

Bursa Teknik Üniversitesi

Mühendislik ve Doğa Bilimleri Fakültesi

Biyoinformatiğe Giriş Dersi Ödev Raporu

Installation:

```
!sudo apt-get update -y
!sudo apt-get install python3.6.5
!python --version
```

```
13s ▶ !apt-get install python3.6

import os
os.environ['PYTHONPATH']="/usr/bin/python3.6"

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libpython3.6-minimal libpython3.6-stdlib python3.6-minimal
Suggested packages:
  python3.6-venv binfmt-support
The following NEW packages will be installed:
  libpython3.6-minimal libpython3.6-stdlib python3.6 python3.6-minimal
0 upgraded, 4 newly installed, 0 to remove and 23 not upgraded.
Need to get 4,294 kB of archives.
After this operation, 22.1 MB of additional disk space will be used.
Get:1 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal/main amd64 libpython3.6-minimal amd64 3.6.15-1+focal3 [569 kB]
Get:2 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal/main amd64 python3.6-minimal amd64 3.6.15-1+focal3 [1,718 kB]
Get:3 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal/main amd64 libpython3.6-stdlib amd64 3.6.15-1+focal3 [1,758 kB]
Get:4 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu focal/main amd64 python3.6 amd64 3.6.15-1+focal3 [248 kB]
Fetched 4,294 kB in 4s (1,213 kB/s)
Selecting previously unselected package libpython3.6-minimal:amd64.
(Reading database ... 128276 files and directories currently installed.)
Preparing to unpack .../libpython3.6-minimal_3.6.15-1+focal3_amd64.deb ...
Unpacking libpython3.6-minimal:amd64 (3.6.15-1+focal3) ...
Selecting previously unselected package python3.6-minimal.
Preparing to unpack .../python3.6-minimal_3.6.15-1+focal3_amd64.deb ...
Unpacking python3.6-minimal (3.6.15-1+focal3) ...
Selecting previously unselected package libpython3.6-stdlib:amd64.
Preparing to unpack .../libpython3.6-stdlib_3.6.15-1+focal3_amd64.deb ...
Unpacking libpython3.6-stdlib:amd64 (3.6.15-1+focal3) ...
Selecting previously unselected package python3.6.
Preparing to unpack .../python3.6_3.6.15-1+focal3_amd64.deb ...
Unpacking python3.6 (3.6.15-1+focal3) ...
Setting up libpython3.6-minimal:amd64 (3.6.15-1+focal3) ...
Setting up python3.6-minimal (3.6.15-1+focal3) ...
Setting up libpython3.6-stdlib:amd64 (3.6.15-1+focal3) ...
Setting up python3.6 (3.6.15-1+focal3) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for mime-support (3.64ubuntu1) ...
```

```
kali@kali: ~
File Actions Edit View Help

~python3.6: command not found

(kali@kali)-[~]
└─$ sudo apt-get update

[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
Get:1 http://kali.download/kali kali-rolling InRelease [41.2 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [19.4 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [45.0 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [116 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [172 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [217 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [927 kB]
Fetched 65.9 MB in 49s (1,344 kB/s)
Reading package lists... Done
```

```
kali@kali: ~
File Actions Edit View Help

└─$ python --version

Python 3.10.5

(kali@kali)-[~]
└─$ pip --version

pip 22.2 from /usr/lib/python3/dist-packages/pip (python 3.10)

(kali@kali)-[~]
└─$ sudo apt-get install python-pip

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python-pip is not available, but is referred to by another package
This may mean that the package is missing, has been obsoleted, or
is only available from another source
```

✓ [12] !python --version
0
sn.

Python 3.9.16

`pip install biopython==1.72`

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting biopython==1.72
  Downloading biopython-1.72.tar.gz (16.4 MB)
    16.4/16.4 MB 13.1 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages (from biopython==1.72) (1.22.4)
Building wheels for collected packages: biopython
  Building wheel for biopython (setup.py) ... done
  Created wheel for biopython: filename=biopython-1.72-cp39-cp39-linux_x86_64.whl size=2415635 sha256=fc2215ffa390534a512918a7c0086797
  Stored in directory: /root/.cache/pip/wheels/0f/bf/cc/bf2ef82b6d793a873c0e0191ec0b0a9be4edb496a11bf304f3
Successfully built biopython
Installing collected packages: biopython
  Attempting uninstall: biopython
    Found existing installation: biopython 1.81
    Uninstalling biopython-1.81:
      Successfully uninstalled biopython-1.81
Successfully installed biopython-1.72
```

✓ [12] `pip install biopython==1.72`
0
sn.

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: biopython==1.72 in /usr/local/lib/python3.9/dist-packages (1.72)
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages (from biopython==1.72) (1.22.4)
```

