МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж

Лабораторна робота №4

з курсу

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

Виконав студент

групи ІТ-21сп

**Петровський Т.О.**

Прийняв

**Щербак С.С.**

Львів - 2023

**Мета:** Створення Генератора ASCII-арту без використання зовнішніх бібліотек

**План роботи:**

Завдання 1: Введення користувача

Створіть програму Python, яка отримує введення користувача щодо слова або фрази, яку вони хочуть перетворити в ASCII-арт.

Завдання 2: Набір символів

Визначте набір символів (наприклад, '@', '#', '\*', тощо), які будуть використовуватися для створення ASCII-арту. Ці символи будуть відображати різні відтінки.

Завдання 3: Розміри Art-у

Запитайте у користувача розміри (ширина і висота) ASCII-арту, який вони хочуть створити. Переконайтеся, що розміри в межах керованого діапазону

Завдання 4: Функція генерації Art-у

Напишіть функцію, яка генерує ASCII-арт на основі введення користувача, набору символів та розмірів. Використовуйте введення користувача, щоб визначити, які символи використовувати для кожної позиції в Art-у.

Завдання 5: Вирівнювання тексту

Реалізуйте опції вирівнювання тексту (ліво, центр, право), щоб користувачі могли вибирати, як їх ASCII-арт розміщується на екрані.

Завдання 6: Відображення мистецтва

Відобразіть створений ASCII-арт на екрані за допомогою стандартних функцій друку Python.

Завдання 7: Збереження у файл

Додайте можливість зберігати створений ASCII-арт у текстовий файл, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

Завдання 8: Варіанти кольорів

Дозвольте користувачам вибирати опції кольорів (чорно-білий, відтінки сірого) для свого ASCII-арту.

Завдання 9: Функція попереднього перегляду

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

Завдання 10: Інтерфейс, зрозумілий для користувача

Створіть інтерфейс для користувача у командному рядку, щоб зробити програму легкою та інтуїтивно зрозумілою для використання.

Код виконання програми наведений нижче.

from .main import execute\_lab4

class AsciiArtGenerator:

def \_\_init\_\_(self, user\_input: str, width: int, height: int, alignment: str, color\_option: str):

self.\_\_user\_input = user\_input

self.\_\_ascii\_chars = ['@', '#', '\*', '+', '=', '-', ':', '.', ' ']

self.\_\_width = width

self.\_\_height = height

self.\_\_alignment = alignment

self.\_\_color\_option = color\_option

def set\_ascii\_chars(self):

if self.\_\_color\_option == "gray":

self.\_\_ascii\_chars = [' ', '.', ':', '-', '=', '+', '\*', '#', '%', '@']

elif self.\_\_color\_option == "bw":

self.\_\_ascii\_chars = [' ', '.', ',', '-', ':', ';', '=', '!', '\*', '#', '$', '@']

def generate\_ascii\_art(self):

ascii\_art = ""

for i in range(self.\_\_height):

for j in range(self.\_\_width):

char\_index = int((ord(self.\_\_user\_input[(i \* self.\_\_width + j) % len(self.\_\_user\_input)]) / 255) \* (

len(self.\_\_ascii\_chars) - 1))

ascii\_art += self.\_\_ascii\_chars[char\_index]

ascii\_art += "\n"

return ascii\_art

def set\_alignment(self):

if self.\_\_alignment == "center":

self.\_\_user\_input = self.\_\_user\_input.center(self.\_\_width)

elif self.\_\_alignment == "right":

self.\_\_user\_input = self.\_\_user\_input.rjust(self.\_\_width)

def set\_color\_option(self):

self.set\_ascii\_chars()

def preview\_ascii\_art(self):

print(self.generate\_ascii\_art())

from Lab4.ascii\_art\_generator import AsciiArtGenerator

from utils import UserInputHelper

from utils.data\_saver import FileHandler

class ConsoleInterface:

@staticmethod

def \_\_get\_user\_input() -> str:

return UserInputHelper.get\_user\_input("Введіть слово або фразу, яку хочете перетворити в ASCII-арт: ")

@staticmethod

def \_\_get\_size() -> tuple[int, int]:

width = UserInputHelper.get\_int\_user\_input("Введіть ширину ASCII-арту: ")

height = UserInputHelper.get\_int\_user\_input("Введіть висоту ASCII-арту: ")

return width, height

@staticmethod

def \_\_get\_alignment() -> str:

return UserInputHelper.get\_limited\_user\_input("Введіть вирівнювання тексту (left, center, right): ",

["left", "center", "right"])

@staticmethod

def \_\_get\_color\_option() -> str:

return UserInputHelper.get\_limited\_user\_input("Виберіть опцію кольорів (bw, gray): ", ["bw", "gray"])

@staticmethod

def run\_program():

while True:

print("1. Створити ASCII-арт")

print("2. Попередній перегляд ASCII-арту")

print("3. Вийти з програми")

choice = UserInputHelper.get\_limited\_user\_input("Виберіть опцію: ", ["1", "2", "3"])

if choice == "1":

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

width, height = ConsoleInterface.\_\_get\_size()

alignment = ConsoleInterface.\_\_get\_alignment()

color\_option = ConsoleInterface.\_\_get\_color\_option()

generator = AsciiArtGenerator(user\_input, width, height, alignment, color\_option)

generator.set\_alignment()

print(generator.generate\_ascii\_art())

ConsoleInterface.\_\_save\_in\_file(generator.generate\_ascii\_art())

generator.set\_color\_option()

elif choice == "2":

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

width, height = ConsoleInterface.\_\_get\_size()

alignment = ConsoleInterface.\_\_get\_alignment()

color\_option = ConsoleInterface.\_\_get\_color\_option()

generator = AsciiArtGenerator(user\_input, width, height, alignment, color\_option)

generator.set\_alignment()

generator.preview\_ascii\_art()

elif choice == "3":

break

@staticmethod

def \_\_save\_in\_file(data):

is\_save = UserInputHelper.get\_limited\_user\_input("Do you want to save it?(y/n)\n", ['y', 'n'])

if is\_save == 'y':

file\_name = UserInputHelper.get\_user\_input("Enter file name:\n")

FileHandler.save\_to\_txt(file\_name, data)

from Lab4.ascii\_art\_interface import ConsoleInterface

def execute\_lab4():

ConsoleInterface.run\_program()

if \_\_name\_\_ == "\_\_main\_\_":

execute\_lab4()

**Висновок:** Виконуючи ці завдання, я створив генератор ASCII-арту з нуля, та надав можливість налаштовувати символи, розміри, вирівнювання та кольори, що дозволить їм глибше розібратися як створюється ASCII-арт