

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

ІНФОРМАТИКА

8

Обробники подій, пов'язаних з елементами керування

За навчальною програмою 2017 року



Урок 42



Під час виконання проєкту з об'єктами можуть відбуватися **події. Так, наприклад, з вікном можуть відбуватися такі події:**

Click

вибір будь-якої точки вікна;

DbClick

установлення вказівника в довільну точку вікна і подвійне клацання лівої кнопки миші;

Close

закривання вікна;

Create

створення (відкриття) вікна;

***Mouse
Move***

переміщення вказівника у вікні виконання проєкту та ін.

*Події можуть відбуватися з клавіатурою. Такою подією, наприклад, може бути подія **KeyPress** (англ. *key press* – натиснення клавіші) – натиснення будь-якої клавіші на клавіатурі.*

Події можуть відбуватися й з іншими об'єктами:

кнопка

напис

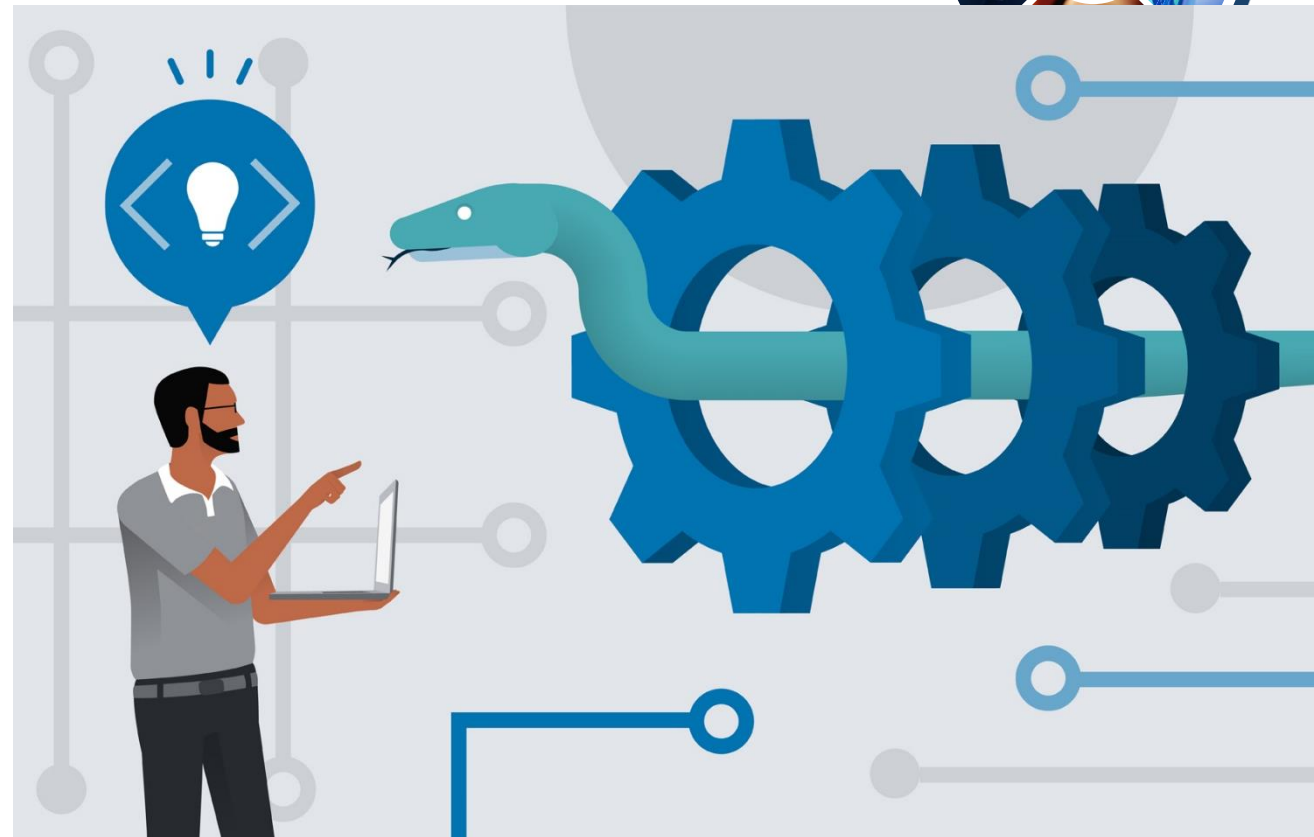
поле та ін.

