# Random Password Generator Documentation

Author: Muzaffar Ali

Version: 1.0

Copyright: (c) Muzaffar Ali

License: Public

## Purpose

This project is designed for the Advanced Python Class by Muzaffar Ali. The Random Password Generator creates secure and unpredictable passwords using Python's built-in random and string modules.

## Requirements

- Random Module: Provides functions to generate random numbers and selections, essential for creating unpredictable passwords.

- String Module: Offers a collection of string constants useful for various text operations.

## Environment Setup

### 1. Create a Virtual Environment

To ensure a clean environment, create a virtual environment using the following commands:

* For macOS/Linux:
* Create: python3 -m venv venv
* Activate: source venv/bin/activate
* For Windows:
* Create: python -m venv venv
* Activate: .\venv\Scripts\activate

### 2. Install Dependencies

With the virtual environment activated, install the necessary dependencies:  
```  
pip install random  
pip install string  
```  
\*Note: random and string are part of Python's standard library, so no installation is required.\*

## Code

The main code for generating the random password is as follows:  
```python  
import random  
import string  
  
def generate\_password(length):  
 if length < 4: # Ensure the minimum length to include all character types  
 raise ValueError("Password length should be at least 4 to include all character types.")  
  
 # Define the characters to use in the password  
 letters = string.ascii\_letters  
 digits = string.digits  
 special\_chars = string.punctuation  
  
 # Ensure the password contains at least one character from each category  
 password = [  
 random.choice(letters),  
 random.choice(digits),  
 random.choice(special\_chars),  
 random.choice(letters + digits + special\_chars)  
 ]  
  
 # Fill the rest of the password length with random choices from all characters  
 if length > 4:  
 password += random.choices(letters + digits + special\_chars, k=length-4)  
  
 # Shuffle the result to avoid predictable patterns  
 random.shuffle(password)  
  
 return ''.join(password)  
  
# Set the desired password length  
password\_length = 20  
  
# Generate and print the password  
print("Generated Password:", generate\_password(password\_length))  
```

## Execution

1. Activate the Virtual Environment:

* Windows:
* .\venv\Scripts\activate
* macOS/Linux:
* source venv/bin/activate

2. Run the Application:

```  
python password\_generator.py  
```

## How It Works

1. \*\*Imports:\*\*  
 - `random`: For generating random numbers and selections.  
 - `string`: For accessing string constants like letters, digits, and punctuation.  
  
2. \*\*Function `generate\_password(length)`:\*\*  
 - \*\*Ensures Minimum Length:\*\* Checks if the provided length is at least 4 to include one character from each category.  
 - \*\*Defines Character Sets:\*\* Specifies letters, digits, and special characters.  
 - \*\*Ensures Inclusion of Each Character Type:\*\* Adds at least one character from letters, digits, and special characters to the password.  
 - \*\*Fills the Remaining Length:\*\* Uses random choices to fill the remaining characters up to the desired length.  
 - \*\*Shuffles the Password:\*\* Shuffles the characters to avoid predictable patterns.  
 - \*\*Returns the Password:\*\* Joins the list of characters into a single string.  
  
3. \*\*Password Generation:\*\*  
 - The password length is set to 20.  
 - The `generate\_password` function is called to generate a random password.  
 - The generated password is printed.

## Output

The program generates and prints a secure random password of the specified length, ensuring it includes a mix of letters, digits, and special characters.