**Тема. Складання та виконання алгоритмів з графічним відображенням даних**

**Мета:** навчити додавати і використовувати графічні об’єкти для створення інтерфейсу користувача; розвивати уміння і навички створення та налагоджування об’єктно-орієнтованих програм; виховувати культуру термінологічного мовлення, допитливість, наполегливість.

**Хід уроку**

**І Організаційний момент**

**ІІ Повторення вивченого раніше**

|  |  |  |
| --- | --- | --- |
| **Умовний оператор**  if (умова):  дія 1  else:  дія 2 | **Складні умови**  if ((умова 1)or(умова 2) or(умова 3)):  дія 1  if ((умова 4)or(умова 5) or(умова 6)):  дія 2 | **Вкладені умови**  if(умова 1) and (умова 2):  дія 1  elif  ((умова 3)or(умова 4)or(умова 5))and(умова 6):  Дія 2  else:  дія 3 |
| if (s < n):  print ("TAK")  else:  print ("HI") | if ((m==1)or(m==2) or(m==12)):  print('Зима')  if ((m==3)or(m==4) or(m==5)):  print('Весна') | if(a==1) and (b!=1):  print (x, 'гривня')  elif ((a==2)or(a==3)or(a==4))and(b!=1):  print (x, 'гривні')  else:  print(x, 'гривень') |

**Шкала чисел (range)**

Цикл з параметром у Python може перебирати значення різних типів: числа, букви, рядки, елементи списку тощо. Функція **range()** створює послідовність чисел у вигляді арифметичної прогресії (шкалу). Ця функція приймає три параметри:

* **початкове число**, яке включається у шкалу;
* **кінцеве число**, яке не включається у шкалу;
* **крок** або значення зміни, яке може бути додатнім чи від'ємним. Крок означає, що наступне значення шкали відрізняється від попереднього на значення цього кроку.

Можна задавати лише один або два параметри цієї функції, тоді решта параметрів прийматимуть значення за умовчанням. Для початкового числа це значення **0**, а крок рівний **1**. Для кінцевого числа значення за умовчанням не існує, тому це **обов'язковий параметр** функції **range()**.

**range (n)**

*Шкала чисел від 0 до n-1*

**range (k, n)**

*Шкала чисел від k до n-1*

**range (k, n, m)**

*Шкала чисел від k до n-1 з кроком m (може бути від'ємним)*

Наприклад, у функції **range(1,10,3)** початковим числом є **1** (включно), кінцевим числом є **10** (не включно), а кроком є **3**. Таким чином ця функція утворює послідовність чисел **[1,4,7]**.

Функція **range(3,-3,-2)** створює набір чисел **[3,1,-1]**, оскільки починає шкалу із значення **3**, і перебирає числа з кроком **-2** поки не досягне значення **-3** (не включаючи це число). Якби нам потрібно було включити й число **-3** у цю шкалу, функцію слід було би записати як **range(3,-4,-2)**.

**Команда циклу**

Цикл з параметром використовується, коли потрібно забезпечити повторення блоку команд вказану кількість разів. Параметр при цьому має набувати значень, що входять у певну шкалу (чисел, букв чи інших елементів).

**for** *змінна* **in** *шкала***:**

*команди*

**Змінна** - це параметр, **лічильник**, котрий змінює своє значення щоразу при виконанні команд циклу.

Одне виконання команд циклу називається **ітерацією**.

**Тіло циклу** складається з команд, котрі виконуються кілька разів.

Важливо дотримуватись правил синтаксичного оформлення циклів у Python, тобто рядок з командою for завершується **двокрапкою**, а команди тіла циклу записуються з **відступом**.

Виконання циклу проходить кілька етапів:

1) Визначається перелік елементів шкали

2) Виконуються команди тіла циклу

3) Відбувається перевірка: якщо у шкалі є невикористані елементи, то виконується пункт 2, інакше виконання передається наступній команді за межами циклу.

**Тіло циклу**

Тіло циклу може складатись з різних команд, зазвичай з арифметичних операцій та команд виведення print.

