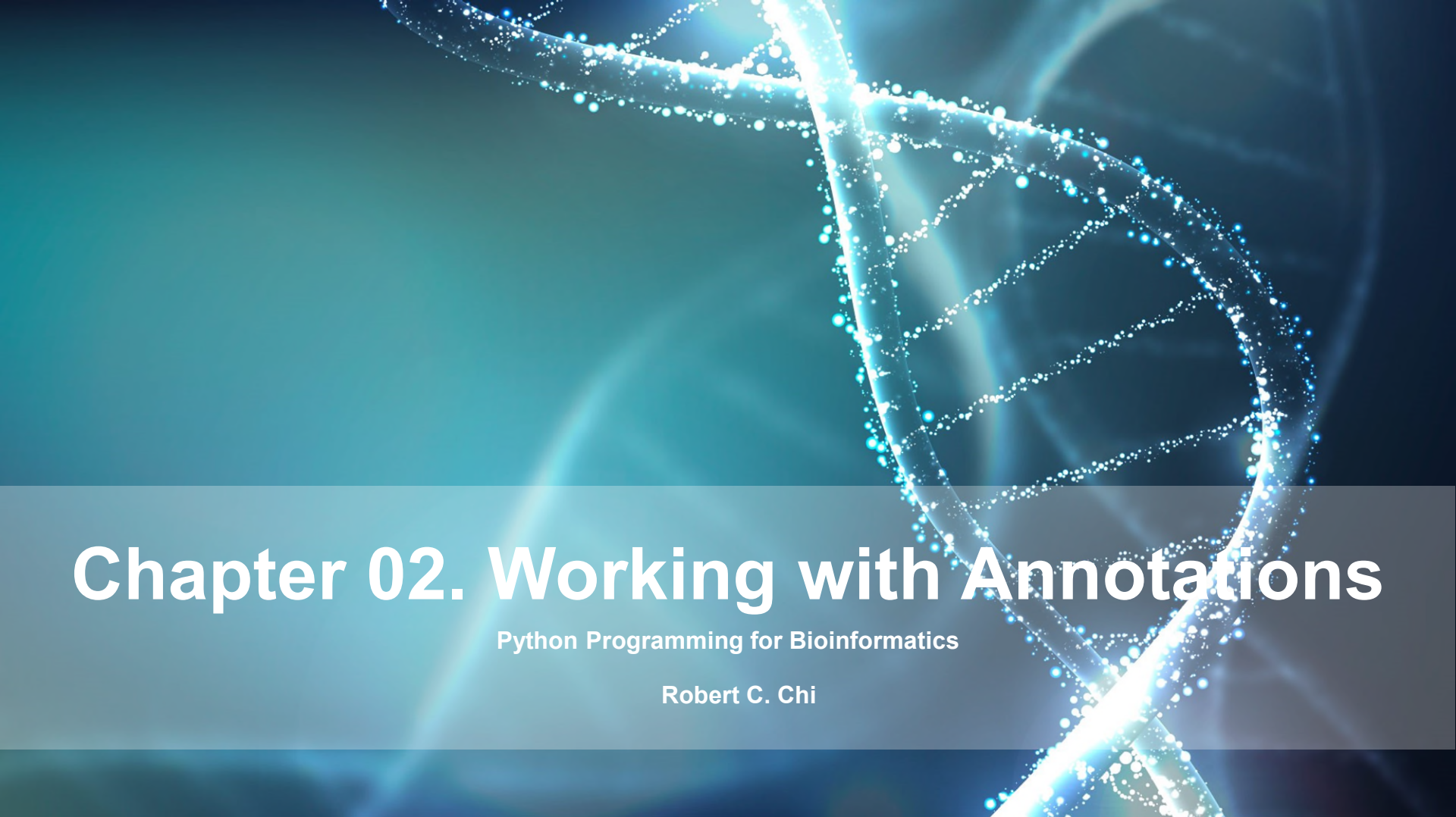


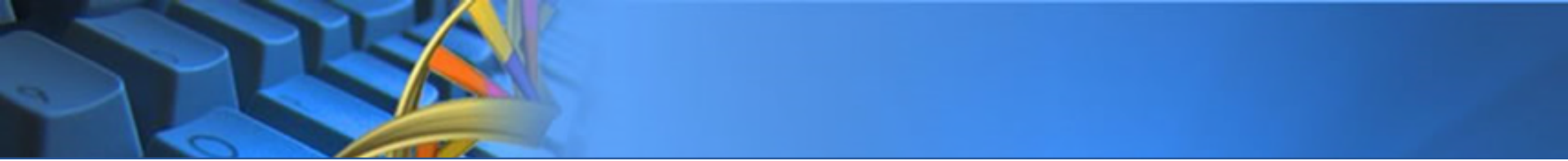
Chapter 02. Working with Annotations

Python Programming for Bioinformatics

Robert C. Chi

Agenda

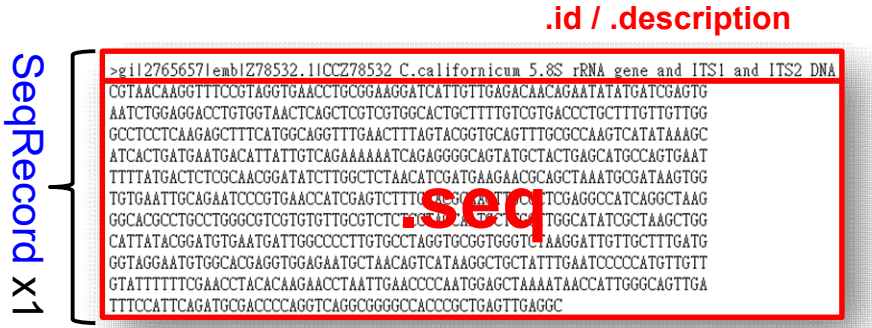
- **Introduction of Annotations**
- **Sequence Records**
  - Bio.SeqRecord.SeqRecord
- **References**
  - Bio.SeqFeature.Reference
- **Features, Locations, Positions**
  - Bio.SeqFeature.SeqFeature
- **Sequences + Features**
  - Bio.Seq.Seq
- **Summary**



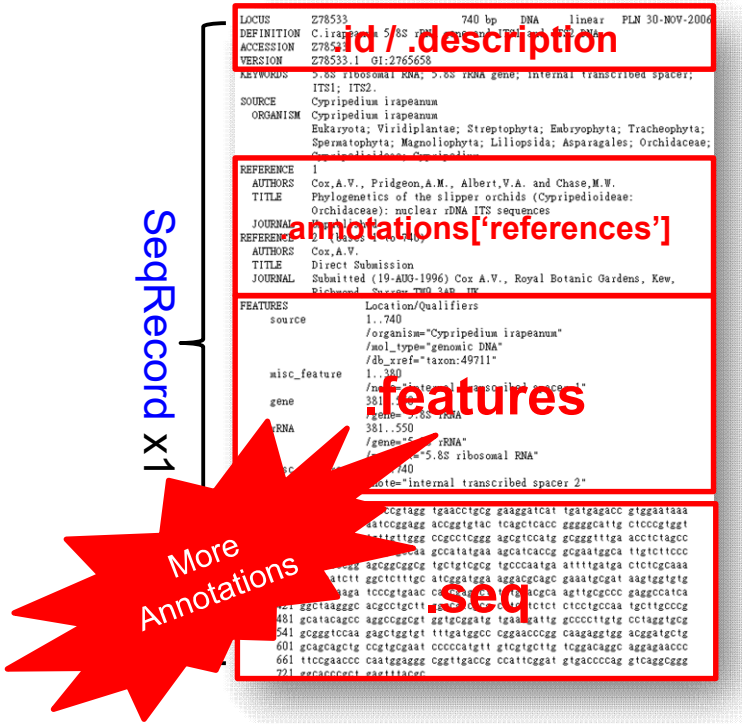
INTRODUCTION OF ANNOTATIONS

What is “Annotations”?

- **FASTA**



- **GenBank**



Definition of Annotations:

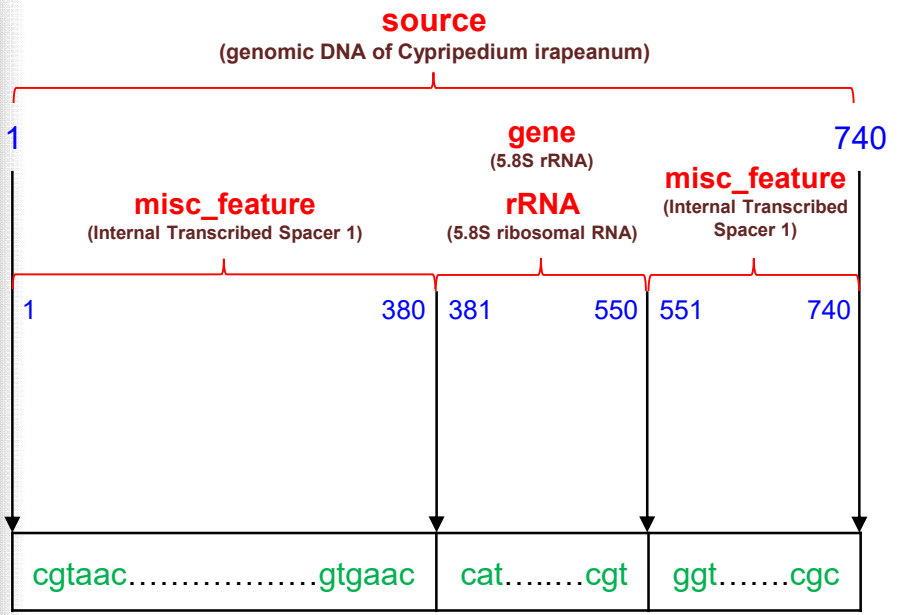
- The metadata other than sequences.
- e.g., ID, Name, Reference, Features…etc.

The Gem of GenBank File

Features

Sequence

```
FEATURES             Location/Qualifiers
1     source         1..740
                     /organism="Cypripedium irapeanum"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:49711"
2     misc_feature   1..380
                     /note="internal transcribed spacer 1"
3     gene           381..550
                     /gene="5.8S rRNA"
4     rRNA           381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"
5     misc_feature   551..740
                     /note="internal transcribed spacer 2"
ORIGIN
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
       61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
      121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
      181 cggcgcagtt tgggcgccaa gccatatgaa agcatcaccg gcgaatggca ttgtcttccc
      241 caaaacccgg agcggcggcg tgctgtcgcg tgcccaatga attttgatga ctctcgcaaa
      301 cgggaatctt ggctctttgc atcggatgga aggacgcagc gaaatgcgat aagtggtgtg
      361 aattgcaaga tcccgtgaac catcgagtct tttgaacgca agttgcgccc gaggccatca
      421 ggctaagggc acgcctgctt gggcgtcgcg cttcgtctct ctcctgccaa tgcttgcccg
      481 gcatacagcc aggccggcgt ggtgcggatg tgaaagattg gcccccttgt cctaggtgcg
      541 gcgggtccaa gagctggtgt tttgatggcc cggaacccgg caagaggtgg acggatgctg
      601 gcagcagctg ccgtgcgaat ccccccatgtt gtcgtgcttg tcggacaggc aggagaaccc
      661 ttccgaaccc caatggaggg cggttgaccg ccattcggat gtgacccccag gtcaggcggg
      721 ggcacccgct gagtttacgc
```



Annotation related Objects

GenBank File

```
LOCUS       Z78533                  740 bp    DNA     linear   PLN 30-NOV-2006
DEFINITION  C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA.
ACCESSION   Z78533
VERSION     Z78533.1  GI:2765658
KEYWORDS    5.8S ribosomal RNA; 5.8S rRNA gene; internal transcribed spacer;
            ITS1; ITS2.
SOURCE      Cypripedium irapeanum
  ORGANISM  Cypripedium irapeanum
            Eukaryota; Viridiplantae; Streptophyta; Embryophyta; Tracheophyta;
            Spermatophyta; Magnoliophyta; Liliopsida; Asparagales; Orchidaceae;
            Cypripedioideae; Cypripedium.
REFERENCE   1
  AUTHORS   Cox,A.V., Pridgeon,A.M., Albert,V.A. and Chase,M.W.
  TITLE     Phylogenetics of the slipper orchids (Cypripedioideae:
            Orchidaceae): nuclear rDNA ITS sequences
  JOURNAL   Unpublished
REFERENCE   2  (bases 1 to 740)
  AUTHORS   Cox,A.V.
  TITLE     Direct Submission
  JOURNAL   Submitted (19-AUG-1996) Cox A.V., Royal Botanic Gardens, Kew,
            Richmond, Surrey TW9 3AB, UK
FEATURES             Location/Qualifiers
     source          1..740
                     /organism="Cypripedium irapeanum"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:49711"
     misc_feature    1..380
                     /note="internal transcribed spacer 1"
     gene            381..550
                     /gene="5.8S rRNA"
     rRNA            381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"
     misc_feature    551..740
                     /note="internal transcribed spacer 2"
ORIGIN
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
       61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
      121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
      181 cggcgcagtt tgggcgccaa gccatatgaa agcatcaccg gcgaatggca ttgtctttcc
      241 caaaacccgg agcggcggcg tgctgtcgcg tgcccaatga attttgatga ctctcgcaaa
      301 cgggaatctt ggctctttgc atcggatgga aggacgcagc gaaatgcgat aagtggtgtg
      361 aattgcaaga tcccgtgaac catcgagtct tttgaacgca agttgcgccc gaggccatca
      421 ggctaagggc acgcctgctt gggcgtcgcg cttcgtctct ctcctgccaa tgcttgcccg
      481 gcatacagcc aggcccggcg ggtgcggatg tgaaagattg gcccctttgt gctaggtgcg
      541 gcgggtccaa gagctggtgt tttgatggcc cggaacccgg caagaggtgg acggatgctg
      601 gcagcagctg ccgtgcgaat cccccatgtt gtcgtgcttg tcggacaggc aggagaaccc
      661 ttccgaaccc caatggaggg cggttgaccg ccattcggat gtgacccccag gtcaggcggg
      721 ggcacccgct gagtttacgc
```

Bio.**SeqIO**

Bio.SeqRecord.**SeqRecord** x1

.name = 278533
.description = C. irapeanum 5.8S…
.id = 278533.1
…..

Bio.SeqFeature.**Reference** x2

```
REFERENCE   1
  AUTHORS   Cox,A.V., Pridgeon,A.M., Albert,V.A. and Chase,M.W.
  TITLE     Phylogenetics of the slipper orchids (Cypripedioideae:
            Orchidaceae): nuclear rDNA ITS sequences
  JOURNAL   Unpublished
```

Bio.SeqFeature.**SeqFeature** x5

```
     source          1..740
                     /organism="Cypripedium irapeanum"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:49711"
```

Bio.Seq.**Seq** x1

```
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
       61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
      121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
      181 cggcgcagtt tgggcgccaa gccatatgaa agcatcaccg gcgaatggca ttgtctttcc
      241 caaaacccgg agcggcggcg tgctgtcgcg tgcccaatga attttgatga ctctcgcaaa
      301 cgggaatctt ggctctttgc atcggatgga aggacgcagc gaaatgcgat aagtggtgtg
      361 aattgcaaga tcccgtgaac catcgagtct tttgaacgca agttgcgccc gaggccatca
      421 ggctaagggc acgcctgctt gggcgtcgcg cttcgtctct ctcctgccaa tgcttgcccg
      481 gcatacagcc aggcccggcg ggtgcggatg tgaaagattg gcccctttgt gctaggtgcg
      541 gcgggtccaa gagctggtgt tttgatggcc cggaacccgg caagaggtgg acggatgctg
      601 gcagcagctg ccgtgcgaat cccccatgtt gtcgtgcttg tcggacaggc aggagaaccc
      661 ttccgaaccc caatggaggg cggttgaccg ccattcggat gtgacccccag gtcaggcggg
      721 ggcacccgct gagtttacgc
```

Bio.SeqFeature.**FeatureLocation**

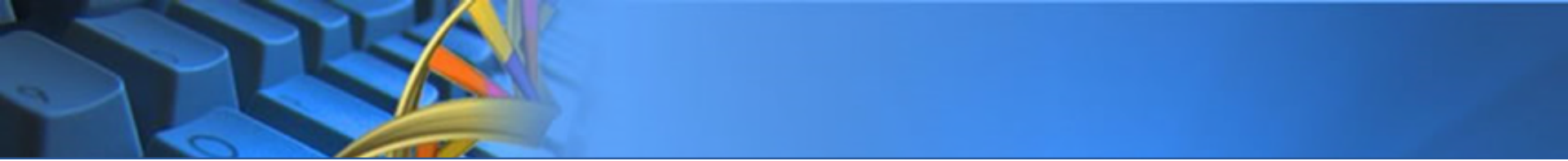
Bio.SeqFeature.**Position** Bio.SeqFeature.**Position**

