CHEATSHEET PYTHON 6: Biopython

Sequence Input/Output

To read and write sequence or alignment data, enter the lines

`from Bio import SeqIO` OR `from Bio import AlignIO`

Command	Description	Example
`SeqIO.read()`	read a single sequence from a file	`SeqIO.read("seq.fasta", "fasta")
`SeqIO.parse()`	read many sequences from a file	`SeqIO.parse("seqs.fasta", "fasta")
`AlignIO.read()`	read a single alignment from a file	`SeqIO.read("align.fasta", "fasta")
`AlignIO.parse()`	read many alignments from a file	`SeqIO.parse("aligns.phy", "phylip")
`SeqIO.write()`	write sequence(s) to a file	`SeqIO.write(seq_record(s), "seq_output.fasta", "fasta")
`AlignIO.write()`	write alignment(s) to a file	`AlignIO.write(alignment, "align_output.fasta", "fasta")

Biopython objects

To use and/or manipulate these objects, enter the lines

```
from Bio.Seq import Seq
from Bio.SeqRecord import SeqRecord
from Bio.Alphabet import *
```

Object	Useful attributes/methods
`SeqRecord`	`.id`, `.seq`, `.description`
`Seq`	`.transcribe()`, `.translate()`, `.complement()`, and more..

Example script that makes a list of sequence names:

```
seqIDs=[]
for record in SeqIO.parse(open(trinity_file,'rU'),'fasta'):
    seqIDs.append(record.id)
```

Example script that alters the description of a SeqRecord and prints them to a file:

```
fastaList=[]
counter=0 # this could be a dictionary that you want to pull information from
for rec in SeqIO.parse(open(trinity_file, 'rU'),'fasta'):
    rec.description = seq.description+' '+counter # or pull info via dictionary[rec.id]
    fastaList.append(rec) #copy record to a list
    counter+=1
with open(output_file, 'w') as f:
    SeqIO.write(fastaList, f, "fasta")
```

Collect NCBI Data

To download from NCBI, enter the line `from Bio import Entrez`. To avoid annoying warnings, also enter the command, `Entrez.email = "myemail@email.com"` (replaced with your email) before fetching!

Command	Description	Example
`Entrez.efetch()`	Fetch a record from the NCBI database (the type of id is up to you!)	`Entrez.efetch(id = "NP_000549.1", db = "protein", remode = "text", retype = "gb")` `Entrez.efetch(id = "88758587", db = "nucleotide", remode = "text", retype = "gb")`