МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж



Лабораторна робота №3

з дисципліни «Спеціалізовані мови програмування»

на тему «Розробка ASCII ART генератора для візуалізації текстових даних»

Виконав студент

групи РІ-32:

Фігура Т.Я

Прийняв:

Щербак С.С

Львів - 2024

Мета: створення додатка Генератора ASCII-арту.

Завдання 1: Введення користувача

Створіть Python-програму, яка приймає введення користувача для слова або фрази, яку треба перетворити в ASCII-арт.

if choice == '1':  
 text = input("Enter text for ASCII Art: ") *#Завдання 1: Введення користувача*  
generator = ASCIIArtGenerator(text, selected\_font, selected\_color)  
 art = generator.generate\_art()  
 print(art)

Завдання 2: Бібліотека ASCII-арту

Інтегруйте бібліотеку ASCII-арту (наприклад, pyfiglet або art) у вашу програму для генерації ASCII-арту з введення користувача

import pyfiglet *#Завдання 2: Бібліотека ASCII-арту*

Завдання 3: Вибір шрифту

Дозвольте користувачам вибирати різні стилі шрифтів для свого ASCII-арту. Надайте список доступних шрифтів та дозвольте їм вибрати один.

def list\_fonts(): *#Завдання 3: Вибір шрифту*  
fonts = pyfiglet.FigletFont.getFonts() *# Отримати всі доступні шрифти*  
for i, font in enumerate(fonts, 1):  
 print(f"{i}. {font}")  
 return fonts

Завдання 4: Колір тексту

Реалізуйте опцію вибору користувачем кольору тексту для їхнього ASCII-арту. Підтримуйте основний вибір кольорів (наприклад, червоний, синій, зелений).

def apply\_color(text, color='white'): *#Завдання 4: Колір тексту*  
if is\_color\_supported():  
 try:  
 return colored(text, color)  
 except KeyError:  
 return colored(text, 'white')  
 else:  
 return text *# Якщо кольори не підтримуються, просто повертаємо текст без кольору*

Завдання 5: Форматування виводу

Переконайтеся, що створений ASCII-арт правильно відформатований та вирівнюється на екрані для зручності читання.

def apply\_color(text, color):  
 return f"{color}{text}\033[0m"  
  
formatted\_art = apply\_color(ascii\_art, colors[selected\_color])  
print(formatted\_art)

Завдання 6: Збереження у файл

Додайте функціональність для збереження створеного ASCII-арту у текстовому файлі, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

def save\_to\_file(self, file\_path, art): *#Завдання 6: Збереження у файл*  
with open(file\_path, 'w') as f:  
 f.write(art)

Завдання 7: Розмір ARTу

Дозвольте користувачам вказувати розмір (ширина і висота) ASCII-арту, який вони хочуть створити. Масштабуйте текст відповідно.

def generate\_art\_with\_size(self, width): *#Завдання 7: Розмір ARTу*  
figlet = pyfiglet.Figlet(font=self.font, width=width)  
 ascii\_art = figlet.renderText(self.text)  
 return self.apply\_color(ascii\_art, self.color)

Завдання 8: Вибір символів

Дозвольте користувачам вибирати символи, які вони хочуть використовувати для створення ASCII-арту (наприклад, '@', '#', '\*', тощо).

symbols = input("Enter symbols to use (e.g., '@#\*'): ")  
ascii\_art = generator.generate\_with\_symbols(text, symbols)

Завдання 9: Функція попереднього перегляду

Реалізуйте функцію попереднього перегляду, яка показує користувачам попередній перегляд їхнього ASCII-арту перед остаточним збереженням.

def preview\_art(generator): *#Завдання 9: Функція попереднього перегляду*  
art = generator.generate\_art()  
 print(art)

Завдання 10: Інтерфейс, зрозумілий для користувача

Створіть зручний для користувача інтерфейс командного рядку для додатка, щоб зробити його інтуїтивно зрозумілим та легким у використанні.

def menu(): *#Завдання 10: Інтерфейс, зрозумілий для користувача*  
text = None  
 generator = None  
 selected\_font = 'slant'  
 selected\_color = 'reset'  
  
 while True:  
 print("\n--- ASCII Art Generator Menu ---")  
 print("1. Generate ASCII Art")  
 print("2. Choose Font")  
 print("3. Apply Color")  
 print("4. Save to File")  
 print("5. Preview Art")  
 print("6. Exit")  
  
 choice = input("Select an option: ")

Посилання на GitHub: [***https://github.com/FihuraTaras/SPL\_Python\_Project.git***](https://github.com/FihuraTaras/SPL_Python_Project.git)

Висновок: навчився створювати універсальний Генератор ASCII-арту, який дозволив налаштовувати свої творіння з різними шрифтами, кольорами, розмірами та символами. Проект дав практичний досвід роботи з введенням користувача, зовнішніми бібліотеками, роботою з файлами та дизайном інтерфейсу користувача в Python.