Ми їх вивчатимемо в наступних пунктах.





*З будь-якою подією, яка може відбутися з вікном, можна пов'язати **фрагмент проекту**, який буде виконуватися одразу після настання цієї події. Такий фрагмент проекту називається **обробник події**.*



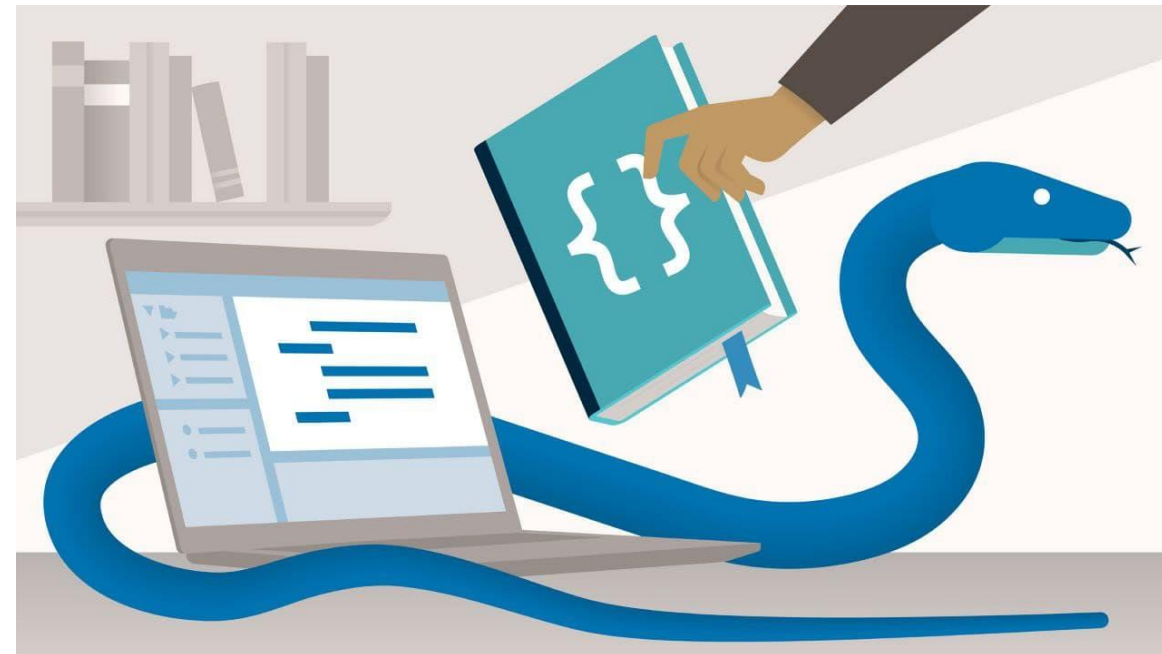
*Фрагмент проекту, який пов'язаний з певним об'єктом, називається **методом цього об'єкта**.*



У мові **Python** обробник події для вікна створюють як окрему **підпрограму** та пов'язують її з вікном, використовуючи метод **bind** (англ. *bind* - в'язати, пов'язувати).



Підпрограма – це окремо виділена частина проекту, яка має своє ім'я та яку можна за цим іменем запуснути на виконання.





У **Python** розрізняють підпрограми двох видів:

процедури

функції

підпрограма, результатом виконання якої є створення компонента (вікна, кнопки та ін.) або змінення значень властивостей існуючого, виведення певного тексту, створення зображення тощо. Зазвичай обробники подій є процедурами.

підпрограма, результатом виконання якої є одне або кілька значень: числа, рядки тексту та ін.



Метод – це процедура або функція, що пов'язана з певним об'єктом.

У загальному вигляді команда пов'язування обробника події з вікном, що має ім'я `root`, має такий вигляд:

```
root.bind('<подія>', <ім'я_обробника_події>)
```





Позначення в мові **Python** деяких подій, які можуть відбутися з вікном.

Позначення подій у мові Python

Подія

Позначення в мові Python

Click – клацання лівої кнопки миші

<**Button-1**> або <**1**>

DbClick – подвійне клацання лівої кнопки миші

<**Double-Button-1**> або
<**Double-1**>

MouseMove – рух вказівника по вікну

<**Motion**> (англ, *motion* - рух)



(Продовження...)

