

Ensembl REST API

16th February 2021

09:30 - 16:30 (GMT) - end time revised to accommodate timetable

Instructors: [Michal Szpak](#)

Helper: Aleena Mushtaq

Event Timetable: <https://training.csx.cam.ac.uk/bioinformatics/event/3748722>

All timings are approximate

09:30 - 10:30 Overview of the Ensembl REST API and GET queries

10:30 - 10:40 Tea/Coffee Break

10:40 - 12:30 Scripting around REST API calls

12:30 - 13:30 Lunch (not provided)

13:30 - 15:00 Content types and chaining REST queries together

15:00 - 15:50 Accessing POST queries with the Ensembl REST API

15:50 - 16:15 Rate-limiting

16:15 - 16:30 Wrap-up and questions

Links

- Please connect to the course via this **Zoom link**:
<https://us02web.zoom.us/j/87699336982?pwd=bXU0SVFFZTZITkJOaWtEMGhkTXZyUT09>
- **Course notes and Jupyter Notebooks website**:
http://training.ensembl.org/events/2021/2021-02-16-CamREST_Feb
- **Ensembl helpdesk**: helpdesk@ensembl.org
- **Course participant introductions**:
<https://docs.google.com/document/d/1J4tSh04XY25FhVvCuQv24Ox7rmndll1sQmDGp7UGDKk/edit?usp=sharing>
- **Course feedback survey**: <https://forms.gle/d4P7EWC8U5NR38mTA>

We would really appreciate it if you could share your thoughts with us regarding these sessions. We are interested in your opinions, how you feel the experience has benefited you and how it could be improved.

If you could find a few minutes to complete a short survey at the end of the last session it would really help us in improving the training we can deliver.

Further actions:

- [Sign up to our mailing list to get notifications of upcoming courses from the Bioinformatics Training Facility](#)

Further Reading:

www.ensembl.org - Ensembl genome browser (chordates)

grch37.ensembl.org - Ensembl archive for browsing data associated with the human GRCh37 genome assembly

www.ensemblgenomes.org - Ensembl Genomes genome browser (non-chordates)

www.ebi.ac.uk - EMBL-EBI website

www.ebi.ac.uk/services - EMBL-EBI databases and tools

[Train Online](#) - EMBL-EBI e-learning for our databases and tools

EMBL-EBI provides an e-learning platform called [Train online](#). Train online provides free courses on Europe's most widely used data resources, created by experts at EMBL-EBI and collaborating institutes. You do not need to have any previous experience of bioinformatics to benefit from this training. We want to help you to become a confident user of our data resources; we are not trying to train you to become a bioinformatician.

Tutorials of interest might include:

- [Ensembl: Quick Tour](#)
- [Ensembl: Browsing Genomes](#)
- [Ensembl Genomes \(non-chordates\): Quick Tour](#)
- [Ensembl REST API](#)

The EMBL-EBI training team also runs a regular [webinar series](#) featuring the EBI resources. See the [Training pages](#) for more information. You can also catch up on any webinars that you might have missed in [Train online](#) or on [Youtube](#).

Recordings

A link to a full recording of this session will be added here shortly after the end of the session. This is being made available to help you reinforce your learning after the event and is not designed for general distribution, we kindly ask that you limit sharing of this link to your fellow attendees of the webinar.

Zoom Recording:

https://us02web.zoom.us/rec/share/BZKHKx6LQWuf1n-R9O5ZY-xOs0dkHBFc4CNCm97GJQtx01MzqdAHT5ArJ6hJLPDH.b8Xe_1IPW8iMXUT7 Passcode: ZD\$*Q3i6

Note that the link will remain active for approximately 1 month although you will be able to download a permanent copy.

Course netiquette

We are expecting a large number of participants in this session. We suggest everyone follows [these few simple rules](#) for the course to run as smoothly as possible.

Questions

If you have any questions/problems that you would like to share and are applicable to the whole class please write them below. A tutor will answer your question.

Write your question after the last one you can see in this document and write your name.

1. **<Maulik>** How do I know that there is an option like “content-type”? Because there is no such option given here:
<http://rest.ensembl.org/documentation/info/lookup>
<Michal> <content-type is a special parameter, which might be included in the URL, but most of the times will be included in the request header - we'll get to it later. Standard parameters are listed in the documentation pages.>
2. **<HW>** <what is GET and POST? Any other similar commands exist for the REST api?>
<Michal> <We'll talk about it later today. They are different HTTP methods and yes, there are other methods too (e.g. DELETE to delete content), but you won't need them to work with Ensembl REST API>
3. **<HW>** <This may be a lazy question: if I wanna download all TFBS or TSS sites information using Ensembl REST API, how could that be done?(e.g. Which endpoint I should use, etc...) >
<Michal> <If you're after whole-genome scale data, then I'd suggest using our FTP site (<http://www.ensembl.org/info/data/ftp/index.html>) and download whole genome flat files. We have ready files with such information - REST API is not the best way of doing it.>
4. **<Yasemin>** <I may have missed this at the beginning, but how would I save a copy of the course Jupityer notebook on my own laptop? (I've already saved the Google collaborative documents.)>
<Mical> <Sadly they're only meant to run in the Google colab. You cannot save them to your laptop.>
5. **<Alexandre>** <You briefly mentioned MySQL, what would be its use related to Ensembl?>
<Michal> <
>

