

Урок 29 Проект Калькулятор Етап III

Отже, зображення у вікні виводить такий код:

```
canvas=Canvas(tk,width=400,height=350)
canvas.pack()
flower_image=PhotoImage(file="flowers.png")
canvas.create_image(200,175,image=flower_image)
```

2) Готуємо саму картину.

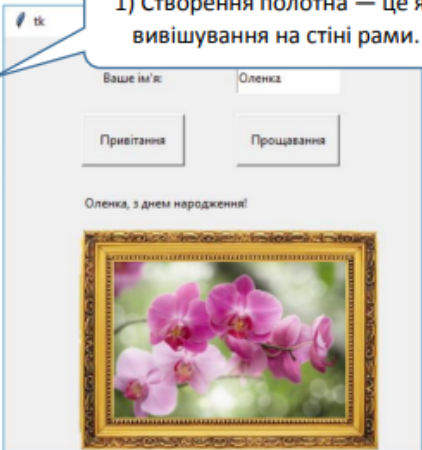
3) Розміщуємо картину в рамі.

1) Створення полотна — це як вивішування на стіні рами.

Ваше ім'я: Оленка

Привітання Прощання

Оленка, з днем народження!

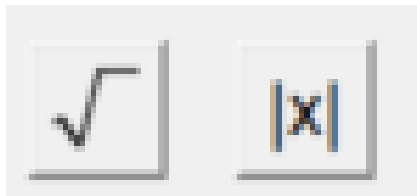


`flower_image=PhotoImage(file='flowers.png')`

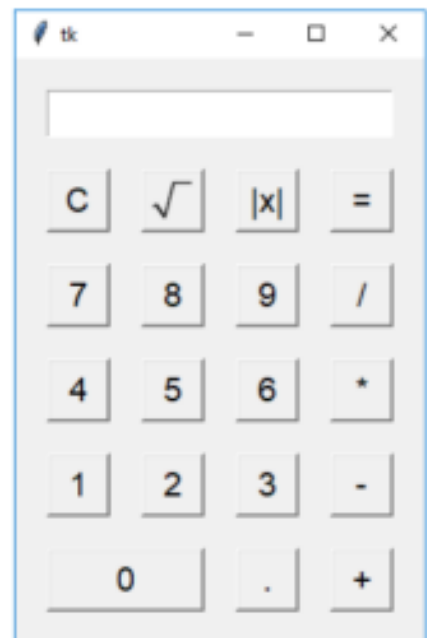
Присвоєння змінній **a** зображення, що міститься у файлі **mi.png**

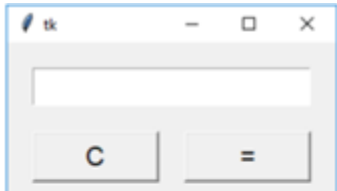
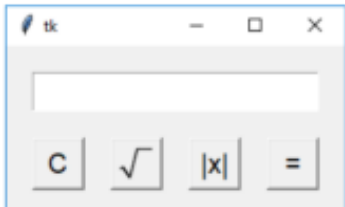
`a=PhotoImage(file='mi.png')`

Додамо кнопки видобування квадратного кореня та визначення модуля числа



На кнопці $\sqrt{\quad}$ — виведено зображення **radical.png**



Був початковий код	Став код
#BC BC=Button(text="C",font="14") BC.place(x=20,y=70,width=100,height=40) #Equal Equal=Button(text="=",font="14") Equal.place(x=140,y=70,width=100,height=40)	#BC BC=Button(text="C",font="14") BC.place(x=20,y=70,width=40,height=40) #Equal Equal=Button(text="=",font="14") Equal.place(x=200,y=70,width=40,height=40) #Radical Radical =Button() Radical.place(x=80,y=70,width=40,height=40) #Modul Modul.place(x=140,y=70,width=40,height=40)
	

#BC

BC=Button(text="C",font="14")

BC.place(x=20,y=70,width=40,height=40)

#Equal

Equal=Button(text="=",font="14")

Equal.place(x=200,y=70,width=40,height=40)

#Radical

Radical =Button()

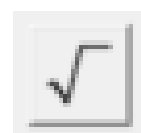
Radical.place(x=80,y=70,width=40,height=40)

#Modul

Modul.place(x=140,y=70,width=40,height=40)

Для кнопки видобування квадратного кореня не існує спеціального знаку на клавіатурі.

