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

**Ans:** No, because we cannot assign a value to a string’s index.

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

**Ans:** No, it will not violate because we are not changing any index character in existing string, while using += operator we are just appending a new string to existing string.

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

**Ans:** 2 ways, accessing character by positive index and accessing character by negative index.

Q4. What is the relationship between indexing and slicing?

**Ans:** Indexing is used to get a particular character on particular index, while slicing is to get substring of a string and slicing can be achieved by using indexing only.

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

**Ans:** Data type of both indexed character and slicing-generated substring is “str”.

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

**Ans:** There is no explicitly character type in python, single character is also string type.

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

**Ans:** operators +, \*(with an integer). Method – join.

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:** It will prevent program to throw index out of range exception.

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

**Ans:** endswith, isalnum, isalpha, iascii, isdecimal, isdigit, isidentifier, islower, isnumeric, isprintable, isspace, istitle, isupper, startswith