

Й.Я. РИВКІНД, Т.І. ЛИСЕНКО,
Л.А. ЧЕРНІКОВА, В.В. ШАКОТЬКО

ІНФОРМАТИКА

8

Властивості та методи елементів керування. Кнопка

За навчальною програмою 2017 року



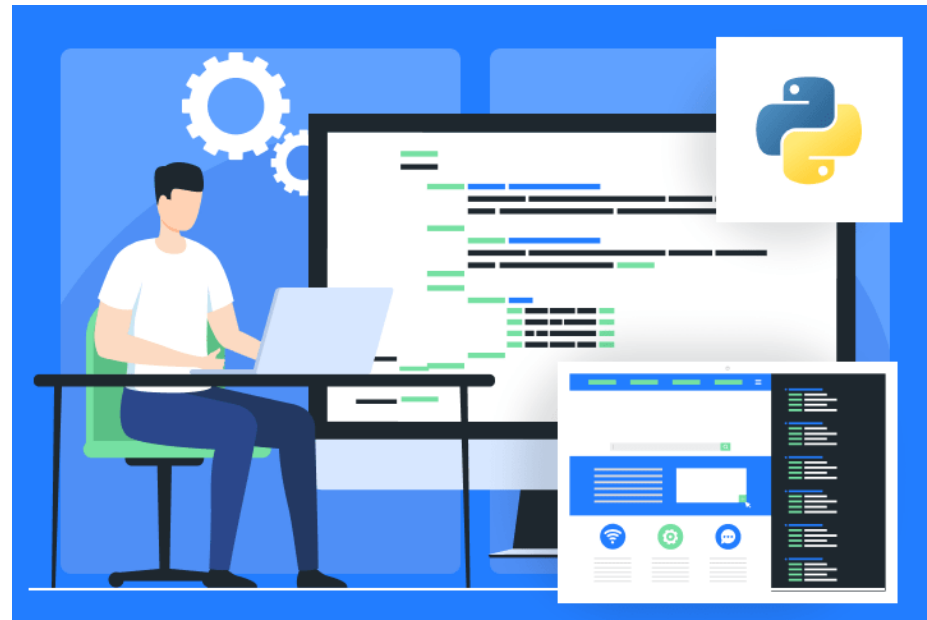
teach-inf.com.ua



1. Які властивості вікна ви знаєте? Як змінити значення його властивостей до початку виконання проєкту й під час його виконання?

2. Які події можуть відбуватися з вікном? У чому полягає кожна з них?

3. Що таке обробник події? Як його створити? Як його виконати?



Одним з компонентів (елементів керування), який можна використати в проєктах, є **кнопка**.





Щоб розмістити кнопку у вікні, потрібно:

1. Створити новий об'єкт типу `Button`, пов'язати його зі змінною, яка визначатиме ім'я об'єкта.

```
button = Button()
```

2. Установити значення властивостей кнопки або залишити їх за замовчуванням.

```
button = Button(text='Змінити', width='15',)
```

3. Розмістити створений об'єкт у вікні.

```
button.pack()
```



Для створення кнопки та пов'язування її зі змінною використовують команду вигляду:

```
<ім'я_кнопки> = Button()
```

```
<ім'я_кнопки> =  
Button(<Набір властивостей  
та їх значень>)
```

буде створено кнопку зі значеннями властивостей висота, ширина і колір, які встановлено за замовчуванням, без напису на ній

**буде створено кнопку, значення властивостей якої задано в цій команді. У дужках через кому вказують пари пари
«властивість» = «значення»**



Для кнопки можуть бути встановлені значення таких властивостей:

text

текст на кнопці;

width

ширина (кількість символів, які можуть бути розміщені поруч по горизонталі);

height

висота (кількість символів, які можуть бути розміщені поруч по горизонталі);

bg

колір фону (значенням може бути назва кольору англійською);

fg

колір символів тексту;



