**Python advance assignment -9**

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

In Python 3.X, the main string object type is str, which can store Unicode text. The bytes object type is also available, which can store binary data in bytes.

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

In Python 3.X, str objects support operations such as string concatenation and slicing, while bytes objects support operations such as byte concatenation and slicing. Strings are immutable while bytes are mutable.

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

In Python 3.X, you can put non-ASCII Unicode characters in a string by prefixing the string literal with "u" or "U" or by using the **str.encode()** method to convert a bytes object to a string object.

**Q4. In Python 3.X, what are the key differences between text-mode and binary-mode files?** In Python 3.X, text-mode files automatically handle Unicode encoding and decoding, while binary-mode files do not. Text-mode files use str objects, while binary-mode files use bytes objects.

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

You can use the **open()** function with the **encoding** parameter to specify the encoding of the text file, and then use the **str.decode()** method to convert the bytes to a string object using the specified encoding.

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

The best way to make a Unicode text file in a particular encoding format is to use the **open()** function with the **mode** parameter set to 'w' or 'wb' and the **encoding** parameter set to the desired encoding. Then use the **write()** method to write the text to the file.

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

ASCII text is considered a form of Unicode text because ASCII is a subset of Unicode. All ASCII characters are represented by the same code points in Unicode, and can be encoded and decoded using the ASCII encoding.

**Q8. How much of an effect does the change in string types in Python 3.X have on your code?** The change in string types in Python 3.X can have a significant effect on code, especially for code that uses non-ASCII characters or works with binary data. It may require changes to how strings are handled and encoded, and may require additional testing to ensure compatibility with different encodings.