2		
ORTHOLOGY	K05627 BAI1-associated protein 2		
ORGANISM	hsa Homo sapiens (human)		
PATHWAY	hsa04520 Adherens junction		
	hsa04810 Regulation of actin cytoskeleton		
BRITE	KEGG Orthology (KO) [BR:hsa00001]		
	09140 Cellular Processes		
	09144 Cellular community - eukaryotes		
	04520 Adherens junction [...]		

ALL THE OTHER [API COMMANDS](#) ARE [INTERFACED](#), BUT  
THAT'S JUST [SCRAPING THE SURFACE](#)

OR WHY DON'T WE SEARCH FOR CONNECTIONS BETWEEN THE  
[HUMAN GENE](#) DATABASE AND THE [ENZYME](#) DATABASE

```
In[1]: sn,tn = kg.keggapi_link(source = "hsa",target=
"enzyme")
```

```
In[2]: list(zip(sn,tn))[:5]
```

```
[('hsa:9344', 'ec:2.7.11.1'),
('hsa:5894', 'ec:2.7.11.1'),
('hsa:673', 'ec:2.7.11.1'),
('hsa:5607', 'ec:2.7.12.2'),
('hsa:5598', 'ec:2.7.11.24')]
```

BUT THIS KIND OF DATA FORMAT IS NOT REALLY [READY-TO-USE](#): LET'S  
INTRODUCE [KEGGGRAPHS](#)

# A QUICK OVERVIEW OF SOME FEATURES

PART 2:  
KEGGGRAPHS,  
KEGGLINKGRAPHS,  
KEGGCHAINS AND  
KEGGPATHWAYS

KEGGUTILS PROVIDES CLASSES TO FOR KEGG DATA STORAGE AND USAGE

- ▶ **KEGGGRAPH** IS THE MAIN **GRAPH** STORAGE CLASS  
OTHER CLASSES DERIVE FROM IT:
  - ▶ **KEGGLINKGRAPH**
    - ▶ USEFUL TO DEAL WITH CROSSLINKS BETWEEN DATABASES
  - ▶ **KEGGCHAIN**
    - ▶ DATA DISCOVERY-ORIENTED CLASS TO CREATE CHAINS OF SEQUENTIAL CROSSLINKS
  - ▶ **KEGGPATHWAY**
    - ▶ DESIGNED TO SPECIFICALLY DEAL WITH PATHWAYS



KEGGUTILS PROVIDES CLASSES TO FOR KEGG DATA STORAGE AND USAGE

ALL THESE CLASSES SHARE NETWORKX.GRAPH INHERITANCE:

- ▶ THEY CAN BE USED AS NORMAL NETWORKX GRAPHS
- ▶ YOU'RE NOT LIMITED TO KEGGUTILS TO ANALYZE YOUR DATA
- ▶ EXPANDED FUNCTIONALITIES RESPECT TO VANILLA NETWORKX.GRAPHS

CREATING A **BIPARTITE** GRAPH DESCRIBING THE  
**CROSSLINKS** BETWEEN TWO KEGG DATABASE IS **SIMPLE**

AND CAN BE DONE IN **ONE LINE**:

```
In[1]: dis_gene = kg.KEGGlinkgraph(source_db = "disease",  
                                     target_db = "hsa")  
  
INFO:root:> Downloading enzyme-hsa-link from KEGG at  
http://rest.kegg.jp/link/hsa/disease INFO:root:succesfully  
downloaded disease-hsa-link  
  
In[2]: dis_gene.linked_nodes["ds:H00773"][:5]  
  
['hsa:55777',  
'hsa:81704',  
'hsa:1013',  
'hsa:84623',  
'hsa:8831']
```

WE CAN CHAIN MULTIPLE KEGGLINKGRAPHS TO SEARCH FOR NONTRIVIAL LINKS

CREATING A CHAIN TAKES NO EFFORT

```
In[1]: mychain = kg.KEGGchain(chain = ["disease", "hsa",  
"enzyme", "reaction"])
```

WE CAN SELECT A SUBSET OF NODES AT THE TOP AND AND LOOK DOWN THE CHAIN TO OBSERVE WHICH NODES ARE SEQUENTIALLY CONNECTED TO THEM

```
In[1]: mychain.directed_propagation(["ds:H00773"])
```

YOU CAN [DOWNLOAD](#), [PARSE](#) AND [EXPLORE](#) AN ENTIRE PATHWAY  
JUST BY SPECIFYING ITS [KEGG IDENTIFIER](#)

```
In[1]: pathway = kg.KEGGpathway( pathway_id = "hsa05215")  
In[2]: print(pathway.title)  
Prostate cancer
```

AND EVEN SEARCH FOR LINKED [ACADEMIC PUBLICATIONS](#) WITH JUST  
A SINGLE [COMMAND](#)

```
In[1]: references = pathway.get_references()  
In[2]: references[0]  
  
{'reference_hook': 'PMID:12878745',  
'authors': 'Nelson WG, De Marzo AM, Isaacs WB.',  
'title': 'Prostate cancer.',  
'journal': 'DOI:10.1056/NEJMra021562'}
```

YOU CAN FIND **ALL** OF THIS AND **MUCH MORE** ON KEGGUTIL'S TUTORIALS:

<https://github.com/filippocastelli/KEGGutils/tree/master/tutorials>

VISIT THE PROJECT'S **REPO** AT

<https://github.com/filippocastelli/KEGGutils/>