(Продовження...) Властивості кнопки

font

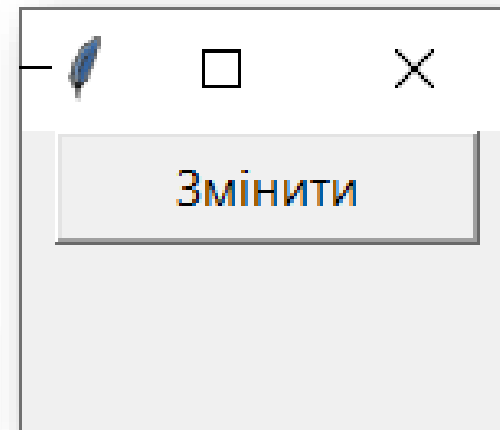
шрифт символів тексту на кнопці
(значення може містити назву шрифту,
розмір символів і накреслення);

state

стан кнопки (**normal** - звичайний, **disable** -
недоступний)

Наприклад, кнопка з іменем **button** з
текстом **Змінити** завширшки 15 символів
може бути створена командою:

```
button = Button(text = 'Змінити', width = 15)
```



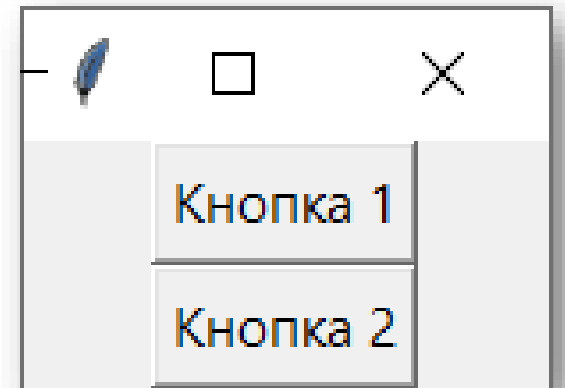


Для розміщення кнопки у вікні можуть бути використані такі методи:

`pack()`

Кнопка автоматично розміщується під рядком заголовка вікна з однаковими відступами від лівої та правої меж вікна. Якщо у вікні розміщується кілька кнопок, то вони будуть відображатися одна під одною без відступів між ними.

```
button1 = Button(text='Кнопка 1')  
button1.pack()  
button2 = Button(text='Кнопка 2')  
button2.pack()
```

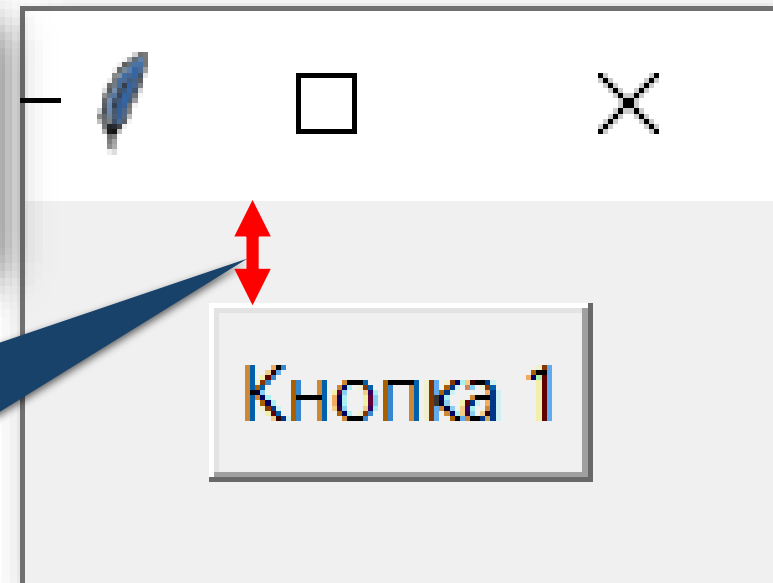


Якщо потрібно розмістити кнопку з деяким відступом від рядка заголовка або від попередньої кнопки, то в дужках можна указати цей відступ у пікселях як значення властивості *padу*.