1 .. 740

● **Blue:** **Annotation related Objects**

Brief of this Section

- **Annotation-related Objects**
  - Parsing: Bio.**SeqIO**
  - Records: Bio.SeqRecord.**SeqRecord** (+ .attributes)
  - References: Bio.SeqFeature.**Reference**
  - Features: Bio.SeqFeature.**SeqFeature**
    - Locations: Bio.SeqFeature.**FeatureLocation**
    - Positions: Bio.SeqFeature.**Position**
  - Sequences: Bio.Seq.**Seq**



Bio.SeqRecord.SeqRecord

SEQUENCE RECORDS

Parse Records from GenBank

```
1  # Create a list to store all parsed SeqRecords
2  seq_records = []
3
4  # Parse one Record a time and add into seq_records
5  from Bio import SeqIO
6  for rec in SeqIO.parse("ls_orchid.gbk", "genbank"):
7    seq_records.append(rec)          [0]  [1]  [2]  ...    [n]
8                    ↑ seq_records = [rec, rec, rec, ...] + [rec]
9  # Show how many records have been parsed
10 print("Total Number of Records:", len(seq_records))
11 print()
12
13 # Show the first record to prove the parsing was successful
14 print("--- First Record ---")
15 print(seq_records[0])
```

```
Total Number of Records: 94
```

```
--- First Record ---
ID: Z78533.1
Name: Z78533
....
Seq('CGTAAC...CGC')
```

Practice

- **Parse Records from GenBank**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

Download the GenBank File

```
[8]  1  import os
     2  if not os.path.isfile("ls_orchid.gbk"):
     3    os.system("wget https://raw.githubusercontent.com/biopython/biopython/master/Doc/examples/ls_orchid.gbk")
```

Get the List of All SeqRecords

```
1  # Create a list to store all parsed SeqRecords
2  seq_records = []
3
4  # Parse one Record a time and add into seq_records
5  from Bio import SeqIO
6  for rec in SeqIO.parse("ls_orchid.gbk", "genbank"):
7    seq_records.append(rec)
8
9  # Show how many records have been parsed
10 print("Total Number of Records:", len(seq_records))
11 print()
12
13 # Show the first record to prove the parsing was successful
14 print("--- First Record ---")
15 print(seq_records[0])
```

(Solution URL of this Practice)

Attributes of SeqRecord

```
1  LOCUS       Z78533                     740 bp    DNA     linear   PLN 30-NOV-2006
2  DEFINITION  C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA.
3  ACCESSION   Z78533
4  VERSION     Z78533.1  GI:2765658
5  KEYWORDS    5.8S ribosomal RNA; 5.8S rRNA gene; internal transcribed spacer;
6              ITS1; ITS2.
7  SOURCE      Cypripedium irapeanum
8  ORGANISM    Cypripedium irapeanum
9              Eukaryota; Viridiplantae; Streptop
10             Spermatophyta; Magnoliophyta; Lili
11             Cypripedioideae; Cypripedium.
```

```
1  # Take the first SeqRecord as Example
2  rec = seq_records[0]
3
4  # Show the attributes of SeqRecord
5  print("Locus Name:", rec.name)
6  print("Sequence Type:", rec.annotations['molecule_type'])
7  print("Sequence Topology:", rec.annotations['topology'])
8  print("Published Date:", rec.annotations['date'])
9  print("Sequence Definition:", rec.description)
10 print("Accession No.:", rec.annotations['accessions'])
11 print("Sequence Version:", rec.annotations['sequence_version'])
12 print("NCBI GenInfo Identifier (G.I.):", rec.annotations['gi'])
13 print("Keywords:", rec.annotations['keywords'])
14 print("Sequence Source:", rec.annotations['source'])
15 print("Organism:", rec.annotations['organism'])
16 print("Taxonomy:", rec.annotations['taxonomy'])
```

Practice

- **Get Attributes of SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

Show Attributes of SeqRecord

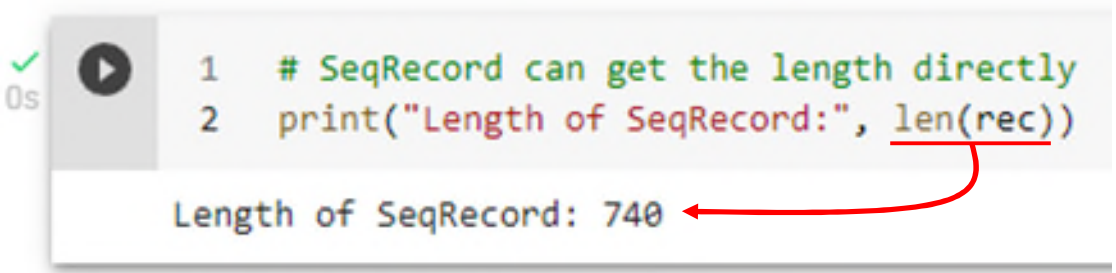
```
[17]  1  # Take the first SeqRecord as Example
      2  rec = seq_records[0]
      3
      4  # Show the attributes of SeqRecord
      5  print("Locus Name:", rec.name)
      6  print("Sequence Type:", rec.annotations['molecule_type'])
      7  print("Sequence Topology:", rec.annotations['topology'])
      8  print("Published Date:", rec.annotations['date'])
      9  print("Sequence Definition:", rec.description)
     10  print("Accession No.:", rec.annotations['accessions'])
     11  print("Sequence Version:", rec.annotations['sequence_version'])
     12  print("NCBI GenInfo Identifier (G.I.):", rec.annotations['gi'])
     13  print("Keywords:", rec.annotations['keywords'])
     14  print("Sequence Source:", rec.annotations['source'])
     15  print("Organism:", rec.annotations['organism'])
     16  print("Taxonomy:", rec.annotations['taxonomy'])
```

```
Locus Name: Z78533
Sequence Type: DNA
Sequence Topology: linear
Published Date: 30-NOV-2006
Sequence Definition: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Accession No.: ['Z78533']
Sequence Version: 1
NCBI GenInfo Identifier (G.I.): 2765658
Keywords: ['5.8S ribosomal RNA', '5.8S rRNA gene', ...]
Sequence Source: Cypripedium irapeanum
```

