

Random Username Generator Project

Python Programming Internship Project

Submitted By: P Mukesh Varma

Submission Date: 05/02/2025

1. Objective

The goal of this project is to create a Python program that generates unique and fun usernames suitable for social media or gaming platforms. This project helps practice fundamental Python concepts, including working with lists, randomization, and file handling.

2. Project Features

- - Generate usernames by combining random adjectives and nouns, such as 'CoolTiger123' or 'HappyDragon!'.
- - Allow customization options, including adding numbers or special characters and setting username length.
- - Save generated usernames to a text file for future use or sharing.
- - Provide interactive user input for specifying preferences, such as adding numbers or special characters.

3. Implementation Details

The project is divided into three main components:

1. Random Username Generation: Combine adjectives and nouns randomly to generate unique usernames.
2. User Preferences: Allow users to specify username length and whether to include numbers or special characters.
3. File Handling: Save generated usernames to a text file.

4. Python Source Code

- Below is the complete Python code for the Random Username Generator:

```
import random
import string
```

```
# Pre-defined lists of adjectives and nouns
```

```
adjectives = ['Happy', 'Cool', 'Brave', 'Smart', 'Funky', 'Silly', 'Wild', 'Charming']
```

```

nouns = ['Tiger', 'Dragon', 'Phoenix', 'Knight', 'Pirate', 'Wizard', 'Lion']

def generate_username(include_number=True, include_special_char=True, length=8):
    adjective = random.choice(adjectives)
    noun = random.choice(nouns)
    username = adjective + noun

    if include_number:
        username += str(random.randint(10, 99))

    if include_special_char:
        username += random.choice(string.punctuation)

    # Adjust length if necessary
    return username[:length] if length < len(username) else username

def save_usernames_to_file(usernames, filename='usernames.txt'):
    with open(filename, 'w') as file:
        for username in usernames:
            file.write(username + '\n')
    print(f'Usernames saved to {filename}')

def main():
    print("Welcome to the Random Username Generator!")
    num_usernames = int(input("How many usernames would you like to generate? "))
    include_number = input("Include numbers? (y/n): ").strip().lower() == 'y'
    include_special_char = input("Include special characters? (y/n): ").strip().lower() == 'y'
    length = int(input("Enter desired username length: "))

    usernames = [generate_username(include_number, include_special_char, length) for _ in
range(num_usernames)]
    print("\nGenerated Usernames:")
    for username in usernames:
        print(username)

    save_usernames_to_file(usernames)

if __name__ == '__main__':
    main()

```

5. Code Explanation

1. ``generate_username()`:` Generates a random username by combining an adjective and a noun, with optional numbers and special characters.
2. ``save_usernames_to_file()`:` Saves the generated usernames to a text file.
3. ``main()`:` Provides an interactive menu for the user to generate usernames and customize options.

6. Output Verification Examples

- Example 1: Generating 3 usernames with numbers and special characters

User Input:

How many usernames? 3

Include numbers? y

Include special characters? y

Desired length: 10

Verified Output:

HappyLion78!

CoolKnight23#

FunkyPirate45@

- Example 2: Generating usernames without numbers or special characters

User Input:

How many usernames? 2

Include numbers? n

Include special characters? n

Desired length: 15

Verified Output:

BravePhoenix

CharmingWizard

- Example 3: Saving usernames to file

Verified Output: Usernames saved to usernames.txt

7. Conclusion

The Random Username Generator successfully demonstrates the use of basic Python concepts such as lists, randomization, file handling, and user interaction. It provides a fun and practical way to generate custom usernames and can be further extended with additional features.