



алгоритмика

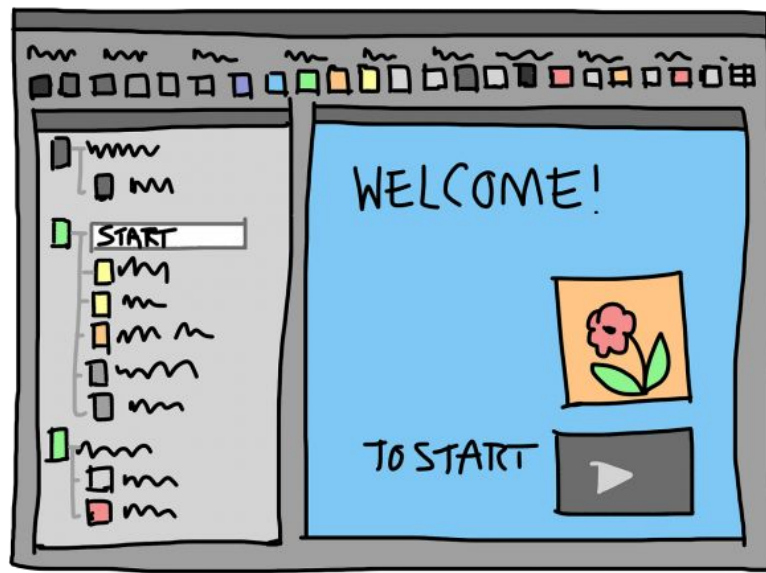
# PYTHON

Занятие 12

# