Exercise: Learning from Object Oriented Programming Code

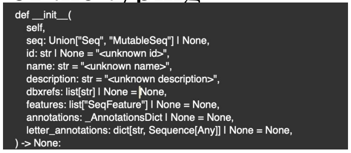
Biopython SeqRecord Class Analysis: Answer by hand except where it says **copy/paste**

Part I: Imported modules

1) Describe the purpose of the typing module:

The purpose of the typing module is to create an error free code by specifying type hints with expected data types such as variables, arguments and return values

2) **Copy/paste** a code snippet example of the use of the typing module in the code. (Check out the __init__):



```
def __init__(
    self,
    seq: Union["Seq", "MutableSeq"] | None,
    id: str | None = "<unknown id>",
    name: str = "<unknown name>",
    description: str = "<unknown description>",
    dbxrefs: list[str] | None = None,
    features: list["SeqFeature"] | None = None,
    annotations: _AnnotationsDict | None = None,
    letter_annotations: dict[str, Sequence[Any]] | None = None,
) -> None:
```

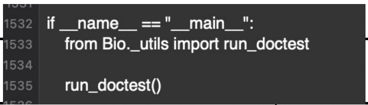
Explain what this code does:

This piece of code initializes SeqRecord object with attributes "Union["Seq", "MutableSeq"] | None," indicates that the following types can be used, "ID" represents the unique identifier and gives unknown ID if none is given. The code utilizes Union, list, dict directly from the typing module. Overall, it ensures each attribute has a correct type.

3) Describe the purpose of the doctest module:

The doctest module checks that docstrings are up to date, tests that interactive examples from a test file work as expected. You can also doctest to write documentation with specific input/output examples. If it does not behave as expected then it raises an error.

4



```
if __name__ == "__main__":
    from Bio._utils import run_doctest

    run_doctest()
```

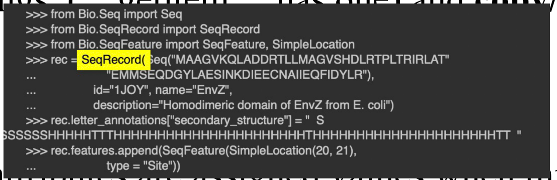
Explain what this code does:

"if__name__=="__main__"" is used to specify a block of test code within a script as opposed to importing as a module. "from Bio._utils import run_doctest" imports run-doctest from biopython and then "run doctest()" is a called from doctest. It runs doctests which are small tests written within docstrings.

Part II: Identify parts of the code using OOP Terms:

1) What kind of Class is SeqRecord?
   Standard class that inherits attributes, it would be a superclass if another class inherits SeqRecord.
2) On what line does the SeqRecod start and what is it called?
   LINE NUMBER: 114 NAME OF METHOD: __init__
   ↖ Starts line 180 (constructor method)
3) What data types are the following attributes (write next to the names):
   a. Id -> Str
   b. Name -> str
   c. Features -> list

4) Find an example of an instance of the class in one of the docstrings (getitem has one) and **copy/paste** it here:



```
>>> from Bio.Seq import Seq
>>> from Bio.SeqRecord import SeqRecord
>>> from Bio.SeqFeature import SeqFeature, SimpleLocation
>>> rec = SeqRecord(Seq("MAAGVKQLADDRTLLMAGVSHDLRTPLTRIRLAT"
...                     "EMMSEQDGYLAESINKDIEECNAIIEQFIDYLR"),
...                 id="1JOY", name="EnvZ",
...                 description="Homodimeric domain of EnvZ from E. coli")
>>> rec.letter_annotations["secondary_structure"] = "  S
SSSSSSHHHHHTTTHHHHHHHHHHHHHHHHHHHHHHTHHHHHHHHHHHHHHHHHHHHHHHHTT  "
>>> rec.features.append(SeqFeature(SimpleLocation(20, 21),
...                     type = "Site"))
```

5) What attributes are assigned values when this instance is created? Write them here:
   Seq, id, name, description

6) What ones are not assigned? Write them here:
   dbxrefs, features, annotations, letter_annotations

7) What is @property and @overload?

@property is used to define a method that can be used as an attribute. @overload is used to give a default method a different function in other words, assign multiple valid ways a method can be called (almost like overwriting).

Part III: Investigate some methods

1) How would you describe the __str__ method in OOP terminology?

→

defines a string representation of an object. Essentially, it overwrites the default representation of an object to a human readable version.

2) What would the following output be based on the __str__ function?

```
>>> record = SeqRecord(Seq("SNACKATTACK"),
id="CHPS_1001", name=" BBQ5", description="fried junk
food")
>>> print(record)
```

Write answer here:

ID = CHPS_1001
Name = BBQ5
Description = fried junk food
Number of features = 0
Seq ('SNACKATTACK')

effect? Write the answer next to the method:

a. __len__ len() → length of object operater
b. __lt__ (<) lt() → less than operater
c. __eq__ (==) eq() → equal to operater
d. __ne__ (!=) → not equal to operater
e. __add__ (+) → addition operater

4) Name three methods that do not affect built-in methods or operators and explain what they do:

upper() returns sequences in uppercases
lower() returns sequences in lowercases
Strip() returs new seq where specific characters removed from beginning and end.

5) What is the purpose of if __name__ == '__main__':

Used within the file where you write modules/classes to test your code, however, this does not get executed once it it is imported to another file.

6) Will this code run if you import SeqRecord? Explain your answer:

No, the purpose is so that you are able to test code but does not interfere once you import into another file. It will only work within the original file within the main code block, otherwise, that main block code will not run elsewhere.