✓ `import Bio`
0
sn. `print(Bio.__version__)`

1.72

```
(kali@kali)-[~]
$ sudo apt-get install python-pip
```

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python-pip is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
However the following packages replace it:
python3-pip
```

E: Package 'python-pip' has no installation candidate

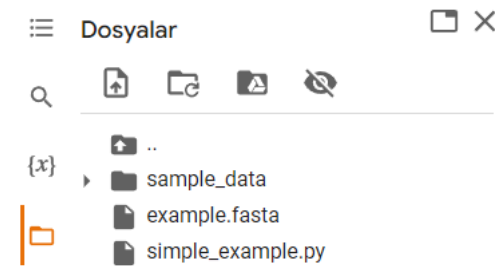
```
(kali@kali)-[~]
$ pip install biopython==1.72
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting biopython==1.72
  Downloading biopython-1.72.tar.gz (16.4 MB)
    16.4/16.4 MB 1.4 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: numpy in /usr/lib/python3/dist-packages (from biopython==1.72) (1.21.5)
Building wheels for collected packages: biopython
  Building wheel for biopython (setup.py) ... done
  Created wheel for biopython: filename=biopython-1.72-cp310-cp310-linux_x86_64.whl size=2374641 sha256=a2e11c5425f3bb6ad3bfce4f19cfa3924756a6a048e27cfd690bf01393dd1f77
  Stored in directory: /home/kali/.cache/pip/wheels/ac/1d/a4/6b6fa7bf61d9220cf66b33f8d6c22c25005d3d4db42ac460d4
Successfully built biopython
Installing collected packages: biopython
Successfully installed biopython-1.72
```

Creating Simple Application:

```
İçindekiler example.fasta x ...
1
2
3 >sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
4 MKLKKTIGAMALATL FATMGASAVEKTI SVTASVDPTVDLLQSDGSALPNSVALTYSPAV
5 NNFEAHTINTVVHTNDSKGVVVKLSADPVLSNVLNPTLQIPVSVNFAGKPLSTTGITID
6 SNDLNFASSGVNKSSTQKLSIHADATRVTTGGALTAGQYQGLVSIILTKSTTTTTTKGT
7
8 >sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
9 MLKIKYLLIGLSLSAMSSYS LAAAGPTLTKEALNVLSPAALDATWAPQDNLTLSNTGVS
10 NTLVGVLTLNNTSIDTVSIASNTVSDTSKNGTVTFAHETNNSASFATTISTDNANITLDK
11 NAGNTIVKTTNGSQLPTNLPLKFITTEGNEHLVSGNYRANITITSTIKGGGTTKGTDDKK
```

```
~/Desktop/example.fasta - Mousepad
File Edit Search View Document Help
simple_example.py x example.fasta x
1
2
3 >sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
4 MKLKKTIGAMALATL FATMGASAVEKTI SVTASVDPTVDLLQSDGSALPNSVALTYSPAV
5 NNFEAHTINTVVHTNDSKGVVVKLSADPVLSNVLNPTLQIPVSVNFAGKPLSTTGITID
6 SNDLNFASSGVNKSSTQKLSIHADATRVTTGGALTAGQYQGLVSIILTKSTTTTTTKGT
7
8 >sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
9 MLKIKYLLIGLSLSAMSSYS LAAAGPTLTKEALNVLSPAALDATWAPQDNLTLSNTGVS
10 NTLVGVLTLNNTSIDTVSIASNTVSDTSKNGTVTFAHETNNSASFATTISTDNANITLDK
11 NAGNTIVKTTNGSQLPTNLPLKFITTEGNEHLVSGNYRANITITSTIKGGGTTKGTDDKK
12
```



