

Requesting data from Entrez in different formats

We can request data as text or as XML

We can request data as text or as XML

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                       retmode="text")

gb_file_contents = handle.read()
handle.close()

print(gb_file_contents)
```

We can request data as text or as XML

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                      retmode="text")
gb_file_contents = handle.read()
handle.close()
```

```
print(gb_file_contents)
```

```
LOCUS          KT220438                1701 bp    cRNA     linear   VRL 20-JUL-2015
DEFINITION     Influenza A virus (A/NewJersey/NHRC_93219/2015(H3N2)) segment 4
                hemagglutinin (HA) gene, complete cds.
ACCESSION     KT220438
VERSION       KT220438.1  GI:887493048
KEYWORDS      .
SOURCE        Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
   ORGANISM   Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
                Viruses; ssRNA viruses; ssRNA negative-strand viruses;
                Orthomyxoviridae; Influenzavirus A.
REFERENCE     1  (bases 1 to 1701)
   AUTHORS    Sitz,C.R., Thammavong,H.L., Balansay-Ames,M.S., Hawksworth,A.W.,
```

We can request data as text or as XML

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                      retmode="xml")

gb_file_contents = handle.read()
handle.close()
```

```
print(gb_file_contents)
```

```
<?xml version="1.0" ?>
<!DOCTYPE GBSet PUBLIC "-//NCBI//NCBI GBSeq/EN"
"https://www.ncbi.nlm.nih.gov/dtd/NCBI_GBSeq.dtd">
<GBSet>
<GBSeq>
  <GBSeq_locus>KT220438</GBSeq_locus>
  <GBSeq_length>1701</GBSeq_length>
  <GBSeq_strandedness>single</GBSeq_strandedness>
  <GBSeq_moltype>cRNA</GBSeq_moltype>
  <GBSeq_topology>linear</GBSeq_topology>
  <GBSeq_division>VRL</GBSeq_division>
  <GBSeq_update-date>20-JUL-2015</GBSeq_update-date>
  <GBSeq_create-date>20-JUL-2015</GBSeq_create-date>
  <GBSeq_definition>Influenza A virus (A/NewJersey/NHRC_93219/2015(H3N2))
```

Pros and cons of text and XML

Text:

- Easier to read for humans
- Requires special parser for each datatype

XML:

- Nearly impossible to read for humans
- Can be parsed with a generic parser

We parse text format with `SeqIO.read()`

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                      retmode="text")
record = SeqIO.read(in_handle, format="gb") # use SeqIO.read() to parse
handle.close()
```

We parse text format with `SeqIO.read()`

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                      retmode="text")
record = SeqIO.read(in_handle, format="gb") # use SeqIO.read() to parse
handle.close()
```

```
print(record)
```

```
ID: KT220438.1
Name: KT220438
Description: Influenza A virus (A/NewJersey/NHRC_93219/2015(H3N2)) segment 4
hemagglutinin (HA) gene, complete cds.
Number of features: 5
/data_file_division=VRL
/date=20-JUL-2015
/accessions=['KT220438']
/sequence_version=1
/keywords=[]
/source=Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
/organism=Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))
/taxonomy=['Viruses', 'ssRNA viruses', 'ssRNA negative-strand viruses',
'Orthomyxoviridae', 'Influenzavirus A']
/references=[Reference(title='GEISS Influenza Surveillance Response Program',
...), Reference(title='Direct Submission', ...)]
```


We parse XML format with `Entrez.parse()`

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                       retmode="xml")
parsed = Entrez.parse(in_handle) # use Entrez.parse() to parse
record = list(parsed)[0] # Need to convert into list and get 1st element
handle.close()
```

We parse XML format with Entrez.parse()

```
handle = Entrez.efetch(db="nucleotide", id="KT220438", rettype="gb", \
                        retmode="xml")

parsed = Entrez.parse(in_handle) # use Entrez.parse() to parse
record = list(parsed)[0] # Need to convert into list and get 1st element
handle.close()

print(record) # Record contains nested dictionaries and lists

{'GBSeq_locus': 'KT220438', 'GBSeq_length': '1701', 'GBSeq_strandedness':
'single', 'GBSeq_moltype': 'cRNA', 'GBSeq_topology': 'linear',
'GBSeq_division': 'VRL', 'GBSeq_update-date': '20-JUL-2015', 'GBSeq_create-
date': '20-JUL-2015', 'GBSeq_definition': 'Influenza A virus
(A/NewJersey/NHRC_93219/2015(H3N2)) segment 4 hemagglutinin (HA) gene, complete
cds', 'GBSeq_primary-accession': 'KT220438', 'GBSeq_accession-version':
'KT220438.1', 'GBSeq_other-seqids': ['gb|KT220438.1|', 'gi|887493048'],
'GBSeq_source': 'Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))',
'GBSeq_organism': 'Influenza A virus (A/New Jersey/NHRC_93219/2015(H3N2))',
'GBSeq_taxonomy': 'Viruses; ssRNA viruses; ssRNA negative-strand viruses;
Orthomyxoviridae; Influenzavirus A', 'GBSeq_references':
[{'GBReference_reference': '1', 'GBReference_position': '1..1701',
'GBReference_authors': ['Sitz,C.R.', 'Thammavong,H.L.', 'Balansay-Ames,M.S.',
'Hawksworth,A.W.', 'Myers,C.A.', 'Brice,G.T.'], 'GBReference_title': 'GEISS
Influenza Surveillance Response Program', 'GBReference_journal':
```

