

Мета: навчитись розрізняти та створювати різноманітні елементи графічного інтерфейсу прискладанні програм з використанням властивостей основних графічних об'єктів та об'єктно-орієнтованого підходу на мові Python в інтерактивному режимі IDLE.

Обладнання: ОС Windows, Microsoft Office2013, середовище програмування IDLE

Хід роботи

1 Використовуючи головне меню ОС Windows запустив інтерактивний режим інтерпретатора Python – середовище програмування IDLE

2 Створив в своєму каталогі папку ЛР21 для збереження наступних завдань

3 За допомогою команди File→NewFile (Ctrl+N) створив новий текстовий документ з відповідною назвою ЛР_21 для розробки наступного коду програми

4 Виконав індивідуальне завдання

Код програми:

```
from tkinter import *
from tkinter.ttk import *
from math import *
from json import dumps
class ButtonSlider(Frame):
    def __init__(self,master,inc_dec,value,label="",onchange=None):
        Frame.__init__(self,master)
        self.inc_dec = inc_dec
        self.value = value
        self.labelText = label
        self.initUI()
        self._onchange = onchange
```

Змн.	Арк.	№ докум.	Підпис	Дата
Розроб.	Мітленко Д.А.			
Перевір.	Саприкіна І.Г.			
Реценз.				
Н. Контр.	Саприкіна І.Г.			
Замерд.	Саприкіна І.Г.			

ДТЗЕ.121РП300ТЕХ2100ЛР

**Створення елементів
графічного інтерфейсу**

Лім. Арк. Аркушів

1 7

Група ПЗ-19-1/9

```

def updateView(self):
    self.entry.delete(0,END)
    self.entry.insert(0,str(round(self.value,3)))

def inc(self,event):
    self.value += self.inc_dec
    self.updateView()
    self._onchange(self)

def dec(self,event):
    self.value -= self.inc_dec
    self.updateView()
    self._onchange(self)

def initUI(self):
    self.but_min = b1 = Button(self,text="-" +
str(round(self.inc_dec,1)),width=5)
    self.but_add = b2 = Button(self,text="+" +
str(round(self.inc_dec,1)),width=5)
    self.entry = e1 = Entry(self,width=10)
    self.label = l1 = Label(self,text=self.labelText + ':')
    b1.bind('<Button-1>', self.dec)
    b2.bind('<Button-1>', self.inc)
    l1.grid(column=1,row=0)
    b1.grid(column=0,row=1)
    e1.grid(column=1,row=1)
    b2.grid(column=2,row=1)
    self.updateView()

def __str__(self):
    return '%f'%self.value

class MainWindow(Tk):
    x_inc = 1

```

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2100ЛР

Арк.

2

```

x_fac = 0.04
y_amp = 80
width = 400
height = 300
def __init__(self, parent):
    Tk.__init__(self, parent)
    self.parent = parent
    self.center = self.height//2
    self._do()
def _do(self):
    def drawData():
        string = 'x_inc=' + str(self.x_inc) + ' x_fac=' + str(round(self.x_fac, 2)) + \
y_amp=' + str(self.y_amp)
        canvas.create_text(10, 20, anchor=SW, text=string)
    def drawSin():
        xy1 = []
        for x in range(self.width):
            xy1.append(x * self.x_inc)
            if funcType.get() == 0:
                xy1.append(int(sin(x * self.x_fac) * self.y_amp) + self.center)
            elif funcType.get() == 1:
                xy1.append(int(cos(x * self.x_fac) * self.y_amp) + self.center)
            elif funcType.get() == 2:
                xy1.append(int(tan(x * self.x_fac) * self.y_amp) + self.center)
        canvas.create_line(xy1, fill='green')
        self.dots = xy1
        if cLineEnable.get(): canvas.create_line(0, self.center, self.width,
self.center, fill='blue')
    def update():

```

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2100ЛР

Арк.