```
İçindekiler example.fasta x ...
1
2
3 >sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
4 MKLKKTIGAMALATL FATMGASAVEKTI SVTASVDPTVDLLQSDGSALPNSVALTYSPAV
5 NNFEAHTINTVVHTNDSKGVVVKLSADPVLSNVLNPTLQIPVSVNFAGKPLSTTGITID
6 SNDLNFASSGVNKSSTQKLSIHADATRVTTGGALTAGQYQGLVSIILTKSTTTTTTKGT
7
8 >sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
9 MLKIKYLLIGLSLSAMSSYS LAAAGPTLTKEALNVLSPAALDATWAPQDNLTLSNTGVS
10 NTLVGVLTLNNTSIDTVSIASNTVSDTSKNGTVTFAHETNNSASFATTISTDNANITLDK
11 NAGNTIVKTTNGSQLPTNLPLKFITTEGNEHLVSGNYRANITITSTIKGGGTTKGTDDKK
```

Disk

İçindekiler example.fasta simple_example.py X

```
1 from Bio.SeqIO import parse
2 from Bio.SeqRecord import SeqRecord
3 from Bio.Seq import Seq
4
5 file = open("example.fasta")
6 records = parse(file, "fasta")
7
8 for record in records:
9     print("Id: %s" % record.id)
10    print("Name: %s" % record.name)
11    print("Description: %s" % record.description)
12    print("Annotations: %s" % record.annotations)
13    print("Sequence Data: %s" % record.seq)
14    print("Sequence Alphabet: %s" % record.seq.alphabet)
15
16
17
18
```

File Edit Search View Document Help

```
1 from Bio.SeqIO import parse
2 from Bio.SeqRecord import SeqRecord
3 from Bio.Seq import Seq
4
5 file = open("example.fasta")
6 records = parse(file, "fasta")
7
8 for record in records:
9     print("Id: %s" % record.id)
10    print("Name: %s" % record.name)
11    print("Description: %s" % record.description)
12    print("Annotations: %s" % record.annotations)
13    print("Sequence Data: %s" % record.seq)
14    print("Sequence Alphabet: %s" % record.seq.alphabet)
15
16
17
18
```

kali@kali: ~/Desktop

File Actions Edit View Help

\$ nano simple_example.py

```
(kali@kali)-[~/Desktop]
$ python simple_example.py
Id: sp|P25730|FMS1_ECOLI
Name: sp|P25730|FMS1_ECOLI
Description: sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
Annotations: {}
Sequence Data: MKLKKTTIGAMALATLFATMGASAVEKTSVTASVDPTVDLLQSDGSAIPNSVALTYSYPAVINFEAHTINTVVTND
SDGQVVKLSADPVLNVLNPTLQIPVSVNFAGKPLSTTGITIDSNDLNFASSGVNKVSSTQKLSIHADATRVGGALTAGQVQGLVSIIL
TKSTTTTTTTKGT
Sequence Alphabet: SingleLetterAlphabet()
Id: sp|P15488|FMS3_ECOLI
Name: sp|P15488|FMS3_ECOLI
Description: sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
Annotations: {}
Sequence Data: MLKIKYLLIGLSLAWSSYSYLAAGPTLTKEALNVLSPAALDATWAPQDNLTLNLTGVSNTLVGVLTLNLTSTIDT
VSIASNTNVSQTSKNGTVTFHAHETNNSASFATTISTDNANITLDKNAGNTIVKTTNGSQLPTNLPKFITTEGNEHLVSGNYRANITITSTI
KGGGTTKGGTTDKK
Sequence Alphabet: SingleLetterAlphabet()

(kali@kali)-[~/Desktop]
```

[45] !python simple_example.py

```
Id: sp|P25730|FMS1_ECOLI
Name: sp|P25730|FMS1_ECOLI
Description: sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
Annotations: {}
Sequence Data: MKLKKTTIGAMALATLFATMGASAVEKTSVTASVDPTVDLLQSDGSAIPNSVALTYSYPAVINFEAHTINTVVTNDSDGQVVKLSADPVLNVLNPTLQIPVSVNFAGKPLSTTGITIDSNDLNFASSGVNKVSSTQKLSIHADATRVGGALTAGQVQGLVSIILTKSTTTTTTTKGT
Sequence Alphabet: SingleLetterAlphabet()
Id: sp|P15488|FMS3_ECOLI
Name: sp|P15488|FMS3_ECOLI
Description: sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
Annotations: {}
Sequence Data: MLKIKYLLIGLSLAWSSYSYLAAGPTLTKEALNVLSPAALDATWAPQDNLTLNLTGVSNTLVGVLTLNLTSTIDTVSIASNTNVSQTSKNGTVTFHAHETNNSASFATTISTDNANITLDKNAGNTIVKTTNGSQLPTNLPKFITTEGNEHLVSGNYRANITITSTIKGGGTTKGGTTDKK
Sequence Alphabet: SingleLetterAlphabet()
```

