МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж

Лабораторна робота №3

з курсу

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

Виконав студент

групи ІТ-21сп

**Петровський Т.О.**

Прийняв

**Щербак С.С.**

Львів - 2023

**Мета:** створення додатка Генератора ASCII-арту.

**План роботи**

Завдання 1: Введення користувача

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

Завдання 2: Бібліотека ASCII-арту

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

Завдання 3: Вибір шрифту

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

Завдання 4: Колір тексту

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

Завдання 5: Форматування виводу

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

Завдання 6: Збереження у файл

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

Завдання 7: Розмір ARTу

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

Завдання 8: Вибір символів

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

Завдання 9: Функція попереднього перегляду

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

Завдання 10: Інтерфейс, зрозумілий для користувача

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

Код виконання наведений нижче.

import pyfiglet

from termcolor import colored

class AsciiArtGenerator:

def \_\_init\_\_(self, font\_choice, color\_choice, char\_choice):

self.font\_choice = font\_choice

self.color\_choice = color\_choice

self.char\_choice = char\_choice

def generate\_art(self, user\_input, width, height):

ascii\_art = pyfiglet.figlet\_format(user\_input, font=self.font\_choice)

colored\_ascii\_art = colored(ascii\_art, color=self.color\_choice)

formatted\_ascii\_art = ""

for line in colored\_ascii\_art.split("\n"):

formatted\_ascii\_art += line.center(width).replace(" ", self.char\_choice) + "\n"

formatted\_ascii\_art \*= height

return formatted\_ascii\_art

import pyfiglet

from Lab3.asci\_generator import AsciiArtGenerator

from utils import UserInputHelper

from utils.data\_saver import FileHandler

class AsciiGeneratorInterface:

def \_\_init\_\_(self):

self.\_\_font\_list = pyfiglet.FigletFont.getFonts()

self.\_\_color\_list = ['red', 'green', 'yellow', 'blue', 'magenta', 'cyan', 'white']

self.\_\_font\_choice = ""

self.\_\_color\_choice = ""

self.\_\_width = 0

self.\_\_height = 0

self.\_\_char\_choice = ""

def \_\_get\_user\_input(self):

print("ASCII Art Generator")

print("-------------------")

user\_input = UserInputHelper.get\_user\_input("Enter a word or phrase to convert to ASCII art: ")

self.\_\_font\_choice = UserInputHelper.get\_limited\_user\_input(

"Choose a font from the list below:\n{}\n".format(", ".join(self.\_\_font\_list)), self.\_\_font\_list)

self.\_\_color\_choice = UserInputHelper.get\_limited\_user\_input(

"Choose a color for the text ({})".format(", ".join(self.\_\_color\_list)), self.\_\_color\_list)

self.\_\_width = UserInputHelper.get\_int\_user\_input("Enter the width of the ASCII art: ")

self.\_\_height = UserInputHelper.get\_int\_user\_input("Enter the height of the ASCII art: ")

self.\_\_char\_choice = UserInputHelper.get\_user\_input("Enter a character to use for the ASCII art: ")

return user\_input

@staticmethod

def \_\_save\_ascii\_art(ascii\_art):

filename = UserInputHelper.get\_user\_input("Enter a filename to save the ASCII art: ")

FileHandler().save\_to\_txt(filename, ascii\_art)

def run(self):

user\_input = self.\_\_get\_user\_input()

ascii\_art\_generator = AsciiArtGenerator(self.\_\_font\_choice, self.\_\_color\_choice, self.\_\_char\_choice)

ascii\_art = ascii\_art\_generator.generate\_art(user\_input, self.\_\_width, self.\_\_height)

print("Preview:")

print(ascii\_art)

self.\_\_save\_ascii\_art(ascii\_art)

from Lab3.asci\_generator\_interface import AsciiGeneratorInterface

def execute\_lab3():

AsciiGeneratorInterface().run()

if \_\_name\_\_ == '\_\_main\_\_':

execute\_lab3()

**Висновки:** Виконуючи ці завдання, я створив універсальний Генератор ASCII-арту, який дозволить користувачам налаштовувати свої творіння з різними шрифтами, кольорами, розмірами та символами. Проект надасть практичний досвід роботи з введенням користувача, зовнішніми бібліотеками, роботою з файлами та дизайном інтерфейсу користувача в Python.