Наприклад, після виконання команди:

```
button1 = Button(text='Кнопка 1')  
button1.pack(pady=20)
```

Кнопку буде розміщено з відступом 20 пікселів від верхньої межі вікна або від попередньої кнопки;





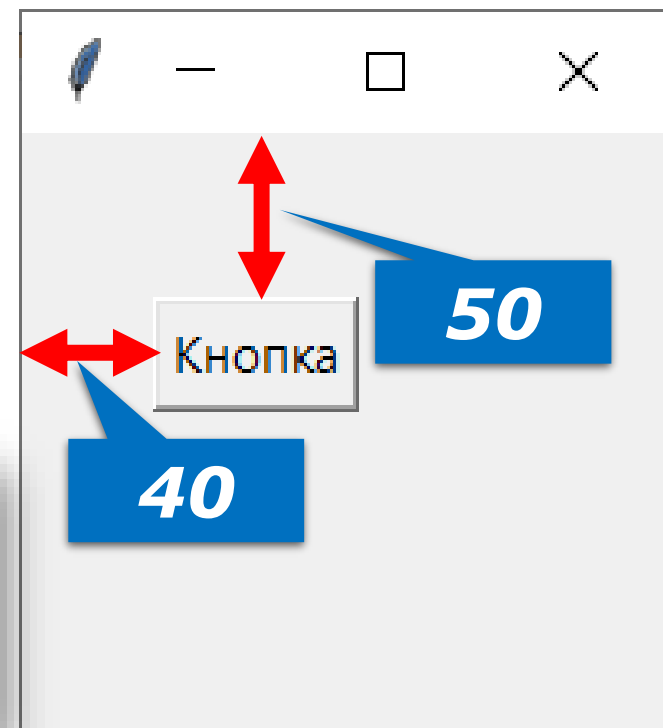
(Продовження...) **Методи кнопки**

***place(x = «відступ від лівої межі вікна»,
y = «відступ від верхньої межі вікна»)***

Верхня ліва вершина кнопки розміщується в точці вікна із заданими координатами *x* та *y*.

Наприклад,

```
button = Button(text='Кнопка')  
button.place(x = 40, y = 50)
```





Під час виконання проєкту може виникнути потреба зробити кнопку невидимою.

Для цього, залежно від методу розміщення кнопки, можна використати методи:

`pack_forget()`

або

`place_forget()`

Щоб зробити кнопку знову видимою, слід застосувати метод:

`pack()`

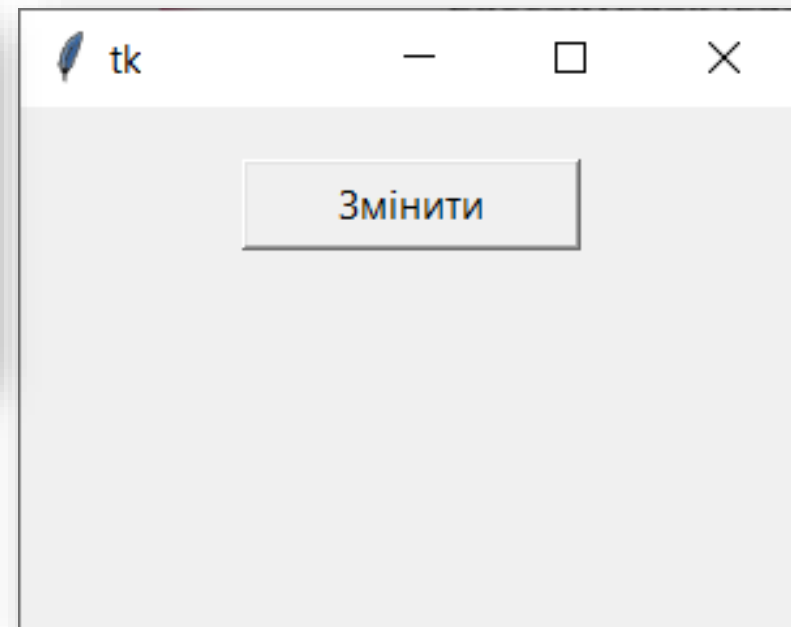
або

`place()`



Текст проєкту зі створення та розміщення кнопки в вікні та результат його виконання.

