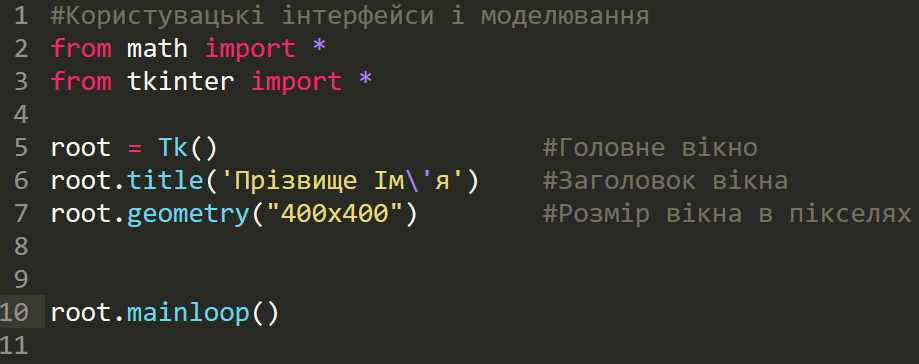
**Лабораторна робота №2**

**«Побудова графіків в Сanvas»**

Бібліотека Tkinter призначена для організації графічних інтерфейсів (GUI - Graphical User Interface) програм на Python, але завдяки наявності елемента графічного інтерфейсу Сanvas ( «полотно») в Tkinter можна використовувати елементи векторної графіки - криві, дуги, еліпси, прямокутники тощо, а також малювати прямі і криві за координатами, розрахованим за формулами.

Розглянемо задачу побудови графіка деякої функції по обчислюваним точкам за допомогою Tkinter.

1. Оскільки Tkinter дозволяє працювати з елементами GUI, створимо вікно заданого розміру, встановимо для нього заголовок і розмір вікна.



Наприкінці програми необхідно для нашого головного вікна root використати метод mainloop(), який говорить приблизно наступне: «ОК, я завершив роботу з настройки і тепер готовий до прийому та обробки подій».

**root.mainloop()** - забезпечує постійну роботу головного вікна і його об'єктів до моменту, коли воно буде закрито.

За параметри вікна відповідає метод **config()**

**root.config(background='steelblue', relief=RAISED, borderwidth=10)**

Розглянемо можливі параметри вікна:

height, width - розміри вікна

background(bg) - колір фону вікна

relief - вид рамки вікна. Може приймати значення

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FLAT** (плоска рамка) | **SUNKEN**(увігнута рамка) | **RAISED** (опукла рамка) | **GROOVE** (у вигляді жолобу) | **RIDGE** (у вигляді хребта) |
| https://disted.edu.vn.ua/media/images/porkhun/inf_8/l59_64/007.gif | https://disted.edu.vn.ua/media/images/porkhun/inf_8/l59_64/008.gif | https://disted.edu.vn.ua/media/images/porkhun/inf_8/l59_64/009.gif | https://disted.edu.vn.ua/media/images/porkhun/inf_8/l59_64/010.gif | https://disted.edu.vn.ua/media/images/porkhun/inf_8/l59_64/011.gif |

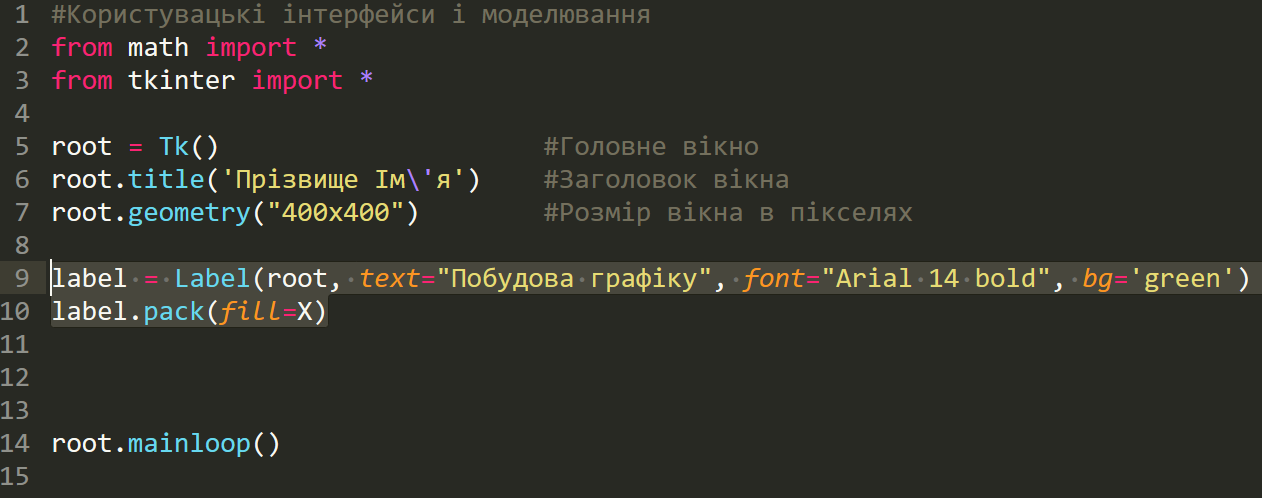
Ширину рамки можна також задати, для цього використовуємо borderwidth. По замовчуванню цей параметр дорівнює 0.