На кнопці – виведено зображення **radical.png**



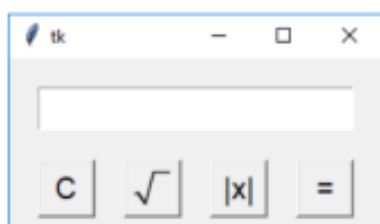
`img=PhotoImage(file='radical.png')` (завантаження зображення з файлу)

`Radical=Button(image=img)` (присвоєння зображення властивості `image`)

Зображення в тій же папці, що і програма

Вправа 1. Відкрийте файл програми створення калькулятора з попереднього уроку. Змініть програмний код, додавши кнопки видобування кореня та обчислення модуля числа.

Для створення кнопки видобування квадратного кореня скористайтеся зображенням **radical.png**, яке можна завантажити з онлайнового підручника.



Для обчислення квадратного кореня та модуля числа треба підключити вбудований модуль **math**, що містить набір математичних функцій:

from math import*

Знаходження кореня квадратного $\sqrt{x}$	sqrt(x)
Знаходження модуля числа $ x $	abs(x)

$y = \sqrt{a}$	математика	$y = \sqrt{3 + x }$	$y = x $
y=sqrt(a)	Python	y=sqrt(3+abs(x))	y=abs(x)

Створення програмного коду для видобування квадратного кореня

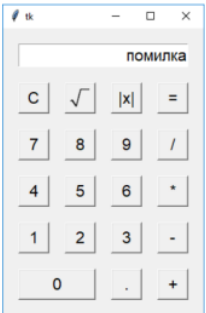
Оскільки для видобування квадратного кореня треба тільки одне число, то ця дія буде виконуватися відразу після натискання кнопки із зображенням кореня (без =)



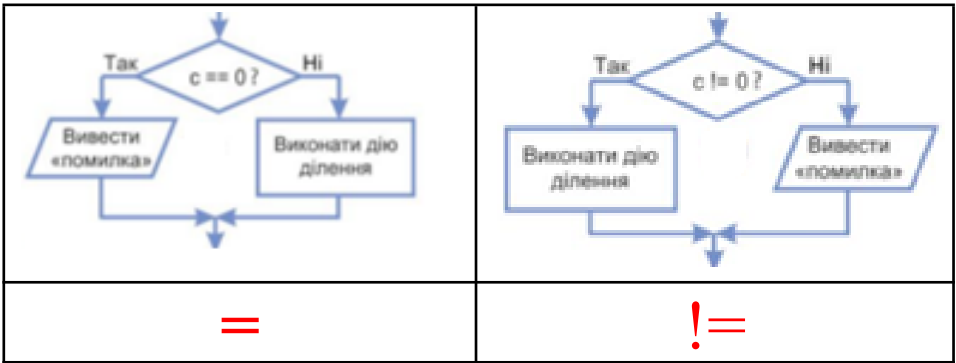
Присвоєння змінній c квадратного кореня значення змінної a	Присвоєння змінній c модуля значення змінної a
c=sqrt(a)	c=abs(a)

Обмеження на видобування квадратного кореня – із від’ємного числа не можна видобувати $\sqrt{}$

Додавання програмного коду, в разі видобування квадратного кореня із від’ємного числа – виводитиме повідомлення помилка



Алгоритм ділення з перевіркою рівності дільника нулю



```
if c!=0:  
    a=a/c  
    ent.insert(END,a)  
else:  
    ent.insert(END,'помилка')
```



Програмний код	<pre> if a>=0: c=sqrt(a) ent.insert(END,c) else: ent.insert(END,"помилка") </pre>	<pre> if a<0: ent.insert(END,"помилка") else: c=sqrt(a) ent.insert(END,c) </pre>
----------------	--	---

Для перевірки роботи кнопки, під час введення від'ємного числа вводити знак – (мінус) з клавіатури (наприклад: -9)



Вправа 2. Підключіть модуль **math** та запишіть програмний код для обробника події натискання кнопки зі знаком кореня.



Вправа 3. Запишіть програмний код для обробника події натискання кнопки, що обчислює модуль числа.

Так само, як і для обчислення кореня числа, створіть окрему функцію і прив'яжіть її до події натискання кнопки модуля.



Дію **ділення** можна виконувати не над будь-якими числами

Вкажіть команду, замість якої потрібно вставити один із двох наведених фрагментів коду.

```
def Equal_click():  
    global a,b  
    c=float(ent.get())  
    ent.delete(0,END)  
    if b=="+":  
        a=a+c  
    if b=="-":  
        a=a-c  
    if b=="*":  
        a=a*c  
    if b=="/":  
        a=a/c  
    ent.insert(END,a)
```

```
if c!=0:  
    a=a/c  
    ent.insert(END,a)  
else:  
    ent.insert(END, 'помилка')
```

```
if c==0:  
    ent.insert(END, 'помилка')  
else:  
    a=a/c  
    ent.insert(END,a)
```

Для опрацювання ділення треба використати

вкладену вказівку розгалуження



Один відступ (клавіша **Tab**) – перший рівень вкладеності

Два відступи (**Tab, Tab**) – другий рівень вкладеності

Вправа 4. Запишіть програмний код в обробнику події натискання кнопки "=", що опрацьовує дію ділення. Запустіть програму та перевірте, як вона працює у разі виконання ділення, у тому числі на нуль.