3

```

        canvas.delete("all")
        drawData()
        drawSin()
        t1.delete(0.0,END)
        t1.insert(0.0,dumps(self.dots))
    def bs1_change(event):
        self.x_inc = event.value
    def bs2_change(event):
        self.x_fac = event.value
    def bs3_change(event):
        self.y_amp = event.value
    cLineEnable = IntVar()
    funcType = IntVar()
    cLineEnable.set(0)
    funcType.set(0)
    canvas = Canvas(self, width = self.width, height =
self.height,borderwidth=2, relief="groove")
    controlFrame = Frame(self)
    bs1 = ButtonSlider(controlFrame,1,self.x_inc,'Інкремент X',bs1_change)
    bs2 = ButtonSlider(controlFrame,0.01,self.x_fac,'Фактор X',bs2_change)
    bs3 = ButtonSlider(controlFrame,5,self.y_amp,'Амплітуда Y',bs3_change)
    b1 = Button(self,text='Оновити',command=update)
    c1 = Checkbutton(controlFrame,text='Відображати центральну
лінію',onvalue=1,offvalue=0,variable=cLineEnable)
    r1 = Radiobutton(controlFrame,text='sin(x)',variable=funcType,value=0)
    r2 = Radiobutton(controlFrame,text='cos(x)',variable=funcType,value=1)
    r3 = Radiobutton(controlFrame,text='tg(x)',variable=funcType,value=2)
    l1 = Label(self,text='JSON:')
    t1 = Text(self,height=4)

```

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2100ЛР

Арк.

```
canvas.pack()
controlFrame.pack(fill=X)
bs1.grid(column=0,row=0)
bs2.grid(column=0,row=1)
bs3.grid(column=0,row=2)
r1.grid(column=1,row=0)
r2.grid(column=1,row=1)
r3.grid(column=1,row=2)
c1.grid(column=2,row=0)
l1.pack(fill=X)
t1.pack(fill=X)
b1.pack(side=BOTTOM)

if __name__ == '__main__':
    app = MainWindow(None)
    app.resizable(False, False)
    app.title('Графік функцій')
    app.geometry('540x560')
    app.mainloop()
```