It allows access to all Ensembl's data at larger scale. REST API doesn't have a complete coverage - we have many endpoints but there's more data in our databases which could be accessed directly via MySQL. This access isn't rate limited which means you can get larger data more efficiently. Have a look here for more information:

<http://www.ensembl.org/info/data/mysql.html>

6. <Name><

Ensembl REST API course

REST calls in the browser

Exercises 1

Making requests with Python

Exercises 2

Exercises 2 – answers

Using results

Exercises 3

Exercises 3 – answers

Other content types

Exercises 4

Exercises 4 – answers

Linking endpoints together

Exercises 5

Exercises 5 – answers

Using POST

Optional parameters

Exercises 6

Exercises 6 – answers

Rate limiting

Exercises 7 – answers

Section

```
print ("Variant\tP-value\tPub-med ID\tRisk allele\tGene")

# define the general URL parameters
server = "http://rest.ensembl.org/"
ext_phen = "/phenotype/term/homo_sapiens/coffee_consumption?"
con = "application/json"

# submit the query
get_phen = fetch_endpoint(server, ext_phen, con)

for variant in get_phen:
    id = variant['Variation']
    pv = str(variant['attributes'].get('p_value'))
    pmid = variant['attributes'].get('external_reference')
    risk = str(variant['attributes'].get('risk_allele'))
    gene = variant['attributes'].get('associated_gene')

    print (id + "\t" + pv + "\t" + pmid + "\t" + risk + "\t" + gene)
```

Variant	p-value	Pub-med ID	Risk allele	Gene
rs17685	4.00e-11	PMID:25288136	A	POR, SNORA14A, TMEM120A
rs17144566	3.00e-6	PMID:31046077	C	RAB30-AS1
rs1503212	3.00e-6	PMID:31046077	A	GRID2
rs17145750	2.00e-10	PMID:31837886	T	MLXIPL
rs329124	3.00e-6	PMID:31046077	A	JADE2

```
KeyError: 'associated_gene'
Traceback (most recent call last):
  <ipython-input-2-06fe3706d718> in <module>()
    30     pmid = variant['attributes'].get('external_reference')
    31     risk = str(variant['attributes'].get('risk_allele'))
--> 32     gene = variant['attributes'].get('associated_gene')
    33
    34     print (id + "\t" + pv + "\t" + pmid + "\t" + risk + "\t" + gene)
```

KeyError: 'associated_gene'

SEARCH STACK OVERFLOW

KeyError: 'associated_gene', this error occurs in exercise 3 answer, not sure what is the problem? Probably some associated_gene are blank in the variant ? >

<Michal>Yes, that's correct, seems some associated_gene values are null, put it in the get() and change to string: gene = str(variant['attributes'].get('associated_gene'))>

7. <Maulik>In case of 1 to many homologous, will the output have many sequences per species? Yes, thanks

<Michal>Yes, the second level nested list would contain one dictionary per homology, each dictionary would have a 'target' (mouse) and 'source' (human) sequence

8. <Maulik> I got bad url error (refer to the picture below): HTTPError: 400 Client Error: Bad Request for url:

<http://rest.ensembl.org/overlap/region/human/1:6423776-6462367>

9.

```
from pprint import pprint

def fetch_endpoint(server,request,content_type):

    gStart=0
    gEnd=0
    gStrand=""

    r=requests.get(server+request,headers={"Accept": content_type})

    if not r.ok:
        r.raise_for_status()
        sys.exit()
    if content_type=="application/json":
        gStart=int(r.json()["start"])
        gEnd=int(r.json()["end"])
        gStrand=r.json()["strand"]
        gChrom=r.json()["seq_region_name"]
        return gStart,gEnd,gStrand,gChrom
    else:
        return r.text

gene_name="ESPN"
server="http://rest.ensembl.org/"
ext_get_lookup="lookup/symbol/homo_sapiens/"+gene_name+"?";
con="application/json"

genStart,genEnd,genStrand,gChrom=fetch_endpoint(server,ext_get_lookup,con)

if genStrand==1:
    genUp=genStart-1000
    genDown=genEnd+1000
else:
    genUp=genEnd+1000
    genDown=genStart-1000

ext_get_lookup="overlap/region/human/"+str(gChrom)+":"+str(genUp)+"-"+str(genDown)+"?";
con="application/x-text"
tmpOut=fetch_endpoint(server,ext_get_lookup,con)
pprint(tmpOut)
```

Thanks, I found the errors.

<Michal><I'm glad you found the answer. It seems to be a typo in the URL indicated by the 400 status code>