```
file = open("example.fasta")
records = parse(file, "fasta")

for record in records:
    print("Id: %s" % record.id)
    print("Name: %s" % record.name)
    print("Description: %s" % record.description)
    print("Annotations: %s" % record.annotations)
    print("Sequence Data: %s" % record.seq)
    print("Sequence Alphabet: %s" % record.seq.alphabet)
```

Id: sp|P25730|FMS1_ECOLI
Name: sp|P25730|FMS1_ECOLI
Description: sp|P25730|FMS1_ECOLI CS1 fimbrial subunit A precursor (CS1 pilin)
Annotations: {}
Sequence Data: MKLKKITIGAMALATLFATMGASAVEKTIISVTASVDPTVDLLQSDGSALPNSVALTYSAPVNNFEAHTINTVHTNDSKGVVVKLSADPVLISNVLNPTLQIPVSVNFAGKPLSTTGITIDSNDLNFASSGVNKNVSSQKLSIHADATRVTGALTAGQYQGLVSIILTKSTTTTTTTTGGT
Sequence Alphabet: SingleLetterAlphabet()
Id: sp|P15488|FMS3_ECOLI
Name: sp|P15488|FMS3_ECOLI
Description: sp|P15488|FMS3_ECOLI CS3 fimbrial subunit A precursor (CS3 pilin)
Annotations: {}
Sequence Data: MLKIKYLLIGLSLSAMSSVSLAAAGPTLTKELALIVLSPAALDATHAPQDWLTLNHTGVSNITLVGVLTLSNITSIDTVSIASNVSDTSKNGTVTFAHETNWSASFATTISTDNANITLDKNAGNITVKTNGSQLPTNLPKFITTEGNEHLVSGHYRANITITSTIKGGGTTKGTDDKK
Sequence Alphabet: SingleLetterAlphabet()

Sequence:

```
kali@kali: ~  
File Actions Edit View Help  
25005d3d4db42ac460d4  
Successfully built biopython  
Installing collected packages: biopython  
Successfully installed biopython-1.72  
  
(kali@kali)-[~]  
$ python  
Python 3.10.5 (main, Jun 8 2022, 09:26:22) [GCC 11.3.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import Bio  
>>> Print(Bio.__version__)  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
NameError: name 'Print' is not defined. Did you mean: 'print'?  
>>> print(Bio.__version__)  
1.72  
>>> from Bio.Seq import Seq  
>>> seq=Seq('AGCT')  
>>>  
>>> print(seq)  
AGCT  
>>> from Bio.Seq import Seq  
>>> myseq =Seq("AGCT")  
>>> myseq  
Seq('AGCT')  
>>>  
>>> myseq.alphabet  
Alphabet()  
>>> from Bio.Seq import Seq  
>>> myseq =Seq("AGCT")  
>>> myseq  
Seq('AGCT')  
>>>
```



```
✓ 0 [26] from Bio.Seq import Seq  
sn. seq=Seq('AGCT')  
  
print(seq)  
  
AGCT
```

```
✓ 0 from Bio.Seq import Seq  
sn. myseq =Seq("AGCT")  
myseq  
  
myseq.alphabet  
  
Alphabet()
```

```
✓ 0 from Bio.Seq import Seq  
sn. myseq =Seq("AGCT")  
myseq  
  
Seq('AGCT')
```

Alphabet Module:

```
kali@kali: ~  
File Actions Edit View Help  
Alphabet()  
>>> from Bio.Seq import Seq  
>>> myseq =Seq("AGCT")  
>>> myseq  
Seq('AGCT')  
>>>  
>>> from Bio.Seq import Seq  
>>> from Bio.Alphabet import single_letter_alphabet  
>>> test_seq =Seq('AGTACACTGGT',single_letter_alphabet)  
>>> test_seq  
Seq('AGTACACTGGT', SingleLetterAlphabet())  
>>> from Bio.Seq import Seq  
>>> from Bio.Alphabet import generic_protein  
>>> test_seq =Seq('AGTACACTGGT', generic_protein)  
>>> test_seq  
Seq('AGTACACTGGT', ProteinAlphabet())  
>>> from Bio.Data import IUPACData  
>>> IUPACData.protein_letters  
'ACDEFGHIKLMNPQRSTVWY'  
>>> █
```