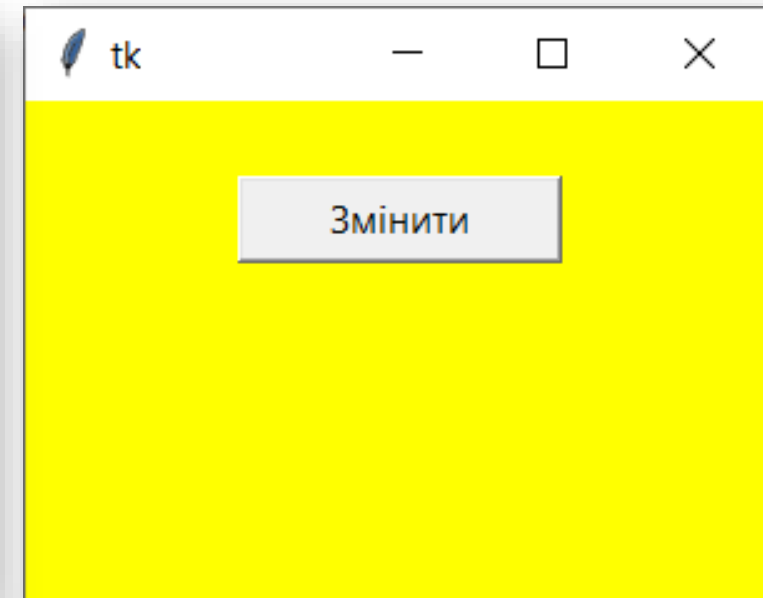
```
from tkinter import *
root = Tk()
root.geometry('300x200+400+400')
button = Button(text='Змінити', width='15')
button.pack(pady=20)
```





Пов'язати з кнопкою обробник будь-якої події можна так само, як з вікном, використовуючи метод `bind()`. Приклад обробника події `Click` для кнопки, яка містить команду встановлення жовтого кольору вікна.

```
from tkinter import *
def click(event):
    root['bg'] = 'yellow'
root = Tk()
root.geometry('300x200+350+200')
button = Button(text='Змінити', width='15')
button.pack(pady=30)
button.bind('<1>', click)
```





Для кнопки подією за замовчуванням є клацання лівої кнопки миші. Тому пов'язати з кнопкою метод опрацювання саме цієї події можна простіше:

➤ **під час створення кнопки вказати ім'я відповідного обробника події як значення атрибуту `command`;**

```
button = Button(text=Кнопка', command=click)
```

➤ **обробник події з указаним іменем описати перед створенням кнопки; у заголовку не потрібно вказувати посилання на подію, що відбулася, – `event`.**



На малюнку наведено приклад обробника події **Click** для кнопки – процедури з іменем **click()**, яка містить команди:

```
from tkinter import *
def click():
    button['bg'] = 'red'
    button['font'] = 'Arial 14'
    button['text'] = 'Змінюються значення властивостей кнопки'
    button['state'] = 'disable'
root = Tk()
root.geometry('600x400+350+200')
button = Button(text='Вибере мене', command=click)
button.pack(pady=30)
```

