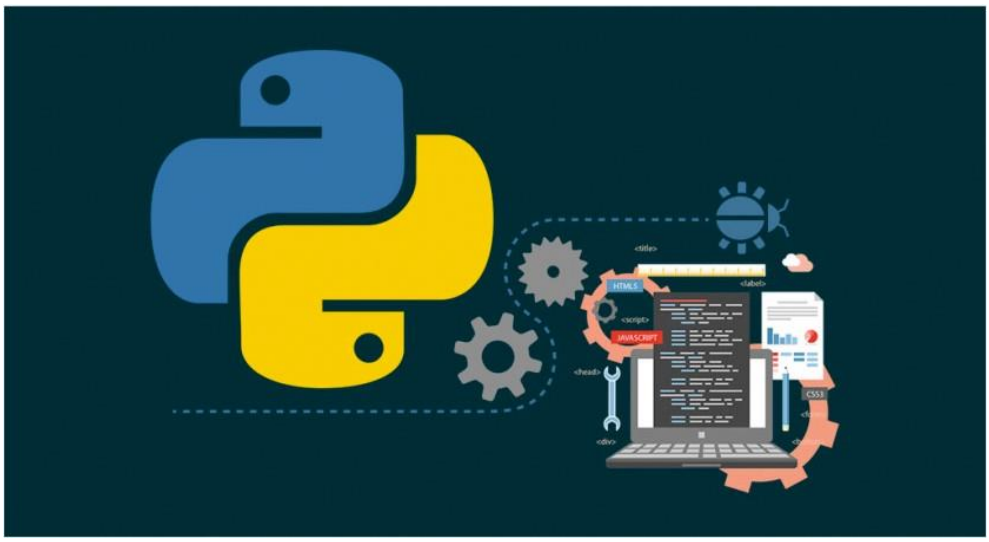

Random Character Generator

Get random characters, random sentences and more...

Small or capital letters, you choose!

Instead of using Python's built-in random module, this library was made by only using the Unix epoch which is the number of seconds that have elapsed since January 1, 1970.

How does this generator work?

Well, firstly I defined epoch seconds which will take a float value(v1), and then declared another natural variable which takes the integer value(v2) of epoch seconds.

Ex: v1 = 1612323.1234567, v2= 1612323

After that, I declared a new variable which takes the value of v1-v2(v3), why? Because the smallest number I take, the faster it will be updated. (Milliseconds are updated faster than seconds, so the faster it updates the faster my generator is)

Ex: v3 = v1-v2 = .1234567(Replaced the '.' with '' to get rid of dot)

Now I have the fastest updated digits in the epoch seconds(v3)

CODE:

```
v1 = time.time()
v2 = int(time.time())
v3 = (v1-v2)
number = int(str(v3).replace('.', '')) #Removing the 0
```

Since I'll be using this code every time in my for loops, I made a function (get_epochseconds()) which returns the '**number**' in the code above. (In this case **1234567**)

What are the functions and how do they work?

random_character(num_of_chars)

Small And Capital Letters Randomization

a. Get the **'number'** from get_epochseconds() function
b. Take the last digit of it by saying **number%10**
c. If it's odd, make the letter at the current index small letter, if it's even make it capital letter.

Converting the number to the characters we want

a. In the ASCII Code List, small letters are between **97-122** and capital letters are between **65-90.**
b. Take the remainder of the last 2 digits of the 'number' divided by 26 (Because there is 26 letters). This will give us a random number between **0** and **25**.
c. For small letters, add the random number we got between **0** and **25** to **97**, for capital letters add it to **65**

CODE:

```
if(int((number%10) % 2) == 0):
    flag = True
else:
    flag = False
if(flag == True):
    letter = chr(((number%100)%26) + 97)
    # letter_number = 97 + number%100%30
else:
    letter = chr(65 + number%100%26)
```

Output: kuiHkeucJ

random_character_between(num_of_chars, start, end)

Get the start and end chars' numbers in the ASCII code

a. Convert them both to Integer by using **ord()**
b. Since the user might want it to be between **z-a** instead of between **a-z**, we need to know which is smaller, we can do that by using **min()** and **max()**.
c. Get the range between them by saying **max()-min()**

Randomizing the letters between

a. Get the random 'number' from the get_epochseconds() function and take the last 2 digits of it (number%100)
b. Declare a variable called **number_between** which takes the value of **min()+1**, this will take the start range, the '+1' is because we don't want the char itself to appear
c. Take the remainder of dividing the last 2 digits by the number between and add the result to the start character.

