

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІСМ



ЗВІТ

про виконання лабораторної роботи №3

з дисципліни

«Спеціалізовані мови програмування»

студента групи ІТ-32

Ткачишина Юрія

Прийняв Щербак С. С.

Львів - 2023

Мета роботи: Створення додатка Генератора ASCII-арту.

Індивідуальне завдання

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.

Хід виконання:

Код:

```
from colorama import Fore, init
```

```
import pyfiglet
```

```
import re
```

```
class ASCIIGenerator:
```

```
    def __init__(self):
```

```
self.text = ""

self.font = "standard"

self.color = "white"

self.width = 80

self.height = 25

self.custom_characters = ".,#@$%&*"

def get_user_input(self):

    print("Ласкаво просимо до Генератора ASCII-арту!")

    self.text = input("Введіть текст, який ви хочете перетворити в ASCII-арт: ")

    # Input validation for the font

    while True:

        self.font = input("Виберіть шрифт (наприклад: standard, banner, big, slant): ")

        valid_fonts = ["standard", "banner", "big", "slant"]

        if self.font in valid_fonts:

            break

        else:

            print("Недійсний шрифт. Будь ласка, виберіть один із наступних: standard, banner, big, slant")

    # Input validation for the color

    while True:

        self.color = input("Виберіть колір тексту (наприклад: red, blue, green, white): ")

        valid_colors = ["red", "blue", "green", "white"]

        if self.color in valid_colors:

            break
```

```

        else:
            print("Недійсний колір. Будь ласка, виберіть один із наступних: red, blue,
green, white")

        # Input validation for the width
        while True:
            try:
                self.width = int(input("Введіть ширину ASCII-арту (за замовчуванням 80):
"))
                if self.width > 0:
                    break
                else:
                    print("Ширина повинна бути додатнім числом.")
            except ValueError:
                print("Недійсне значення ширини. Введіть додатне ціле число.")

        # Input validation for the height
        while True:
            try:
                self.height = int(input("Введіть висоту ASCII-арту (за замовчуванням 25):
"))
                if self.height > 0:
                    break
                else:
                    print("Висота повинна бути додатнім числом.")
            except ValueError:
                print("Недійсне значення висоти. Введіть додатне ціле число.")

```

```
# Input validation for custom characters

while True:

    self.custom_characters = input("Введіть символи для створення ASCII-арту  
(наприклад: ., #@$%&*): ")

    valid_characters = re.compile(r'^[\w\s]+$')

    if valid_characters.match(self.custom_characters):

        break

    else:

        print("Недійсні символи. Будь ласка, використовуйте букви, цифри,  
пробіли та спеціальні символи.")
```

```
def generate_ascii_art(self):

    try:

        ascii_art = pyfiglet.Figlet(font=self.font)

        result = ascii_art.renderText(self.text)

        return result

    except pyfiglet.FigletError as e:

        return f"Помилка: {str(e)}"
```

```
def resize_ascii_art(self, ascii_art):

    lines = ascii_art.split('\n')

    resized_lines = []

    for line in lines:

        resized_line = ""

        for char in line:

            if char.isalnum() or char.isspace():
```

```

        resized_line += char

    else:

        resized_line += self.custom_characters[0]

    resized_lines.append(resized_line.ljust(self.width)[:self.width])

    resized_lines.extend([' ' * self.width] * (self.height - len(resized_lines)))

    return '\n'.join(resized_lines)


def display_ascii_art(self, ascii_art, color=None):
    if color:
        init(autoreset=True)

        colored_art = getattr(Fore, self.color.upper()) + self.resize_ascii_art(ascii_art)

        print(colored_art)

    else:

        print(self.resize_ascii_art(ascii_art))


def save_ascii_art(self, filename):
    ascii_art = self.generate_ascii_art()

    filename_with_extension = filename + ".txt"

    with open(filename_with_extension, 'w') as file:
        file.write(self.resize_ascii_art(ascii_art))


def preview_ascii_art(self):
    print("Попередній перегляд вашого ASCII-арту:")

    ascii_art = self.generate_ascii_art()

    self.display_ascii_art(ascii_art, self.color)


def run(self):

```

```
self.get_user_input()

self.preview_ascii_art()

save_option = input("Зберегти ASCII-арт у файл? (y/n): ")

if save_option.lower() == 'y':

    filename = input("Введіть ім'я файлу для збереження (без типу розширення): ")

    self.save_ascii_art(filename)

    print(f"ASCII-арт збережено у файлі: {filename}.txt")


if __name__ == "__main__":

    generator = ASCIIGenerator()

    generator.run()
```

Висновок: Виконавши ці завдання, я створив додаток генератора ASCII-арту.