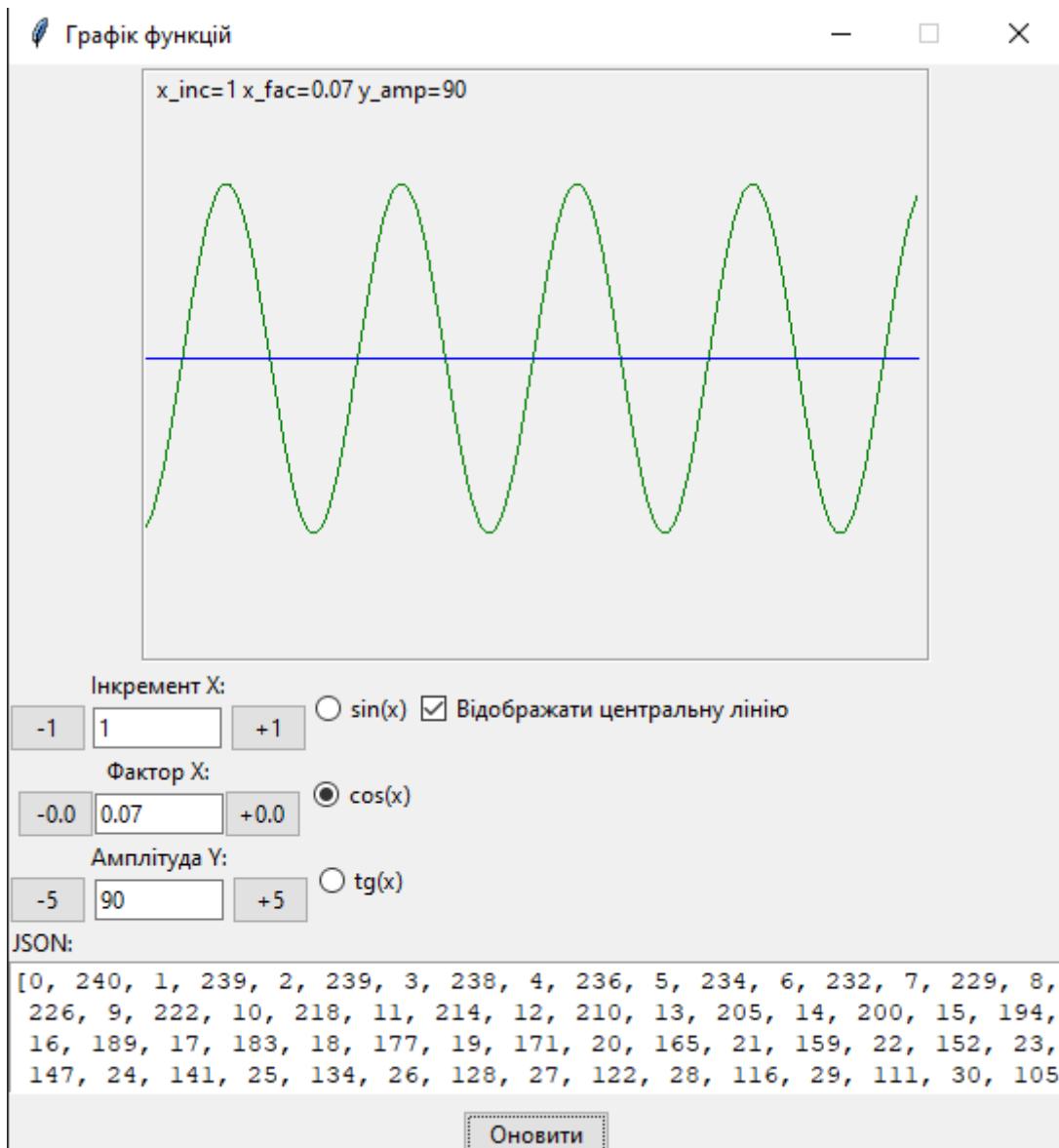
Результат зображеній на скріншоті 1

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2100ЛР

Арк.

5



Скріншот 1

Відповіді на контрольні запитання

1 Для чого призначено модуль tkinter?

Цей модуль призначений для створення графічного інтерфейсу користувача

2 Які основні об'єкти містить цей модуль?

До основних об'єктів відносяться: кнопка, мітка, поле вводу, радіокнопка, пропорці.

3 За допомогою якої інструкції створюються віджети?

Віджети створюються як і звичайний клас, бо вони і є класи

4 Які переваги та недоліки має графічний інтерфейс користувача?

		Mітленко Д.А.		
Змін.	Арк.	№ докум.	Підпис	Дата

Переваги: графічний інтерфейс є «дружнім» для користувачів, котрі розпочали знайомство з комп'ютером з графічного інтерфейсу. Недоліки: більше споживання пам'яті в порівнянні з текстовим інтерфейсом.

5 Поясніть загальний порядок створення графічного інтерфейсу користувача?

Порядок створення:

- 1) Створення головного вікна
- 2) Створення віджетів та ініціалізація даних.
- 3) Зв'язування подій з функціями обробки.
- 4) Розміщення віджетів у вікні (вікнах).
- 5) Запуск головного циклу.

6 Яку загальну структуру має функція для визначення події?

Має таку форму: змінна.bind("<Button>", ім'я функції)

7 Які основні етапи потрібно виконати для створення вікна?

Для створення вікна потрібно:

- 1) Створити об'єкт вікна
- 2) Призначити йому розміри і назву
- 3) Запустити головний цикл вікна

Висновок: я навчився розрізняти та створювати різноманітні елементи графічного інтерфейсу прискладанні програм з використанням властивостей основних графічних об'єктів та об'єктно-орієнтованого підходу на мові Python в інтерактивному режимі IDLE.

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2100ЛР

Арк.

7