Установлення для кнопки
червоного кольору фону

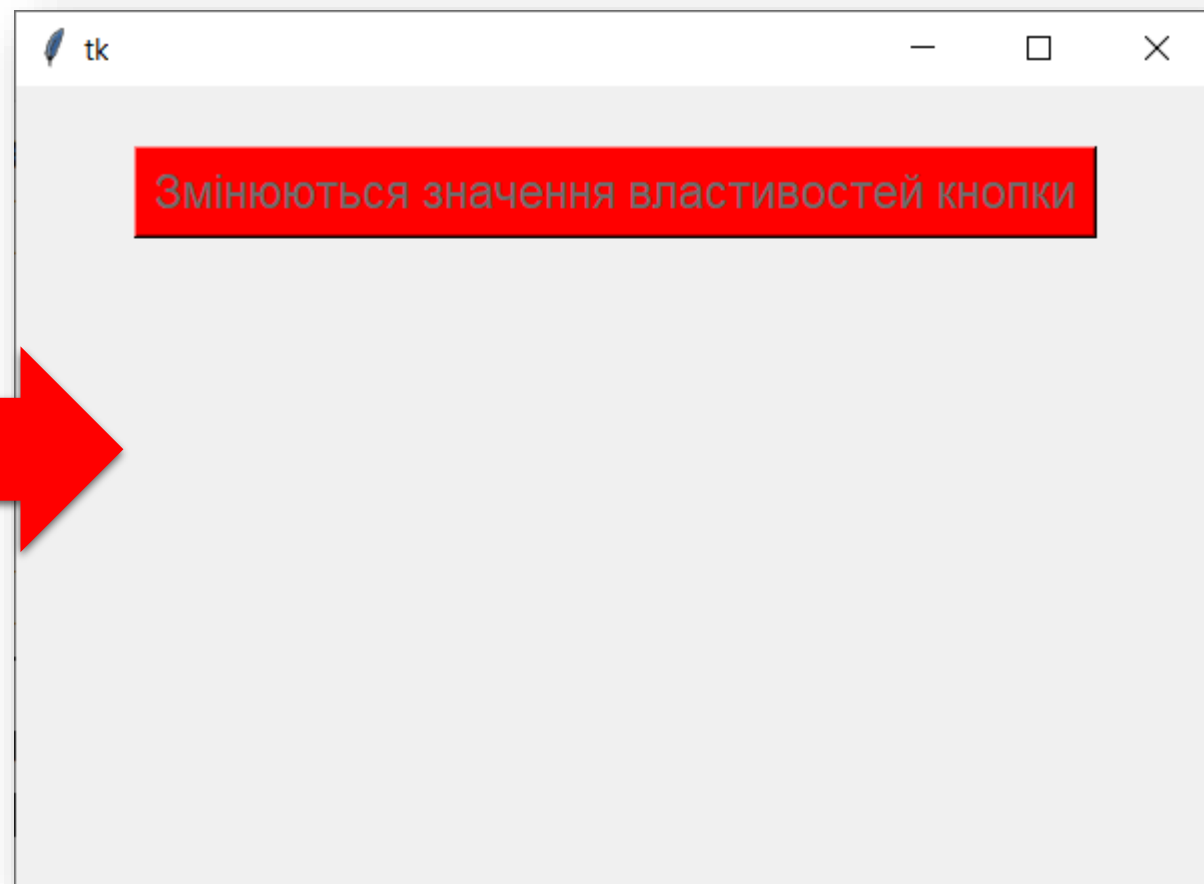
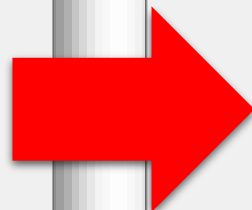
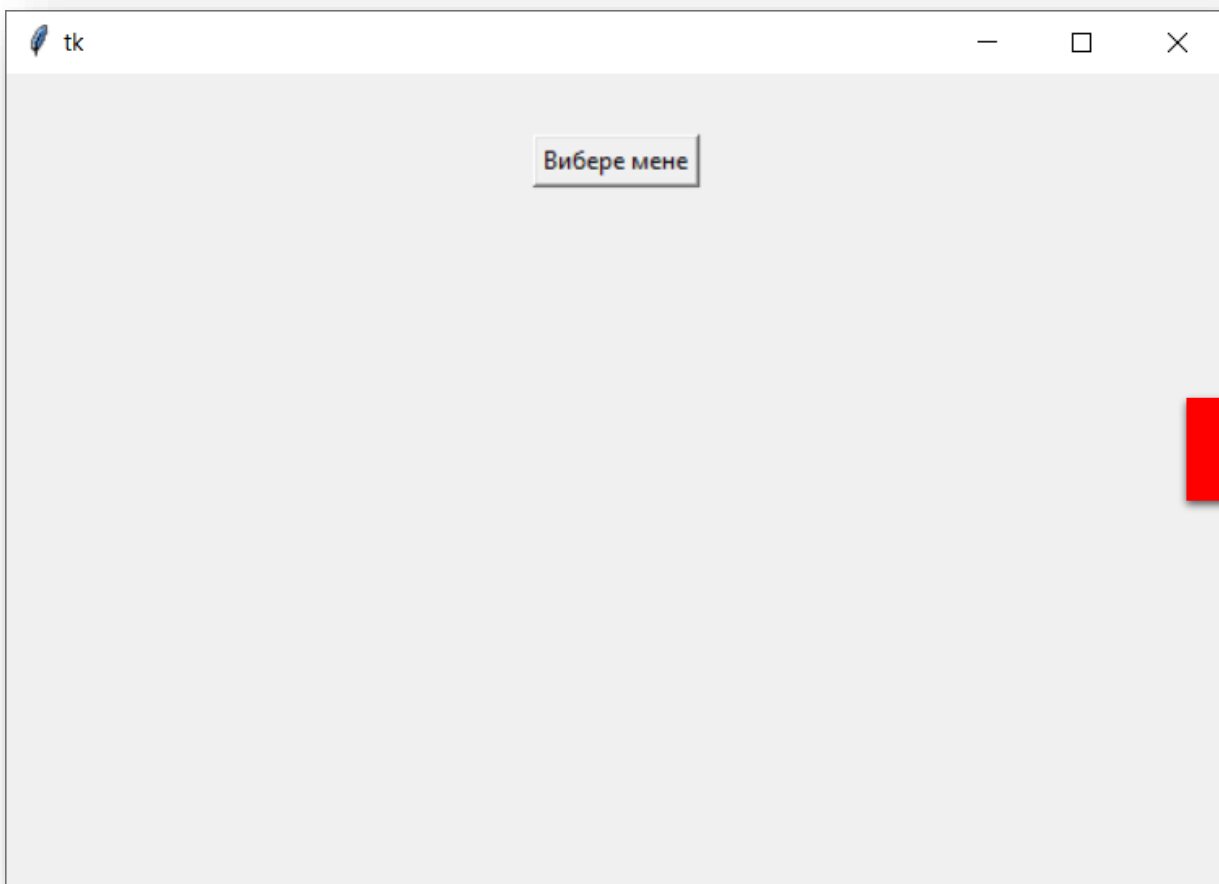
шрифт Arial розміром 14

установлення для
кнопки стану недоступна

виведення
на кнопці
вказаного
тексту



Приклад обробника події **Click**





1. Як створити кнопку?

2. Які властивості кнопки ви знаєте? Що визначає кожна з них?

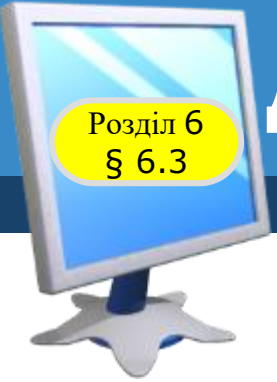
3. Як задати значення властивостей кнопки?

4. Як розмістити кнопку у вікні?

5. Якими способами можна призначити для кнопки обробники подій?

6. Як призначити для кнопки обробник події *Click*?





Домашнє завдання



Проаналізувати
§ 6.3, ст. 173-180