Password Generator - Python

About Password Generator:

A password is critical for authentication. This prevents unauthorized access to any website or portal where a possibility for misuse of information or identity theft persists. A password is a combination of lowercase, uppercase, symbols and digits. An ideal or a strong password contains a combination of these four. The length is also crucial in determining a password's strength with the suitable length being above 10 characters unless specified otherwise.

Python Hangman Project:

The python implementation of this project is using random and string modules. This project is suitable for beginners who are starting with python.

Project Prerequisites:

Password generator requires no extra installation of modules. Random and string modules are predefined.

Download Password Generator Code:

Download the code from the following link: Password Generator Python Code

Project File Structure:

Let's have a look at the steps to build the project:

1. Import necessary module
2. Define a variable
3. Read inputs and generator password

Let us look at the implementation in detail.

1. **Import modules**

```
#PythonGeeks program to generate a random password
#Import the necessary modules
import string
import random
```

Code Explanation:

- **Import random:** To randomly generate a password, which is a subset of string, we use this module
- **Import string:** Use this module to use predefined variables

2. **Define a variable:**

```
#Define a string containing letters, symbols and numbers
character_string = string.ascii_letters + string.punctuation +
string.digits
print("DEFINED STRING: "+character_string)
```

Code explanation:

- **Character_string:** Declaration and assignment of a variable to contain letters, symbols and digits. `string.ascii_letters` contains a uppercase and lowercase letters, `string.punctuation` contains symbols and `string.digits` comprises of numbers

3. **Read inputs and generator password:**

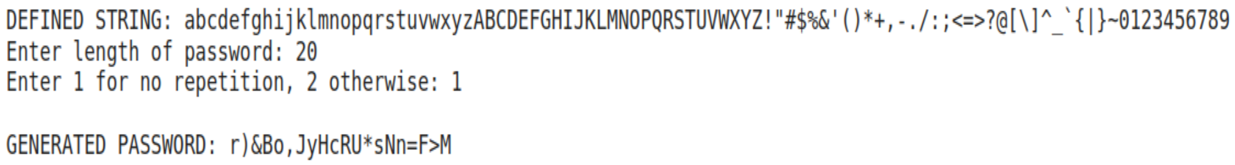
```
length = int(input("Enter length of password: "))
repeat = int(input("Enter 1 for no repetition, 2 otherwise: "))
if repeat == 1:
 password = random.sample(character_string,length)
else:
 password = random.choices(character_string,k=length)
password=''.join(password)

print("\nGENERATED PASSWORD: "+password)
```

Code explanation:

- **Text inputs:** Read the length of the password and check for repetition of characters from the user.
- **Repeat test condition:** If repeat is not enabled, use sample function and choice, if enabled. **Sample** randomly selects characters without repetition, whereas choice allows repetition.
- **Password = ''.join(password):** Both the random functions return a list. Hence join is used to convert the list to string using '' (no space or empty).

Project output:

Enter the inputs and view the output:



```
DEFINED STRING: abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~0123456789
Enter length of password: 20
Enter 1 for no repetition, 2 otherwise: 1

GENERATED PASSWORD: r)&Bo,JyHcRU*sNn=F>M
```

Summary

Thus using python, we created a simple password generator. This project also provided an introduction to string and random modules.