(Solution URL of this Practice)

Operation: Get Length

▾ Length of SeqRecord



Practice

- **Get the Length of Sequence in SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

Length of SeqRecord

```
[ ]  1  # SeqRecord can get the length directly
     2  print("Length of SeqRecord:", len(rec))
```

```
Length of SeqRecord: 740
```

(Solution URL of this Practice)

Operation: Slicing

```
FEATURES             Location/Qualifiers
     gene            381..550
                     /gene="5.8S rRNA"

     rRNA            381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"

   1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
  61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
 121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
 181 cggcgcagtt tgggcgccaa gccatatgaa agcatcaccg gcgaatggca ttgtcttccc
 241 caaaacccgg agcggcggcg tgctgtcgcg tgcccaatga attttgatga ctctcgcaaa
 301 cgggaatctt ggctctttgc atcggatgga aggacgcagc gaaatgcgat aagtggtgtg
 361 aattgcaaga tcccgtgaac catcgagtct tttgaacgca agttgcgccc gaggccatca
 421 ggctaagggc acgcctgctt gggcgtcgcg cttcgtctct ctcctgccaa tgcttgcccg
 481 gcatacagcc aggccggcgt ggtgcggatg tgaaagattg gccccttgtg cctaggtgcg
 541 gcgggtccaa gagctggtgt tttgatggcc cggaacccgg caagaggtgg acggatgctg
 601 gcagcagctg ccgtgcgaat cccccatgtt gtcgtgcttg tcggacaggc aggagaaccc
 661 ttccgaaccc caatggaggg cggttgaccg ccattcggat gtgaccccag gtcaggcggg
 721 ggcacccgct gagtttacgc
```

NCBI Index (1-based): [381 : 550]

Python Index (0-based): [380 : 549]

Upper bound Exclusive: [380 : 550)

```python
# Set the Start and End Position
startIndex = 381 - 1  # Python is 0-based
endIndex = 550  # Upperbound is exclusive

# Slice from the SeqRecord
sub_rRNA = rec[startIndex:endIndex]
print(sub_rRNA)
print(sub_rRNA.features)
```

```
ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 2
/molecule_type=DNA
Seq('CATCGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCAGGCTAAGGGCACGC...CAA')
[SeqFeature(FeatureLocation(ExactPosition(0), ExactPosition(170), strand=1), type='gene'),
 SeqFeature(FeatureLocation(ExactPosition(0), ExactPosition(170), strand=1), type='rRNA')]
```

Slice + Annotations Preserved !!

Practice

- **Slice a SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

Slicing of SeqRecord

```
# Set the Start and End Position
startIndex = 381 - 1  # Python is 0-based
endIndex = 550  # Upperbound is exclusive

# Slice from the SeqRecord
sub_rRNA = rec[startIndex:endIndex]
print(sub_rRNA)
print(sub_rRNA.features)
```

```
ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 2
/molecule_type=DNA
Seq('CATCGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCAGGCTAAGGGCACGC...CAA')
[SeqFeature(FeatureLocation(ExactPosition(0), ExactPosition(170), strand=1), type='gene'),
 SeqFeature(FeatureLocation(ExactPosition(0), ExactPosition(170), strand=1), type='rRNA')]
```

(Solution URL of this Practice)

Operation: Reverse Complement

```
1  # Reverse Complement
2  print("Original 5' to 3':")
3  print(rec)
4
5  print("Reverse Complement 5' to 3':")
6  print(rec.reverse_complement())
```

```
Original 5' to 3':
ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 5
/molecule_type=DNA
...
Seq('CGTAAC.....CGC')
```

```
Reverse Complement 5' to 3':
ID: <unknown id>
Name: <unknown name>
Description: <unknown description>
Number of features: 5
Seq('GCGTAA.....ACG')
```

- **Not all the Annotations Preserved !! But .features were kept!**
- **No .complement() support currently**

Practice

- **Reverse Complement of a SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

▾ Reverse Complement of SeqRecord

```
[17]  1  # Reverse Complement
      2  print("Original 5' to 3':")
      3  print(rec)
      4
      5  print("Reverse Complement 5' to 3':")
      6  print(rec.reverse_complement())
```

(Solution URL of this Practice)

Operation: Drop Nucleotides

- **Drop a Single Nucleotide at index=2 (0-based)**

CATCGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCAGGCTAAGGGCACGC...CAA

CA_CGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCAGGCTAAGGGCACGC...CAA

0~1 ~~2~~ 3~

```
1  # Drop the Single Nucleotide at index=2 (0-based)
2  print("Original rRNA:", sub_rRNA)
3  modified_rRNA = sub_rRNA[:2] + sub_rRNA[3:]
4  print("Modified rRNA:", modified_rRNA)
```

```
Modified rRNA: ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 0
/molecule_type=DNA
Seq('CACGAGTCTTTTGAACGCAAGTTGCGCCCG...CAA')
```

Practice

- **Drop Nucleotides of SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

Drop Nucleotides of SeqRecord

```
# Drop the Single Nucleotide at index=2 (0-based)
print("Original rRNA:", sub_rRNA)
modified_rRNA = sub_rRNA[:2] + sub_rRNA[3:]
print("Modified rRNA:", modified_rRNA)
```