**Наприклад,**

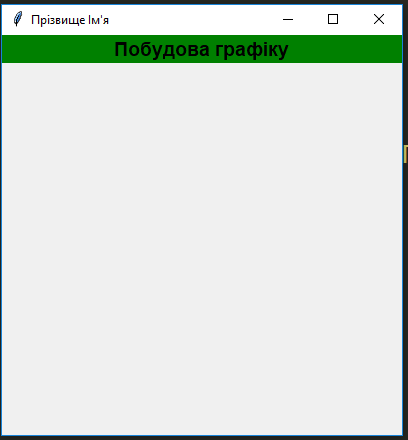
**root.config(width=400, height=400, background='steelblue', borderwidth=10, relief=RAISED)**

1. У вікні можна розміщувати різні об'єкти. Для розміщення тексту у вікні використаємо клас Label (з англ. ярлик).

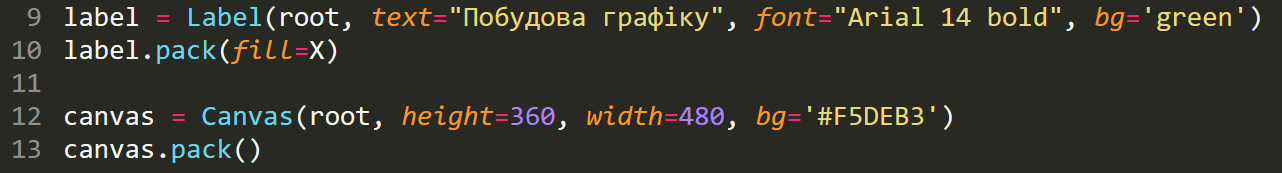
Добавимо віджет-мітку Label з написом «Побудова графіку» (рядки 9-10):



Якщо запустимо код, то отримаємо вікно такого вигляду



1. Добавте у вікні root елемент полотна:

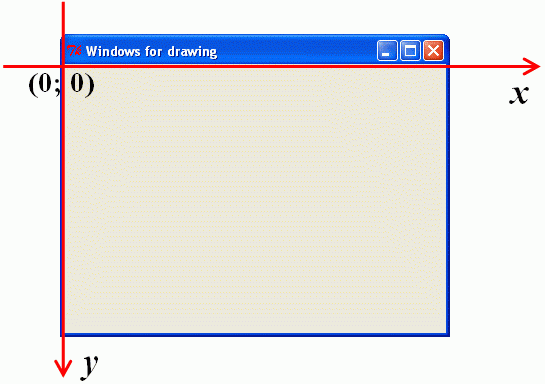


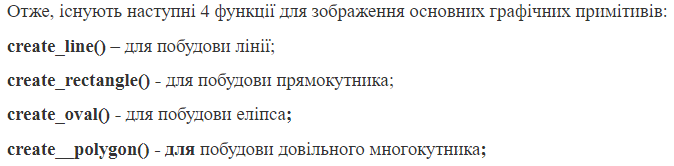
Перевірте чи змінився колір?

Змініть розмір вікна, задавши висоту в 500 пікселів:



На цьому  полотні є система координат, але початок координат знаходиться завжди у лівому верхньому куті, додатня частина осі ***х*** спрямована звичайним чином вправо, а от додатня частина вісі ***y*** спрямована **донизу**.  Тобто, у вікні видно лише першу координатну чверть - частина фігури, що не потрапляє у цю чверть просто не відображається.



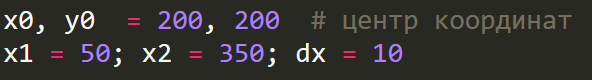


1. Запровадимо змінні:

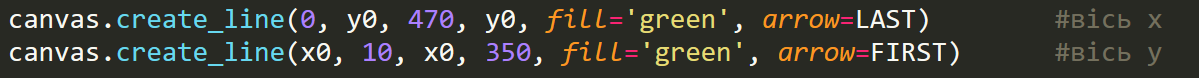
(x0, y0) – точка, де буде міститись початок координат.