Позначення подій у мові Python

Подія

Позначення в мові Python

KeyPress – натискання будь-якої клавіші на клавіатурі

<KeyPress>

Create – створення вікна

<Expose> (англ, expose – розкривати)

Close – закривання вікна

<FocusOut> (англ, focus out – вихід фокуса)



Наприклад, якщо обробнику події **Click** (клацання лівої кнопки миші) надати ім'я **click**, то команда пов'язування цієї процедури з вікном з іменем **root** має такий вигляд:

```
root.bind('<1>', click)
```

Загальний вигляд обробника події такий:

Рядок заголовка

```
def <ім'я_обробника_події> (event):
```

```
    <команди, які будуть виконуватися під час настання події>
```



Наприклад, для обробника події з іменем **click** рядок заголовка буде таким:

```
def click (event):  
    root ['bg'] = 'green'
```

Команди обробника події потрібно записувати з відступом праворуч від лівої межі вікна редактора коду.

Відступ утворюється автоматично, якщо після символу «**:**» натиснути клавішу **Enter**, або його можна зробити самотійно, натиснувши один раз клавішу **Tab**.



Якщо потрібно, щоб у результаті настання події змінилися значення певних властивостей вікна, обробник події повинен містити команди змінення значень цих властивостей. Це можуть бути команди застосування до вікна деяких методів, наприклад:

geometry()

title()

*команда
присвоювання*

Загальний вигляд команди присвоювання такий:

<ім'я об'єкта>['<властивість>'] = «значення»

Знак присвоювання



Наприклад, виконання команди присвоювання:

```
root ['bg'] = 'green'
```

*Установить зелений колір вікна з іменем **root**.
Застосувавши метод:*

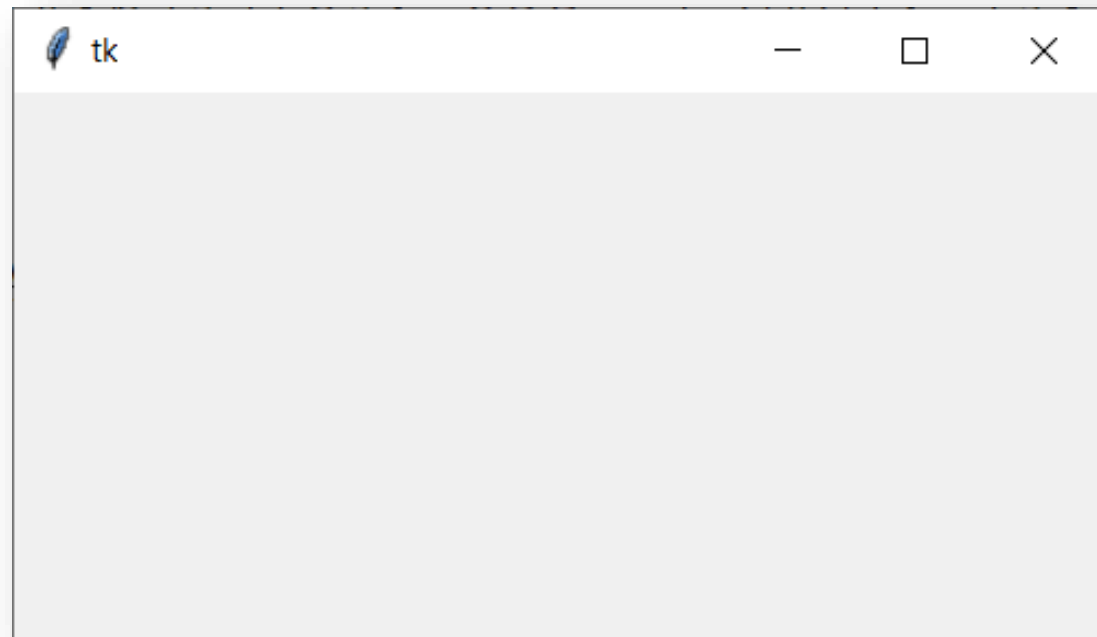
```
root.title('Зміна значень властивостей  
вікна')
```

Можна вивести вказаний у лапках текст у рядку заголовка вікна.



