Q1. In Python 3.X, what are the names and functions of string object types?

**Ans:** There are many name and functions of string object type in python 3.x. Few of them are below:

Upper(), lower(), isalnum(), isalpha(), islower(), isnumeric(), isspace(), istitle(), isupper(), isupper(), len(), join(), split(), replace().

Q2. How do the string forms in Python 3.X vary in terms of operations?

**Ans:**

Python 3 default storing of strings is Unicode whereas Python 2 stores need to define Unicode string value with "u."

Q3. In 3.X, how do you put non-ASCII Unicode characters in a string?

**Ans:**

The prefix 'u' in front of the quote indicates that a non ascii Unicode string is to be created.

Q4. In Python 3.X, what are the key differences between text-mode and binary-mode files?

**Ans:**

A text file stores data in the form of alphabets, digits and other special symbols by storing their ASCII values and are in a human-readable format. for example, any file with a .txt, .c,etc extension. whereas binary file contains a sequence or a collection of bytes which are not in a human-readable format.

Q5. How can you interpret a Unicode text file containing text encoded in a different encoding than your platform's default?

**Ans:**

To get encoding type in file:

import chardet

rawdata = open(file, "r").read()

result = chardet.detect(rawdata)

charenc = result['encoding']

to get default encoding on platform:

import sys

encoding = sys.getdefaultencoding()

Q6. What is the best way to make a Unicode text file in a particular encoding format?

**Ans:**

unicode\_text = u'ʑʒʓʔʕʗʘʙʚʛʜʝʞ'

encoded\_unicode = unicode\_text.encode("utf8")

a\_file = open("textfile.txt", "wb")

a\_file.write(encoded\_unicode)

Q7. What qualifies ASCII text as a form of Unicode text?

**Ans:** Appending ‘u’ in start of text.

Q8. How much of an effect does the change in string types in Python 3.X have on your code?

**Ans:**

Python3.x supports unicode characters better. Text strings are Unicode by default in Python3.x and the text strings are ASCII by default in Python2.x. You can also store string as unicode in Python 2.x by adding u while storing the strings. Unicode strings are more versatile and can store foreign language letters, numerals, symbols, emojis, etc.,