Create Fasta files of a set number of sequences of the GH7 family

Use sequences downloaded to the to the database cazycsj5

Imports

```
from Bio import SeqIO
from Bio.SeqRecord import SeqRecord
import random
from saintBioutils.utilities.file_io import make_output_directory
from pathlib import Path
```

Load in Data

```
infile = "../Data/GH7seqs/GH7.fasta"
```

Create Dictionary

Use SeqIO to look through the fasta sequences and write each to a dictionary. The key is the sequence ID and the value is the sequence itself.

```
newdict = {} # Create an empty dictionary called newdict.
with open(infile, "r") as fh: # Opens the GH7 fasta file in read mode.
    for record in SeqIO.parse(fh, "fasta"): # Cycle through each
record in the file
        newdict[f"{record.id}"] = record.seq # Write the record to the
dictionary.
```

```
len(newdict.keys()) # Check length of the dictionary
```

```
10221
```

Create file of a specified number of random sequences from GH7

Function creates fasta file of randomly selected GH7 sequences. The number of sequences is chosen as 'nseqs'.

For the file containing 100 sequences, each entry is written to its own file and stored in Sourmash_input.

```
def randomselect(nseqs, dictionary): # Function is called
randomselect.
    """Function randomly picks 'nseqs' number of sequences from the
GH7 dictionary and writes them to a
    new fasta file. Then it creates invidudule fasta file for each
sequence for analysis in Sourmash """
    newlist = random.choices(list(dictionary.keys()), k=nseqs)
    # Create list of 'nseqs' number of randomly selected GH7
    # sequences.
    print(f'selected {len(newlist)} sequences')
    newseqs = [] # Create empty list called newseqs.
    for name in newlist:
```

```
        seq = dictionary[name]
        record = SeqRecord(seq, id=name)
        newseqs.append(record)
    print(f'2: selected {len(newseqs)} sequences')
    SeqIO.write(newseqs, f"../Data/{nseqs}seqs.fasta", "fasta")
    # Write the records in the list newseqs to a fasta file stored in
    #Data
# Build Sourmash input files
    output_dir = Path(f'../Data/Sourmash/sourmash_{nseqs}') # Explain
where the output is to be stored.
    make_output_directory(output_dir, True, False)
    # Create an output folder in the correct storage location which
overwrites
    # anything that is already there.
    for record in newseqs: # Loop to go though the list of chosen
sequences.
        filepath = output_dir/f"{record.id}.fasta"
        # Tells the new files to go to
Data/Sourmash_input/<sequenceID>.fasta.
        with open(filepath, "w") as ohandle: # Open the folder and
files in write mode.
            SeqIO.write([record],ohandle, "fasta") # Write the new
file to the folder.
    return record

selected = randomselect(10000, newdict) # Run function for a sequence
size of 10000,
selected = randomselect(5000, newdict) # 5000,
selected = randomselect(3000, newdict) # 3000,
selected = randomselect(1000, newdict) # 1000,
selected = randomselect(500, newdict) # 500,
selected = randomselect(300, newdict) # 300,
selected = randomselect(100, newdict) # 100.

help(randomselect) # Call functions doc string to explain what the
function does.
```

```
Output directory ../Data/Sourmash/sourmash_10000 exists, nodelete is
False. Deleting content currently in output directory.

selected 10000 sequences
2: selected 10000 sequences
selected 5000 sequences
2: selected 5000 sequences

Output directory ../Data/Sourmash/sourmash_5000 exists, nodelete is
False. Deleting content currently in output directory.
Output directory ../Data/Sourmash/sourmash_3000 exists, nodelete is
False. Deleting content currently in output directory.
```

```
selected 3000 sequences
2: selected 3000 sequences

Output directory ../Data/Sourmash/sourmash_1000 exists, nodelete is
False. Deleting content currently in output directory.

selected 1000 sequences
2: selected 1000 sequences

Output directory ../Data/Sourmash/sourmash_500 exists, nodelete is
False. Deleting content currently in output directory.
Output directory ../Data/Sourmash/sourmash_300 exists, nodelete is
False. Deleting content currently in output directory.

selected 500 sequences
2: selected 500 sequences
selected 300 sequences
2: selected 300 sequences

Output directory ../Data/Sourmash/sourmash_100 exists, nodelete is
False. Deleting content currently in output directory.

selected 100 sequences
2: selected 100 sequences
Help on function randomselect in module __main__:

randomselect(nseqs, dictionary)
    Function randomly picks 'nseqs' number of sequences from the GH7
dictionary and writes them to a
    new fasta file. Then it creates invidudule fasta file for each
sequence for analysis in Sourmash
```