Задача 1. Надрукувати числа від 1 до **10**

for chyslo in range(1,**11**):

print(chyslo)

**Цикли з умовами**

**while** (*умова виконання*) :

*команди*

**break** *Припинення виконання циклу, перехід до наступної команди*

**Задача 1**. Вводити з клавіатури числа, поки не введемо нуль

a=int(input("Введіть число"))

while (a!=0):

a=int(input("Введіть число"))

print ("The End")

**Задача 2**. Генерувати випадкові числа від 1 до 6, поки не випаде 1

import random

a=random.randint(1,6)

while (a!=1):

print(a)

a=random.randint(1,6)

print ("The End")

**ІІІ Актуалізація опорних знань**

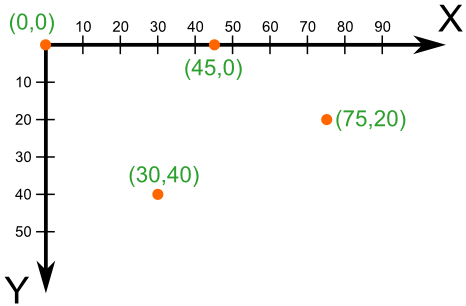
Фронтальне опитування

1. Що називають об’єктом у мові програмування?
2. Що таке властивості об’єкта?
3. Наведіть приклади властивостей об’єкта у Скретч
4. Які дії можна виконувати над об’єктом у Скретч?
5. Який інтерфейс у програмі Скретч?

ІV Мотивація навчальної діяльності

Проблемне питання. З інтерфейсом якого виду зручніше працювати користувачу? Як створити графічний інтерфейс у Пайтон?

**V Пояснення нового матеріалу**

**Міні-конспект: Складання та виконання алгоритмів з графічним відображенням даних в Python**

 Підєднайте модуль

from Tkinter import \*

window = Tk()

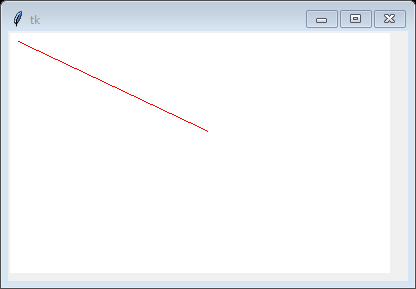
window.geometry('400x250')

c = Canvas(window, width = 380, height = 240, bg = 'white')

c.grid(column = 0, row = 0)

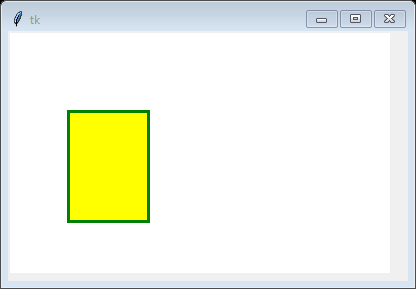
…….

window.mainloop()

*Геометричні примітиви:*

c.create\_line((х1,у1),(х2, у2), fill =‘колір')

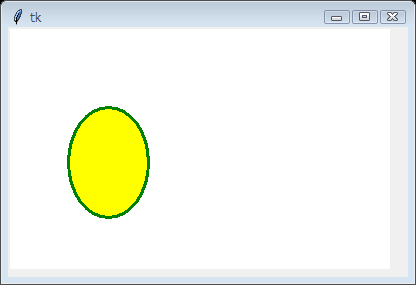
c.create\_line((10 , 10),(200 , 100), fill ='red')



c.create\_rectangle((х1, у1), (х2,у2), fill = ‘*колір прямокутника*', outline = ‘*колір межі*',

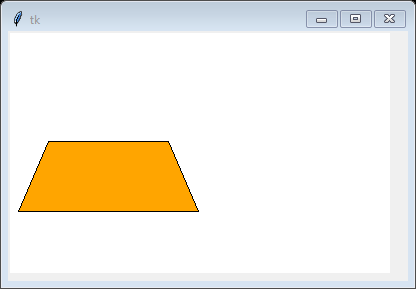
width = *товщина межі*, activedash = (5, 4))

c.create\_rectangle((60, 80), (140, 190), fill = 'yellow', outline = 'green', width = 3, activedash = (5, 4))

c.create\_oval((х1, у1), (х2,у2), fill = ‘*колір прямокутника*', outline = ‘*колір межі*',

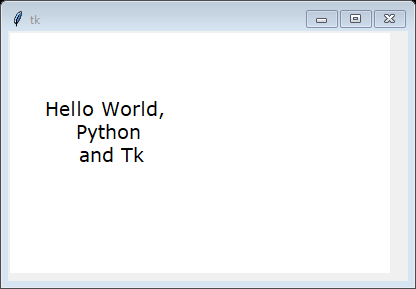
width = *товщина межі*, activedash = (5, 4))

c.create\_oval((60, 80), (140, 190), fill = 'yellow', outline = 'green', width = 3, activedash = (5, 4))



c.create\_polygon((х1,у1),(х2,у2),… (х,у), fill = ‘*колір фігури*',outline = ‘*колір межі*')

c.create\_polygon((40, 110),(160, 110), (190, 180), (10, 180), fill = 'orange', outline = 'black')