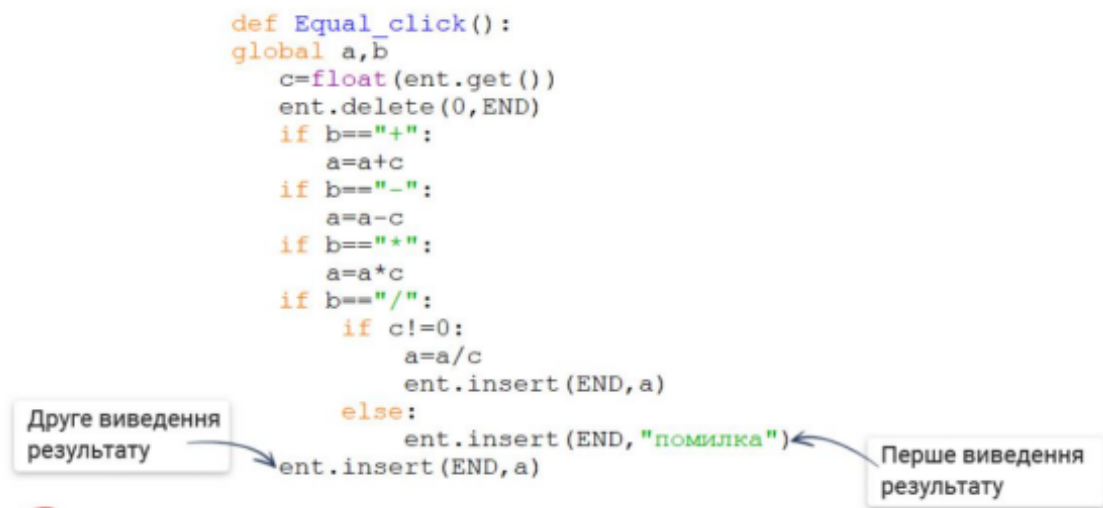
*Відступи в коді робити за допомогою **Tab***

В разі ділення на нуль – програма працює не зовсім так, як треба

Тому що, у коді дублюється команда виведення результату

```
def Equal_click():
    global a,b
    c=float(ent.get())
    ent.delete(0,END)
    if b=="+":
        a=a+c
    if b=="-":
        a=a-c
    if b=="*":
        a=a*c
    if b=="/":
        if c!=0:
            a=a/c
            ent.insert(END,a)
        else:
            ent.insert(END,"помилка")
    ent.insert(END,a)
```

у коді дублюється команда виведення результату



РЕЗУЛЬТАТ ДІЛЕННЯ ВИВОДИТЬСЯ ДВІЧІ

(див перше виведення результату, друге виділення результату)

Для вирішення проблеми дублювання результату – **треба розташувати команду виведення результату після дії кожного знаку**

```
def Equal_click():
    global a,b
    c=float(ent.get())
    ent.delete(0,END)
    if b=="+":
        a=a+c
        ent.insert(END,a)
    if b=="-":
        a=a-c
        ent.insert(END,a)
    if b=="*":
        a=a*c
        ent.insert(END,a)
    if b=="/":
        if c!=0:
            a=a/c
            ent.insert(END,a)
        else:
            ent.insert(END,"помилка")
    ent.insert(END,a)
```

Вправа 5. Змініть програмний код таким чином, щоб у разі ділення на 0 виводилося лише повідомлення "помилка".



Останній рядок коду перемістіть у вказані місця.

```
def Equal_click():  
    global a,b  
    c=float(ent.get())  
    ent.delete(0,END)  
    if b=="+":  
        a=a+c  
    if b=="-":  
        a=a-c  
    if b=="*":  
        a=a*c  
    if b=="/":  
        if c!=0:  
            a=a/c  
            ent.insert(END,a)  
        else:  
            ent.insert(END,"помилка")  
    ent.insert(END,a)
```

Diagram showing the movement of the last line of code (`ent.insert(END,a)`) to three locations:

- Before the first `if` statement (`if b=="+":`).
- Before the second `if` statement (`if b=="-":`).
- Before the third `if` statement (`if b=="*":`).



(Я)К (Г)ОП(Х) (КІ)НЬ