

Інтерфейс користувача на мові Python. Створення вікон та налаштування їх властивостей

Кожна програма, має свій інтерфейс.

Інтерфейс – це сукупність засобів і правил, що забезпечують взаємодію пристроїв обчислювальної системи або програм. Існує 2 види інтерфейсу програм: **командний** та **графічний**. Що стосується командного інтерфейсу, то взаємодія користувача та комп'ютера відбувається за допомогою команд, які користувач вводить в командний рядок з клавіатури. А от графічний інтерфейс є більш зручнішим для користувача, адже взаємодія відбувається за допомогою кнопок, полів, та інших елементів управління.

Мова Python більше орієнтована на командний інтерфейс, але в ній існує спеціальний модуль за допомогою якого можна створити інтерфейс користувача.

Модуль, який нам знадобиться для створення графічного інтерфейсу, програми називається **tkinter**. За допомогою цього модуля можна виконувати як окремі графічні побудови, так і створити повноцінний графічний інтерфейс користувача. Отже, перший рядок коду в нашому новому файлі буде завжди виглядати так:

```
from tkinter import *
```

Після підключення модуля, створимо вікно. Для створення вікна використовується функція **Tk()**, але, так як таких вікон можна створити нескінченно багато, то кожному вікну присвоюється певне ім'я:

```
назва_вікна=Tk()
```

І ще дуже важлива річ: після створення вікна та його елементів управління потрібно вказати інтерпретатору, що ми закінчили працювати з вікном за допомогою метода **mainloop()**. Застосовується він так:

```
назва_вікна.mainloop()
```

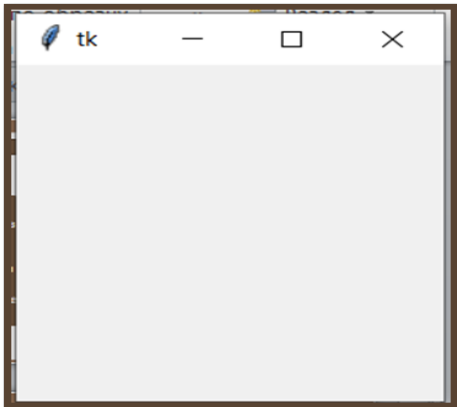
Приклад програми з використанням вікна:

```
from tkinter import * # підключення модулю
```

```
Window1=Tk() # створення вікна
```

```
Window1.mainloop() # закінчення роботи з вікном
```

Після запуску наше вікно виглядає так:



Основні властивості вікна:

title(“Заголовок”) – заголовок вікна. За замовчуванням заголовок вікна «tk»;

minsize(x,y) – мінімальний розмір вікна у пікселях, x- ширина, y – висота. Якщо не встановити цю властивість, то вікно не матиме обмежень у зменшенні;

maxsize(x,y) – максимальний розмір вікна у пікселях, x- ширина, y – висота. Якщо не встановити цю властивість, вікно може приймати повні розміри екрану користувача;

geometry(“400×200+450+150”) – розміри та положення вікна відносно розширення екрана у пікселях. 400 – ширина, 200-висота, 450- відстань від лівого краю екрану, 150 – відстань від верхнього краю екрану. Параметри відступу (відстань від лівого краю екрану та відстань від верхнього краю екрану) не є обов'язковими, вони додаються за бажанням.

Зверніть увагу! Всі числа були введені як приклад;

resizable(x,y) – чи може користувач змінювати розміри вікна і на скільки: x – ширина, y – висота. Для заборони змінення розмірів вікна замість параметрів встановити нулі.

Це основні властивості, які ви будете використовувати у навчанні для створення простих проектів за допомогою мови програмування Python. До вікна вони застосовуються таким чином:

```
назва_вікна.властивість(параметри)
```

Але наприклад, для того, щоб застосувати певний колір фону нашого вікна виконується інша конструкція:

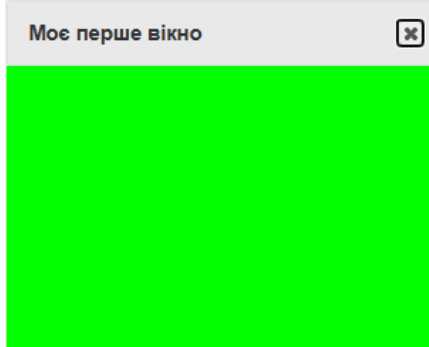
```
назва_вікна[“bg”]=“колір”
```

Bg скорочено від background (фон), а параметром є назва кольору на англійській мові взята в лапки.

Ну ось і все, що вам потрібно знати на цьому етапі. Ну а тепер, давайте розглянемо приклад завдання. Отже, завданням буде: створити вікно світло-блакитного кольору зі сталими розмірами: ширина 600, висота 500 та з заголовком «Мое перше вікно».

Код даної програми буде виглядати наступним чином:

```
from tkinter import * # підключення модулю
Window1=Tk() # створення вікна
Window1.title("Мое перше вікно")
Window1["bg"]="green" # вибір кольору вікна
Window1.mainloop() # закінчення роботи з вікном
А ось так виглядатиме готове вікно:
```



Визначений розмір вікна і його розміщення

```
from tkinter import * # підключення модулю
Window1=Tk() # створення вікна
Window1.title("Мое перше вікно")
Window1.geometry("700x300+100+0")
Window1["bg"]="yellow"
Window1.mainloop() # закінчення роботи з вікном
```