c.create\_text(100, 100, text = «Текст1,

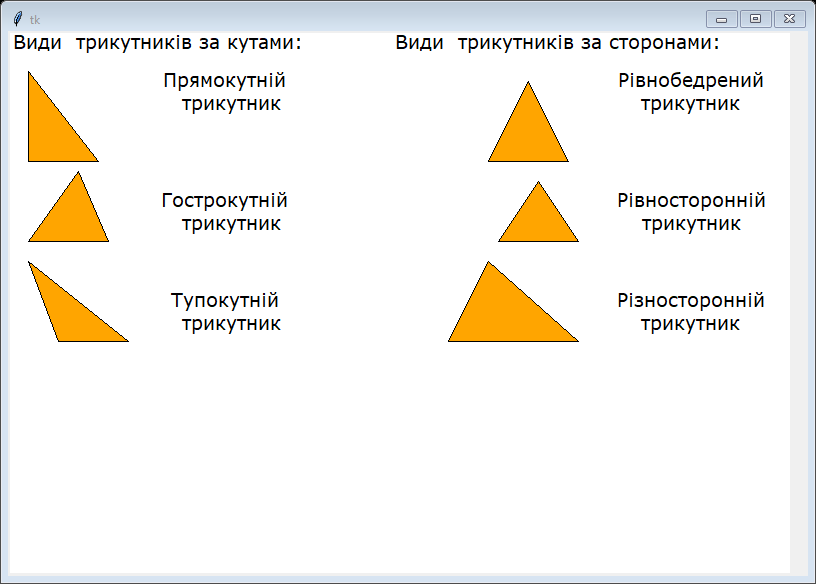
\n текст2 \n and текст0",justify = *вирівнювання вмісту тексту*, font = «*тип і розмір шрифта*")

c.create\_text(100, 100, text = "Hello World,

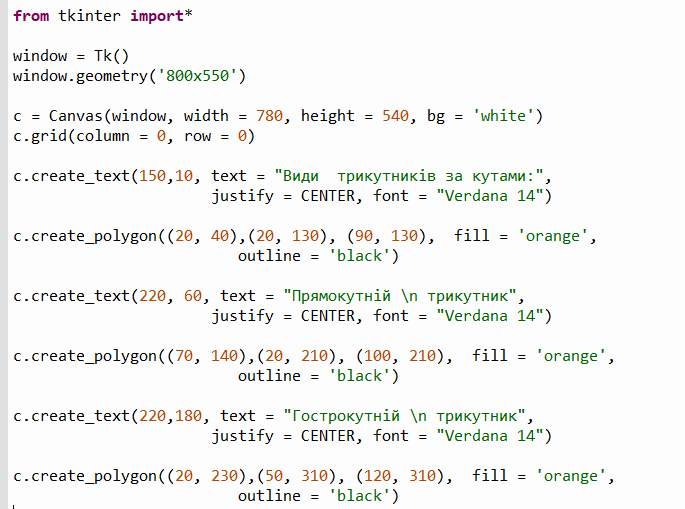
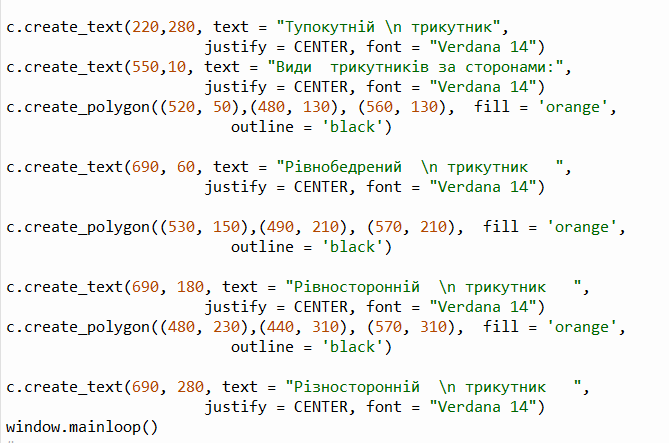
\n Python \n and Tk", justify = CENTER,

font = "Verdana 14")

Практичне завдання

Створіть програму в якій на формі відобразіть геометричні фігури і зробіть для них підписи. Наприклад:

Програма матиме вигляд:



**VІ Застосування знань, вироблення умінь**

* Створити проект за зразком
* Вправи для зняття зорової втоми

**VІІІ Підсумок уроку**

**ІХ Домашнє завдання**

Повторити тему «Графічні побудови»

Вивчити алгоритм створення графічного інтерфейсу користувача на мові Пайтон