(Solution URL of this Practice)

Operation: Reorganize two Genes

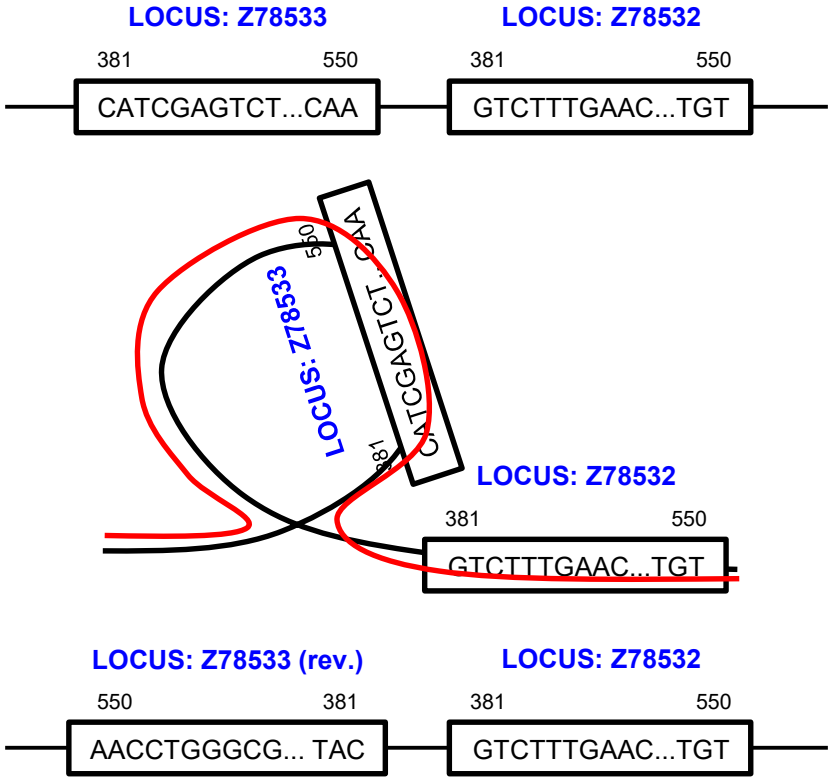
GenBank File

```
LOCUS       Z78533                  740 bp    DNA     linear   PLN 30-NOV-2006
DEFINITION  C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA.
...
FEATURES             Location/Qualifiers
     gene            381..550
                     /gene="5.8S rRNA"
     rRNA            381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"
ORIGIN
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
       61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
      121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
```

Locus ID: Z78533

```
//
LOCUS       Z78532                  753 bp    DNA     linear   PLN 30-NOV-2006
DEFINITION  C.californicum 5.8S rRNA gene and ITS1 and ITS2 DNA.
...
FEATURES             Location/Qualifiers
     gene            381..550
                     /gene="5.8S rRNA"
     rRNA            381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"
ORIGIN
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgttgagaca acagaatata
       61 tgatcgagtg aatctggagg acctgtggta actcagctcg tcgtggcact gcttttgtcg
      121 tgaccctgct ttgttgttgg gcctcctcaa gagctttcat ggcaggtttg aactttagta
```

Locus ID: Z78532



Operation: Reorganize two Genes

```python
# Reorganize two genes
sub_rRNA1 = seq_records[0][380:550]
print("rRNA 1 (5'-3'): ------")
print(sub_rRNA1)
print()

sub_rRNA2 = seq_records[1][380:550]
print("rRNA 2 (5'-3'): ------")
print(sub_rRNA2)
print()

# Reverse the sub_rRNA1
sub_rRNA1 = sub_rRNA1[::-1]
print("rRNA 1 (3'-5', Reversed): ------")
print(sub_rRNA1)
print()

# Join two rRNA
new_rRNA = sub_rRNA1 + sub_rRNA2
print("New rRNA : ------")
print(new_rRNA)
print()
```

```
rRNA 1 (5'-3'): ------
ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 2
/molecule_type=DNA
Seq('CATCGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCAGGCTAAGGGCACGC...CAA')
```

slice record[0]

```
rRNA 2 (5'-3'): ------
ID: Z78532.1
Name: Z78532
Description: C.californicum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 2
/molecule_type=DNA
Seq('GTCTTTGAACGCAAGTTGCGCTCGAGGCCATCAGGCTAAGGGCACGCCTGCCTG...TGT')
```

slice record[1]

```
rRNA 1 (3'-5', Reversed): ------
ID: Z78533.1
Name: Z78533
Description: C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
Number of features: 0
/molecule_type=DNA
Seq('AACCTGGGCGGCGTGGATCCGTGTTCCCCGGTTAGAAAGTGTAGGCGTGGTGCG...TAC')
```

features lost

reverse rRNA 1

```
New rRNA : ------
ID: <unknown id>
Name: <unknown name>
Description: <unknown description>
Number of features: 2
/molecule_type=DNA
Seq('AACCTGGGCGGCGTGGATCCGTGTTCCCCGGTTAGAAAGTGTAGGCGTGGTGCG...TGT')
```

information lost

join

Practice

- **Reorganize two Genes of SeqRecord**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

```
# Reorganize two genes
sub_rRNA1 = seq_records[0][380:550]
print("rRNA 1 (5'-3'): ------")
print(sub_rRNA1)
print()

sub_rRNA2 = seq_records[1][380:550]
print("rRNA 2 (5'-3'): ------")
print(sub_rRNA2)
print()

# Reverse the sub_rRNA1
sub_rRNA1 = sub_rRNA1[::-1]
print("rRNA 1 (3'-5', Reversed): ------")
print(sub_rRNA1)
print()

# Join two rRNA
new_rRNA = sub_rRNA1 + sub_rRNA2
print("New rRNA : ------")
print(new_rRNA)
print()
```