All information from parsed XML format can be accessed using dict & list methods

```
# extract all the features
```

```
features = record['GBSeq_feature-table']
```

```
# print feature key & location for all features
```

```
for feature in features:
```

```
    print(feature['GBFeature_key'] + ": " + \
          feature['GBFeature_location'])
```

```
source: 1..1701
```

```
gene: 1..1701
```

```
CDS: 1..1701
```


```
mat_peptide: 49..1035
```

```
mat_peptide: 1036..1698
```

All information from parsed XML format can be accessed using dict & list methods

```
# extract all the features
features = record['GBSeq_feature-table']

# print feature key & location for all features
for feature in features:
    print(feature['GBFeature_key'] + ": " + \
          feature['GBFeature_location'])
```




```
source: 1..1701
gene: 1..1701
CDS: 1..1701
mat_peptide: 49..1035
mat_peptide: 1036..1698
```

All information from parsed XML format can be accessed using dict & list methods

```
# extract all the features
features = record['GBSeq_feature-table']

# print feature key & location for all features
for feature in features:
    print(feature['GBFeature_key'] + ": " + \
          feature['GBFeature_location'])

source: 1..1701
gene: 1..1701
CDS: 1..1701
mat_peptide: 49..1035
mat_peptide: 1036..1698
```



Running searches through Entrez

Example: Literature search using pubmed



US National Library of Medicine
National Institutes of Health

PubMed

wilke co

Search

Create RSS Create alert Advanced

Help

Article types

Clinical Trial
Review
Customize ...

Text availability

Abstract
Free full text
Full text

PubMed Commons

Reader comments
Trending articles

Publication dates

5 years
10 years
Custom range...

Species

Humans
Other Animals

[Clear all](#)

[Show additional filters](#)

Format: Summary ▾ Sort by: Most Recent ▾ Per page: 20 ▾

Search results

Items: 1 to 20 of 125

<< First < Prev Page 1 of 7 Next > Last >>

- [Biophysical Models of Protein Evolution: Understanding the Patterns of Evolutionary Sequence Divergence.](#)

Echave J, **Wilke CO.**

Annu Rev Biophys. 2017 Mar 15. doi: 10.1146/annurev-biophys-070816-033819.

[Epub ahead of print]

PMID: 28301766

[Similar articles](#)

- [Accelerated simulation of evolutionary trajectories in origin-fixation models.](#)

Teufel AI, **Wilke CO.**

J R Soc Interface. 2017 Feb;14(127). pii: 20160906. doi: 10.1098/rsif.2016.0906.

PMID: 28228542

[Similar articles](#)

Send to ▾

Filter your results:

All (125)

[Free Full Text \(96\)](#)

[Review \(7\)](#)

[Manage Filters](#)

Find related data

Database:

Select

Find items

Search details

wilke co[Author]

Search

See more

Example: Literature search using pubmed

```
handle = Entrez.esearch(db="pubmed", # database to search
                        term="Wilke CO", # search term
                        retmax=5)      # max. number of results
record = Entrez.read(handle)
handle.close()

# search returns PubMed IDs (pmids)
pmid_list = record["IdList"]
print(pmid_list)

['28301766', '28228542', '27834632', '27713835', '27535929']
```


We retrieve search results with `efetch()`

```
# For references, the file format is called "Medline"
from Bio import Medline

handle = Entrez.efetch(db="pubmed", id=pmid_list,
                      rettype="medline", retmode="text")
records = Medline.parse(handle)
# Must not close handle yet!

for record in records:
    print(record['AU']) # author list
    print(record['TI']) # title
    print(record['SO']) # source (reference)
    print()
handle.close() # Close after all records have been processed
```

We retrieve search results with `efetch()`

```
['Echave J', 'Wilke CO']
```

```
Biophysical Models of Protein Evolution: Understanding the Patterns of  
Evolutionary Sequence Divergence.
```

```
Annu Rev Biophys. 2017 Mar 15. doi: 10.1146/annurev-biophys-070816-033819.
```

```
['Teufel AI', 'Wilke CO']
```

```
Accelerated simulation of evolutionary trajectories in origin-fixation models.
```

```
J R Soc Interface. 2017 Feb;14(127). pii: 20160906. doi:
```

```
10.1098/rsif.2016.0906.
```

```
['Lipsitch M', 'Barclay W', 'Raman R', 'Russell CJ', 'Belser JA', 'Cobey S',  
'Kasson PM', 'Lloyd-Smith JO', 'Maurer-Stroh S', 'Riley S', 'Beauchemin CA',  
'Bedford T', 'Friedrich TC', 'Handel A', 'Herfst S', 'Murcia PR', 'Roche B',  
'Wilke CO', 'Russell CA']
```

```
Viral factors in influenza pandemic risk assessment.
```

```
Elife. 2016 Nov 11;5. pii: e18491. doi: 10.7554/eLife.18491.
```

```
['McWhite CD', 'Meyer AG', 'Wilke CO']
```

```
Sequence amplification via cell passaging creates spurious signals of positive  
adaptation in influenza virus H3N2 hemagglutinin.
```

```
Virus Evol. 2016 Jul;2(2). pii: vew026. Epub 2016 Oct 3.
```

```
['Spielman SJ', 'Wan S', 'Wilke CO']
```

```
A Comparison of One-Rate and Two-Rate Inference Frameworks for Site-Specific  
dN/dS Estimation.
```