

Converting characters to unicode

C ANISH

Abstract—This is a document explaining conversion of characters from any language to its unicode value.

Download all python codes from

```
svn co https://github.com/chakki1234/
  Winter_intern/tree/main/unicode_convert/
  codes
```

process is repeated for the remaining characters in the word and for all the remaining words.

```
for i in words :
    if(i != ' ' and not(i.isdigit())) :
        for j in i :
            uni_file.write('U+' + hex(ord(j)).replace('x',' '))
            uni_file.write(' ')
        uni_file.close()
```

1 SOLUTION

- 1.1. *aigiri.txt* contains a slokam in Telugu it is read and all the characters are stored to the variable *sloka_txt* with the help of the code below.

```
sloka = open('aigiri.txt', 'r')
sloka_txt = sloka.read()
sloka.close()
```

- 1.2. *sloka_txt* is a string containing all the words it is split into individual words and the list of words are saved to the variable *words*. A new file *unicode.txt* is opened to write the converted unicode.

```
words = sloka_txt.split()
uni_file = open('unicode.txt', 'w')
```

- 1.3. A for loop runs through all the words in the list and checks if the word is ' ' or a number. If so it does not convert the word to its unicode value and proceeds with the next word. If the word is neither ' ' nor a number, an other for loop is used to access each character of the word. Each character is then passed on to a function called *ord()* which returns the integer code point value of the character. The integer value is then converted to its hexadecimal value using the function *hex()*. The hexadecimal string contains the character 'x' which is replaced with a null character to get the actual hexadecimal value it is then appened to the string 'U+' and the resultant is written onto the txt file. The