(Solution URL of this Practice)

Operation: Print in Other Formats

```
1  # Print the first SeqRecord in FASTA format
2  print(seq_records[0].format("fasta"))
```

```
>Z78533.1 C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA
CGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTGATGAGACCGTGGAATAAA
CGATCGAGTGAATCCGGAGGACCGGTGTACTCAGCTCACCGGGGGCATTGCTCCCGTGGT
GACCCTGATTTGTTGTTGGGCCGCCTCGGGAGCGTCCATGGCGGGTTTGAACCTCTAGCC
CGGCGCAGTTTGGGCGCCAAGCCATATGAAAGCATCACCGGCGAATGGCATTGTCTTCCC
CAAAACCCGGAGCGGCGGCGTGCTGTCGCGTGCCCAATGAATTTTGATGACTCTCGCAAA
CGGGAATCTTGGCTCTTTGCATCGGATGGAAGGACGCAGCGAAATGCGATAAGTGGTGTG
AATTGCAAGATCCCGTGAACCATCGAGTCTTTTGAACGCAAGTTGCGCCCGAGGCCATCA
GGCTAAGGGCACGCCTGCTTGGGCGTCGCGCTTCGTCTCTCTCCTGCCAATGCTTGCCCG
GCATACAGCCAGGCCGGCGTGGTGCGGATGTGAAAGATTGGCCCCTTGTGCCTAGGTGCG
GCGGGTCCAAGAGCTGGTGTTTTGATGGCCCGGAACCCGGCAAGAGGTGGACGGATGCTG
GCAGCAGCTGCCGTGCGAATCCCCCATGTTGTCGTGCTTGTCGGACAGGCAGGAGAACCC
TTCCGAACCCCAATGGAGGGCGGTTGACCGCCATTCGGATGTGACCCCAGGTCAGGCGGG
GGCACCCGCTGAGTTTACGC
```

Supported Format

Format name	Read	Write	Index	Notes
abi	1.58	No	N/A	Reads the ABI "Sanger" capillary sequence traces files, including the PHRED quality scores for the base calls. This allows ABI to FASTQ conversion. Note each ABI file contains one and only one sequence (so there is no point in indexing the file).
abi-trim	1.71	No	N/A	Same as "abi" but with quality trimming with Mott's algorithm.
ace	1.47	No	1.52	Reads the contig sequences from an ACE assembly file. Uses Bio.Sequencing.Ace internally
cif-atom	1.73	No	No	Uses Bio.PDB.MMCIFParser to determine the (partial) protein sequence as it appears in the structure based on the atomic coordinates.
cif-seqres	1.73	No	No	Reads a macromolecular Crystallographic Information File (mmCIF) file to determine the complete protein sequence as defined by the `_pdbx_poly_seq_scheme` records.

Full List: https://bit.ly/3c5q4Fs

(Note: Some format only support for read, not write)

Practice

- **Print SeqRecord in Other Formats**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

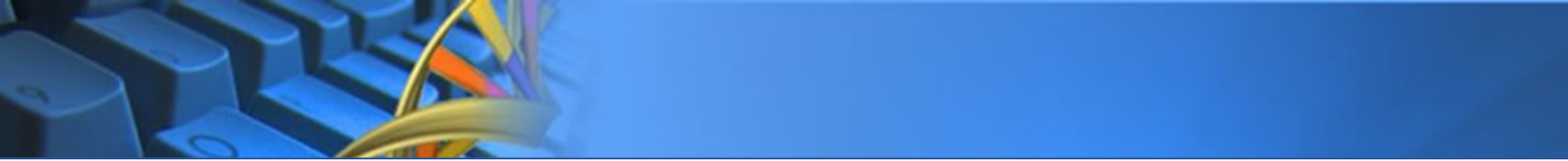
▾ Print SeqRecord in Other Formats

```
1  # Print the first SeqRecord in FASTA format
2  print(seq_records[0].format("fasta"))
```

(Solution URL of this Practice)

Brief of this Section

- **Bio.SeqRecord.SeqRecord**
  - Attributes: .id, .name, .description, .annotations[]...
  - Objects: .features[], .seq...
  - Operations:
    - Get Length: len(seq_rec)
    - Slicing: seq_rec[0:20] + Annotation Preserved
    - Reverse Complement: seq_rec.reverse_complement() + features kept
    - Drop Nucleotides: seq_rec[:2] + seq_rec[3:]
    - Reorganize: seq_rec[0][380:550:-1] + seq_rec[1][380:550]
    - Output Format: seq_rec.format("fasta")



Bio.SeqFeature.Reference

REFERENCES

What is “Reference”

```
LOCUS       Z78533                  740 bp    DNA     linear   PLN 30-NOV-2006
DEFINITION  C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA.
ACCESSION   Z78533
VERSION     Z78533.1  GI:2765658
KEYWORDS    5.8S ribosomal RNA; 5.8S rRNA gene; internal transcribed spacer;
            ITS1; ITS2.
SOURCE      Cypripedium irapeanum
  ORGANISM  Cypripedium irapeanum
            Eukaryota; Viridiplantae; Streptophyta; Embryophyta; Tracheophyta;
            Spermatophyta; Magnoliophyta; Liliopsida; Asparagales; Orchidaceae;
            Cypripedioideae; Cypripedium.
REFERENCE   1
  AUTHORS   Cox,A.V., Pridgeon,A.M., Albert,V.A. and Chase,M.W.
  TITLE     Phylogenetics of the slipper orchids (Cypripedioideae:
            Orchidaceae): nuclear rDNA ITS sequences
  JOURNAL   Unpublished
REFERENCE   2  (bases 1 to 740)
  AUTHORS   Cox,A.V.
  TITLE     Direct Submission
  JOURNAL   Submitted (19-AUG-1996) Cox A.V., Royal Botanic Gardens, Kew,
            Richmond, Surrey TW9 3AB, UK
```