А для того щоб розміри вікна стали 400 x 200 пікселів, відстань від лівої межі екрана – 500 пікселів, від верхньої – 70 пікселів, потрібно застосувати метод

```
root.geometry('400x200+500+70')
```





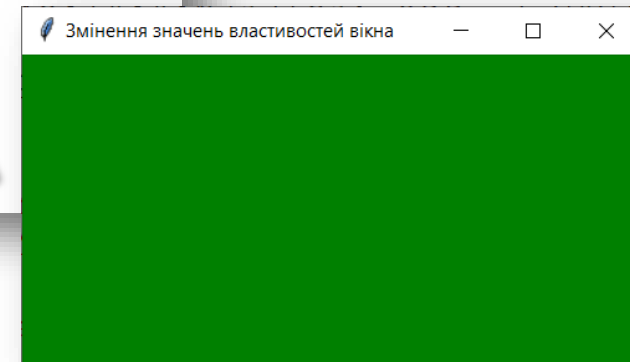
*Обробник події записують перед його використанням, найчастіше на початку тексту проєкту. Запис команд після обробника події повинен розпочинатися від лівої межі вікна. Для переміщення курсора в цю позицію потрібно використати клавішу **Backspace**. Так, після обробника події слід записати команди створення вікна та пов'язування з ним обробника події:*

```
root = Tk()  
root.bind('<1>', click)
```



На малюнку наведено приклад процедури **click** – обробника події **Click** для вікна, що складається з вищеописаних команд, під час виконання якої зміняться значення властивості **bg**, текст заголовка, розміри і положення вікна, і результат її виконання.

```
from tkinter import *
def click (event):
    root ['bg'] = 'green'
    root.title('Змінення значень властивостей вікна')
    root.geometry('400x200+500+70')
root = Tk()
root.bind('<1>', click)
```





Іноді треба під час опрацювання події встановити нове значення властивості об'єкта, що обчислюється на основі поточного значення цієї самої властивості.

Наприклад, потрібно, щоб після клацання лівої кнопки миші вікно опустилося на 50 пікселів, тобто слід збільшити відстань вікна від верхньої межі екрана на 50 пікселів.



У такому разі треба отримати поточне значення відстані, змінити його на 50 і запам'ятати результат в окремій змінній, щоб встановити нове положення вікна.



Для отримання поточних значень розмірів і положення вікна використовують такі методи вікна:

winfo_width()

(англ. *window information* - інформація про вікно) ширина вікна;

winfo_height()

висота вікна;

winfo_x()

відстань лівої межі вікна від лівого краю екрана;

winfo_y()

відстань верхньої межі вікна від верхнього краю екрана.



Наприклад, щоб дізнатися поточну відстань вікна з іменем **root** від верхньої межі екрана та збільшити її на 50, потрібно виконати команду:

```
t = root.wininfo_y() + 50
```

Щоб установити нове значення, потрібно в рядок визначення положення та розмірів вікна методу **geometry()** замість числа підставити значення змінної. Для цього на місці, де потрібно вставити значення змінної, увести символи **{}** і до рядка застосувати метод підстановки **format()** з іменем змінної, відокремивши його крапкою:

```
root.geometry('300x200+500+{}'.format(t))
```




Якщо потрібно підставити в рядок значення кількох змінних, то на всіх відповідних місцях потрібно ввести символи `{}`, а у методі `format()` перерахувати імена змінних через кому.





Наприклад, для того щоб після настання події клацання лівої кнопки миші вікно опустилося на 50 пікселів униз, а його ширина збільшилася на 40 пікселів, можна виконати проєкт:

```
from tkinter import *
def click (event):
    t=root.wininfo_y()+50
    w=root.wininfo_width()+40
    root.geometry(' {}x320+200+{} '.format(w,t) )
root = Tk()
root.geometry('480x320+200+200')
root.bind('<1>', click)
```



Після запуску проєкту на виконання потрібно дочекатися відкриття вікна і вибрати будь-яку його точку. Тим самим відбудеться подія **Click для вікна та буде виконана процедура – обробник події, результатом виконання якої буде змінення положення та розмірів вікна.**



Звертаємо вашу увагу, що в одному проєкті можуть бути створені процедури опрацювання кількох різних подій з одним і тим самим або з різними об'єктами.



1. Які події можуть відбуватися з вікном?

2. Що називають обробником події?

3. Як пов'язати з подією її обробник?

4. Що називають методом об'єкта?

5. Що називають процедурою?

6. Як оформити обробник події в Python?

7. Якими командами можна змінити значення властивостей об'єкта? Який їх загальний вигляд? Наведіть приклади таких команд.



Розділ 6
§ 6.2

Домашнє завдання



Проаналізувати
§ 6.2, ст. 160-165

Працюємо за комп'ютером

Розділ 6
§ 6.2



**Сторінка
172**



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

ІНФОРМАТИКА

8

Дякую за увагу!

За навчальною програмою 2017 року



Урок 42