Q1. Does assigning a value to a string's indexed character violate Python's string immutability?

Ans: As strings are immutable, String’s indexed character cannot be assigned to new value.

Q2. Does using the += operator to concatenate strings violate Python's string immutability? Why or why not?

Ans: The += operator is used to concatenate strings. This does not violate Python's string immutability property. This creates new associations between data and variables. For example: str\_1="a" and str\_1+="b. These instructions create the string ab and reassign it to the variable str\_1, but do not actually change the string data.

Q3. In Python, how many different ways are there to index a character?

Ans: There are 2 ways of indexing, positive indexing and negative indexing

Q4. What is the relationship between indexing and slicing?

Ans: Indexing and slicing both are used to access the elements. Indexing is to access individual elements and slicing is to access sequence of elements

Q5. What is an indexed character's exact data type? What is the data form of a slicing-generated substring?

Ans: String datatype.

Q6. What is the relationship between string and character "types" in Python?

Ans: An object which contains a sequence of character datatypes is called String.

Q7. Identify at least two operators and one method that allows you to combine one or more smaller strings to create a larger string.

Ans: +, += and \* allow to combine one or more smaller strings to create a larger string. .join() method joins elements of iterable type like list and tuple to get a combined string.

Q8. What is the benefit of first checking the target string with in or not in before using the index method to find a substring?

Ans: Checking the target string using the in or not operator before searching for a substring using the index method is only useful for checking the availability of the substring, hence the ValueError avoid occurrence.

Q9. Which operators and built-in string methods produce simple Boolean (true/false) results?

Ans: The following built-in methods will produce simple boolean results.

in

not

.isalpha()

.isalnum()

.isdecimal()

.isdigit()

.islower()

.isnumeric()

.isprintable()

.isspace()

.istitle()