Bio.**SeqIO**

Bio.SeqRecord.**SeqRecord**

.**annotations**[‘references’]

Bio.SeqFeature.**Reference**

Bio.SeqFeature.**Reference**

Attributes of References

```python
# Get the first SeqRecord
rec = seq_records[0]

# Iterate each reference
for ref in rec.annotations['references']:
    print("Author:", ref.authors)
    print("Title:", ref.title)
    print("Journal:", ref.journal)
    print("Medline ID:", ref.medline_id)
    print("PubMed ID:", ref.pubmed_id)
    print("Comments:", ref.comment)
    print()
```

```
Author: Cox,A.V., Pridgeon,A.M., Albert,V.A. and Chase,M.W.
Title: Phylogenetics of the slipper orchids (Cypripedioideae: Orchidaceae): nuclear rDNA ITS sequences
Journal: Unpublished
Medline ID:
PubMed ID:
Comments:

Author: Cox,A.V.
Title: Direct Submission
Journal: Submitted (19-AUG-1996) Cox A.V., Royal Botanic Gardens, Kew, Richmond, Surrey TW9 3AB, UK
Medline ID:
PubMed ID:
Comments:
```

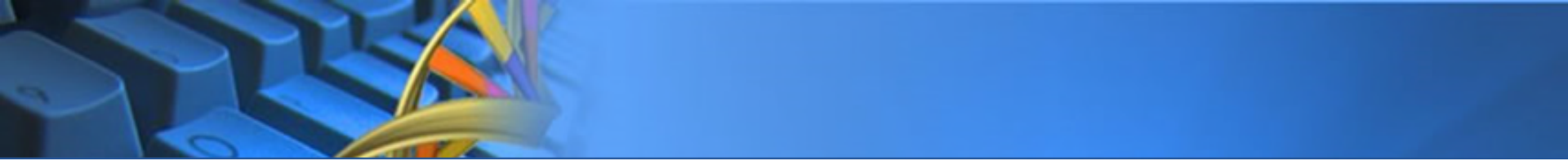
Practice

- **Show the Attributes of Bio.SeqFeature.Reference**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

```
# Get the first SeqRecord
rec = seq_records[0]

# Iterate each reference
for ref in rec.annotations['references']:
  print("Author:", ref.authors)
  print("Title:", ref.title)
  print("Journal:", ref.journal)
  print("Medline ID:", ref.medline_id)
  print("PubMed ID:", ref.pubmed_id)
  print("Comments:", ref.comment)
  print()
```

(Solution URL of this Practice)



Bio.SeqFeature.SeqFeature

FEATURES, LOCATIONS, POSITIONS

What is “Features”

- The **Annotations** to explain **sub-sequences**

```
LOCUS       Z78533                  740 bp    DNA     linear   PLN 30-NOV-2006
DEFINITION  C.irapeanum 5.8S rRNA gene and ITS1 and ITS2 DNA.
FEATURES             Location/Qualifiers
     source          1..740
                     /organism="Cypripedium irapeanum"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:49711"
     misc_feature    1..380
                     /note="internal transcribed spacer 1"
     gene            381..550
                     /gene="5.8S rRNA"
     rRNA            381..550
                     /gene="5.8S rRNA"
                     /product="5.8S ribosomal RNA"
     misc_feature    551..740
                     /note="internal transcribed spacer 2"
ORIGIN
        1 cgtaacaagg tttccgtagg tgaacctgcg gaaggatcat tgatgagacc gtggaataaa
       61 cgatcgagtg aatccggagg accggtgtac tcagctcacc gggggcattg ctcccgtggt
      121 gaccctgatt tgttgttggg ccgcctcggg agcgtccatg gcgggtttga acctctagcc
          ...
```

Bio.**SeqIO**

Bio.SeqRecord.**SeqRecord**

.**features**[ ]

- Bio.SeqFeature.**SeqFeature**
- Bio.SeqFeature.**SeqFeature**
- Bio.SeqFeature.**SeqFeature**
- Bio.SeqFeature.**SeqFeature**
- Bio.SeqFeature.**SeqFeature**

Attributes of SeqFeature

```
FEATURES             Location/Qualifiers
     ① source        ② 1..740
                        /organism="Cypripedium irapeanum"
                     ③ /mol_type="genomic DNA"
                        /db_xref="taxon:49711"
```

```
1  # Get the first SeqRecord
2  rec = seq_records[0]
3
4  # Iterate each feature
5  for feat in rec.features:
6      ① print("Feature Type:", feat.type)
7      ② print("Feature Location:", feat.location)
8      ③ print("Feature Qualifiers:", feat.qualifiers)
9      print()
```

.type (string)

- The type of feature.

.location (Bio.SeqFeature.**FeatureLocation**)

- The location of a feature.
- A compound object containing:
  - .start: The start position (e.g., 1)
  - .end: The end position (e.g., 740)

.qualifiers (OrderedDict)

- A dictionary stores the additional information.
  - Key: The name of information.
  - Value: The information itself.

Practice

- **Show the Attributes of Bio.SeqFeature.SeqFeature**
  - Write and Run the following codes on a Colab page called “AnnotationObjects.ipynb”:

```
# Get the first SeqRecord
rec = seq_records[0]

# Iterate each feature
for feat in rec.features:
  print("Feature Type:", feat.type)
  print("Feature Location:", feat.location)
  print("Feature Qualifiers:", feat.qualifiers)
  print()
```

(Solution URL of this Practice)