```

✓ 0 [32] from Bio.Seq import Seq
sn.   from Bio.Alphabet import single_letter_alphabet
      test_seq =Seq( 'AGTACACTGGT' ,single_letter_alphabet)
      test_seq
      #seq( ' AGTACACTGGT ',SingleLetterAlphabet ( ) )

```

```
Seq('AGTACACTGGT', SingleLetterAlphabet())
```

```

✓ 0 ▶ from Bio.Seq import Seq
sn.   from Bio.Alphabet import generic_protein
      test_seq =Seq('AGTACACTGGT', generic_protein)
      test_seq

```

```
Seq('AGTACACTGGT', ProteinAlphabet())
```

```

✓ 0 [36] from Bio.Data import IUPACData
sn.   IUPACData. protein_letters

```

```
'ACDEFGHIKLMNPQRSTVWY'
```

BASIC OPERATIONS:

```

✓ 0 [38] seq_string= Seq( "AGCTAGCT" )
sn.   seq_string[0]

```

```
'A'
```

```

kali@kali: ~
File Actions Edit View Help
>>> seq_string[0]
'A'
>>> seq_string= Seq( "AGCTAGCT" )
>>> seq_string[0]
'A'
>>> 

```

```

>>> seq_string[0:2]
Seq('AG')
>>> seq_string[0:2]
Seq('AG')
>>> 

```

✓ [43] seq_string[0:2]
0
sn.

Seq('AG')

✓ [42] seq_string[:]
0
sn.

Seq('AGCTAGCT')

✓ [25] len(seq_string)
0
sn.

8

✓ [26] seq_string.count('A')
0
sn.

2

```
>>> seq_string[ : ]  
Seq('AGCTAGCT')  
>>>  
>>> len(seq_string)  
8  
>>> seq_string.count('A')  
2  
>>> █
```

✓ [62] from Bio.Seq import Seq
0
sn. from Bio.Alphabet import generic_dna, generic_protein
seq1= Seq("AGCT", generic_dna)
seq2=Seq("TCGA", generic_dna)

seq1+seq2

Seq('AGCTTCGA', DNAAlphabet())

```
>>> from Bio.Seq import Seq  
>>> from Bio.Alphabet import generic_dna, generic_protein  
>>> seq1= Seq("AGCT", generic_dna)  
>>> seq2=Seq( "TCGA", generic_dna)  
>>> seq1+seq2  
Seq('AGCTTCGA', DNAAlphabet())  
>>> █
```



```
0 sn. ▶ from Bio.Seq import Seq
from Bio.Alphabet import generic_dna, generic_protein
dna_seq = Seq('AGTACACTGGT', generic_dna)

protein_seq = Seq('AGUACACUGGU', generic_protein)
dna_seq + protein_seq

-----
TypeError                                Traceback (most recent call last)
<ipython-input-49-dcd1945ff8d6> in <module>
      4
      5 protein_seq = Seq('AGUACACUGGU', generic_protein)
----> 6 dna_seq + protein_seq

/usr/local/lib/python3.9/dist-packages/Bio/Seq.py in __add__(self, other)
    307         if not Alphabet._check_type_compatible([self.alphabet,
    308                                                other.alphabet]):
--> 309             raise TypeError(
    310                 "Incompatible alphabets {0!r} and {1!r}".format(
    311                     self.alphabet, other.alphabet))

TypeError: Incompatible alphabets DNAAlphabet() and ProteinAlphabet()
```

SEARCH STACK OVERFLOW

```
kali@kali: ~
File Actions Edit View Help
>>>
>>>
>>> from Bio.Seq import Seq
>>> from Bio.Alphabet import generic_dna, generic_protein
>>> dna_seq = Seq('AGTACACTGGT', generic_dna)
>>>
>>> protein_seq = Seq('AGUACACUGGU', generic_protein)
>>>
>>> dna_seq + protein_seq
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/home/kali/.local/lib/python3.10/site-packages/Bio/Seq.py", line 309, in __add__
    raise TypeError(
TypeError: Incompatible alphabets DNAAlphabet() and ProteinAlphabet()
>>> █
```

```
0 sn. ▶ from Bio .Alphabet import generic_dna
list =[Seq( "AGCT" , generic_dna) , Seq( "TCGA" , generic_dna) , Seq( "AAA" , generic_dna) ]
for s in list:
    print(s)
```

AGCT
TCGA
AAA

