

Міністерство освіти і науки України
Національний університет “Львівська політехніка”
Кафедра інформаційних систем та мереж

ЗВІТ
про виконання лабораторної роботи № 3
“Розробка ASCII ART генератора для візуалізації текстових даних”
з дисципліни **“ Спеціалізовані мови програмування”**

Виконала студентка групи ІТ-32

ДЕНИСЯК С.-М. Т.

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 програми:

ascii_art_generator.py

```
import colorama
import pyfiglet
from colorama import Fore

colorama.init(autoreset=True)

fonts = dict(enumerate(sorted(pyfiglet.FigletFont.getFonts())))
colors = dict(enumerate(sorted(Fore.__dict__.keys())))

def display_fonts() -> None:
    for i in fonts:
        print(str(i) + ". " + fonts[i])

def display_colors() -> None:
    for i in colors:
        print(str(i) + ". " + colors[i])

def get_text(text, font, color_position, width) -> str:
    fig = pyfiglet.Figlet(font)
    fig.width = width
    formatted_text = fig.renderText(text)
    return Fore.__getattr__(colors[color_position]) + formatted_text

def write_into_file(file_path, text) -> None:
    with open(file_path, "w") as file:
        file.write(text)

def read_from_file(file_path) -> str:
    with open(file_path, "r") as file:
        return file.read()

while True:
    try:
        initial_text = str(input("Enter text containing all ASCII characters in order to display: "))
        if not initial_text.isascii():
            print("Text must contain only ASCII characters")
            continue
        display_fonts()
        font_position = int(input("Enter position of font you would like to use: "))
        display_colors()
        color_position = int(input("Enter position of color you would like to use: "))
        width = int(input("Enter width of text you would like to display: "))
        modified_text = get_text(initial_text, fonts[font_position], color_position, width)
        print(modified_text)
        write_into_file("output.txt", modified_text)

        if input("Would you like to continue? Enter 'Y' or 'y' if you do, or anything else if you don't. Your response is ").lower() == "y":
            continue
        else:
            break
```

```
except ValueError as e:
    print("Cannot be parsed into int value")
except KeyError:
    print("You have entered a wrong value for key of fonts or color")
except pyfiglet.CharNotPrinted as e:
    print(str(e))
```

Висновки. Виконуючи ці завдання, студенти створять 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.