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

No it does not. Strings are immutable, which means you cannot change an existing string.

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

It does not, strings are immutable. Strings are immutable on the Python side.

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

There are 2 ways to index a character. Accessing characters by a positive index number and negative index number.

Q4. What is the relationship between indexing and slicing?

The indexing is used as a resource to obtain individual elements and slicing is used to obtain a sequence of those elements.

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

Integer(non-negative) is the indexed character’s data type.

Integer is the data form for a slicing generated substring

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

The collection of characters in sequence forms a string. And Strings are used with single or double quotes.

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

+ operator and \* operator that allow to combine.

join() method is used to create a larger 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?

It helps to identify any exception if a substring is not found.

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

There are two operators in operator and not in operator that produce true and false results.