```

>>>
>>>
>>> from Bio.Alphabet import generic_dna
>>> list=[Seq( "AGCT" , generic_dna) , Seq( "TCGA" , generic_dna)
>>> , Seq( "AAA" , generic_dna) ]
>>> for s in list:
...     print(s)
...
AGCT
TCGA
AAA
>>>
>>>

```

```

✓ 0 sn.
▶ final_seq =Seq(" ",generic_dna)
  for s in list:
    final_seq = final_seq + s
  final_seq

Seq(' AGCTTCGAAAA', DNAAlphabet())

```

```

kali@kali: ~
File Actions Edit View Help
>>>
>>>
>>>
>>> final_seq =Seq(" ",generic_dna)
>>> for s in list:
...     final_seq = final_seq + s
...
>>> final_seq
Seq(' AGCTTCGAAAA', DNAAlphabet())
>>>
>>>

```

```

✓ 0 sn.
▶ from Bio.Alphabet import generic_rna
  rna=Seq("agct",generic_rna)

  rna.upper()

Seq('AGCT', RNAAlphabet())

```

```

kali@kali: ~
File Actions Edit View Help
>>>
>>> from Bio.Alphabet import generic_rna
>>> rna=Seq("agct",generic_rna)
>>>
>>> rna.upper()
Seq('AGCT', RNAAlphabet())
>>>
>>>
>>>
>>>

```

```

✓ 0 sn.
[72] rna=Seq("agct",generic_rna)
     'a' in rna

```

True

```

✓ 0 sn.
[73] 'A' in rna

```

False

```

✓ 0 sn.
▶ rna1=Seq("agct",generic_rna)
  rna is rna1

```

False

```

kali@kali: ~
File Actions Edit View Help
>>>
>>>
>>> rna=Seq("agct",generic_rna)
>>> 'a' in rna
True
>>> 'A' in rna
False
>>> rna1=Seq("agct",generic_rna)
>>> rna is rna1
False
>>>

```

```
✓ 0 sn. [82] protein_seq =Seq( 'AGUACACUGGU' ,generic_protein)
protein_seq.find('G')
```

1

```
✓ 0 sn. ▶ protein_seq.find('g')
-1
```

```
✓ 0 sn. [81] protein_seq.find('GG')
8
```

Burada kendim denemek
istedim çıktığı merak ettiğim
için

```
kali@kali: ~
File Actions Edit View Help
>>>
>>>
>>> protein_seq =Seq( 'AGUACACUGGU' ,generic_protein)
>>> protein_seq.find('G')
1
>>>
>>>
```

```
kali@kali: ~
File Actions Edit View Help
>>>
>>> protein_seq.find('G')
1
>>>
>>> protein_seq.find('g')
-1
>>>
>>> protein_seq.find('GG')
8
>>>
```

```
✓ 0 sn. ▶ protein_seq =Seq( 'AGUACACUGGU', generic_protein)
protein_seq. split( 'A' )
```

```
📄 [Seq('', ProteinAlphabet()),
Seq('GU', ProteinAlphabet()),
Seq('C', ProteinAlphabet()),
Seq('CUGGU', ProteinAlphabet())]
```

```
kali@kali: ~
File Actions Edit View Help
>>>
>>>
>>> protein_seq =Seq( 'AGUACACUGGU', generic_protein)
>>>
>>> protein_seq. split( 'A' )
[Seq('', ProteinAlphabet()), Seq('GU', ProteinAlphabet()),
Seq('C', ProteinAlphabet()), Seq('CUGGU', ProteinAlphabet())]
>>>
>>>
```

✓
0
sn.

```
[91] strip_seq =Seq("    AGCT    ")
```

```
strip_seq
```

```
Seq('    AGCT    ')
```

✓
0
sn.



```
strip_seq.strip()
```

```
Seq('AGCT')
```

```
File Actions Edit View Help
>>>
>>>
>>> strip_seq =Seq("    AGCT    ")
>>> strip_seq
Seq('    AGCT    ')
>>>
>>>
>>>
```

AÇIKLAMA: Ödev Google-colab (beyaz ekran görüntüleri) ve sanal makina kali linux (siyah ekran görüntüleri) kullanılarak hazırlanmıştır.

Hazırlayan:

HÜMEYRA ÇİMEN

19360859053