[x1, x2] – відрізок, на якому будемо будувати графік функції, де

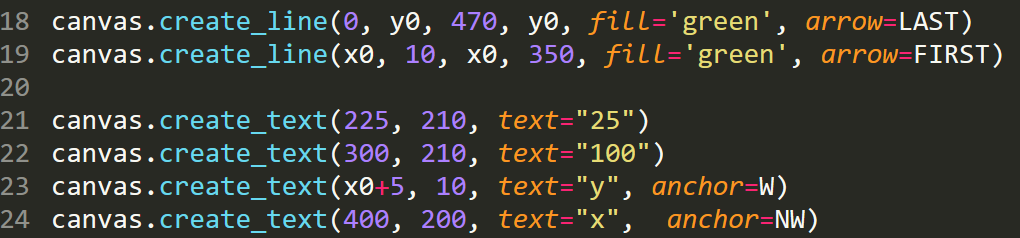
x1 – початок відрізка, x2 – кінець відрізка, dx – крок.



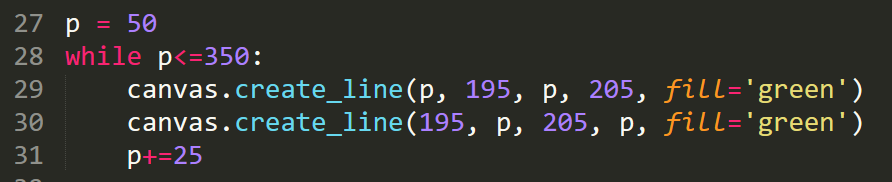
1. Далі намалюємо осі:



Позначимо осі та маштаб:

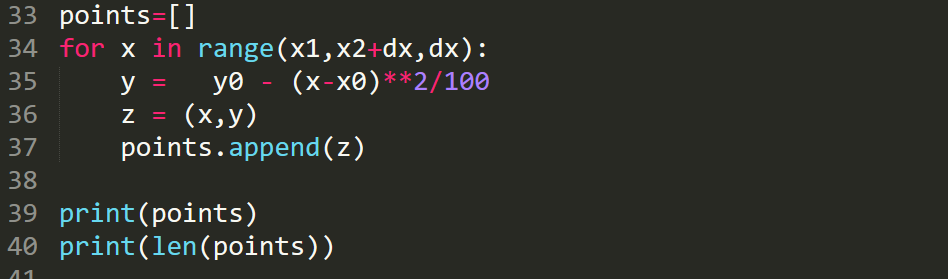


1. Організуємо цикл для нанесення поділок (невеликих ліній) на осях:



1. Створимо пустий список і організуємо цикл для заповнення цього списку точками z = (x,y).

Функція y задається як парабола, що проходить через центр (x0, y0) і точку (100,100):

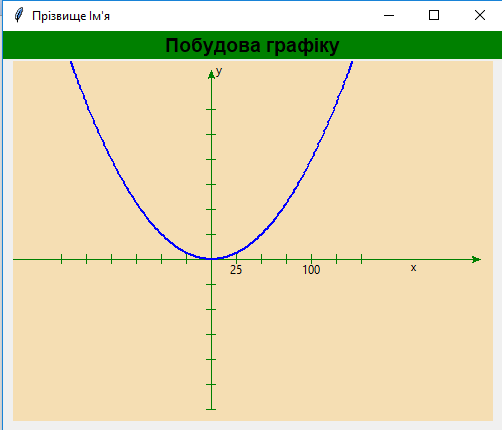


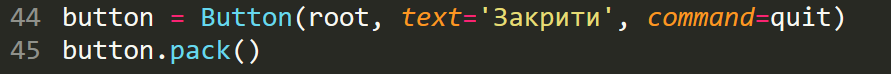
У рядку 39-40 виводяться сам список і його довжина.

1. Намалюємо графік параболи командою



Отримаємо



1. Добавте кпопку для виходу з програми
2. Збережіть програму з розширенням pyw. Закрийте робоче середовище. Перейдіть в папку, де міститься збережена програма, та запустіть її на виконання.

**Завдання для самостійного виконання (на основі попередньої програми):**

1. Поставте позначення осі «x» біля відповідної стрілки, аналогічно до «y».
2. Перенесіть початок координат на 100 пікселів вниз, і разом з тим координатні лінії та параболу.
3. Модифікуйте відповідний цикл для нанасення поділок (міток осі) з кроком 10 пікселів.
4. Позначіть текстом значення маштабу на осі «y» в 100 пікселів від початку координат.
5. Далі перенесіть параболу вправо на 30 пікселів, видозмінивши відповідно код.
6. Намалюйте червону пряму y = |x-30|, але так, щоб вона перетинала параболу.
7. Позначте початок координат маленьким оранжевим квадратиком, розміром 4x4 пікселі.

Посилання https://naurok.com.ua/urok-na-temu-zobrazhennya-osnovnih-grafichnih-ob-ektiv-u-python-53271.html