CODE:

```
start = ord(x)
end = ord(y)
number_between = min(start, end)+1
number_between += ((number%100)%(max(start,end)-(min(start,end))))-1
if((int((number_between%10) % 2) == 0):
    flag = True
else:
    flag = False
letter = chr(number_between)
print(letter,end='')
```

Output(l,z): qovvxyqwnv

random_character_from(num_of_chars,char1,char2,char3, char4,char5,char6)

Randomly choosing a character between the 6 characters

a. Get the 'random' number from the get_epochseconds function.

b. Redefine 'number' by getting the remainder of it divided by 6 because we want to choose between 6 characters. (number = number%6)

c. Make 6 if-elif statements to choose between the 6 letters.

CODE:

```
number = self.get_epochseconds()
number = number%6 #There will be 6 choices if we count the 0 and 5 (6 is impossible)
if(number == 0):
    letter = a
elif(number == 1):
    letter = b
elif(number == 2):
    letter = c
elif(number == 3):
    letter = d
elif(number == 4):
    letter = e
else:
    letter = f
print(letter, end='')
```

Output(a, n, x, v, n, u): vunu

random_words(num_of_words)

The number of words and the length of them

a. The first for loop will be used for **randomizing the lengh** of the word and the second loop will be used for **randomizing the characters** of the current word.
b. Take the last digit of the random 'number' and take the remainder of it divided by 10. This will give us a random number between **0-9** (Which will be the length of our word)
c. Number of words will be taken as parameter from the user.
d. Randomize the letters by getting the last 2 digits of the random 'number', then get the remainder of the last 2 digits divided by **26** (Because there is 26 letters) which will look like this: **(number%100)%26**, then apply **the odd and even logic** to randomize the letters so it can be capital or small letter.

Spaces between words

a. While printing the letters, we should use **end='' ** so it prints the word, and for the spaces we can simply say **print(" ", end='')** in the outsider loop. The end='' should be applied here as well so it doesn't jump to a new line.
b. Summary: The outsider loop will print the **spaces**, the inner one will print the **letters(words)**.

Adding 2 more functions, 1 for words in small letters and 1 for words in capital letters

a. If we don't use the odd and even logic, we can easily make another 2 functions to let the user choose if he wants Randomized sentences with small and capital letters, sentences with only small letters or sentences with only capital letters.

CODE:

```
for q in range(n):
    number = self.get_epochseconds()

    sentence_size = number%10
    for _ in range(sentence_size):
        number = self.get_epochseconds()

        if((int(number%10) % 2) == 0):
            flag = True
        else:
            flag = False

        if(flag == True):
            letter = chr(((number%100)%26) + 97)
        else:
            letter_number = 65 + number%100%26
            letter = chr(65 + number%100%26)

        print(letter,end='')
    print(" ", end='')
```

Output(4): ky VsHqk cssHVyFg FiuqN

General Output:

```
C:\Program Files\Python39\python.exe
5 Random Sentence(s) with capital letters: ZYOPP C ZHTVNU A TFVPMASS
3 Random Sentence(s) with small letters: nyuepxjhn gfwio dn
4 Random Sentence(s): Vc DiVVL FgPPo sPJim
Random 10 character(s) between (n,z): wtpwtruuvo
Random 4 character(s) from (a, n, x, v, n, u): nnxa
Random 9 character(s): HqsaZLusw

Process returned 0 (0x0)        execution time : 1.463 s
Press any key to continue . . .
```

Qusay Gazawy

190905320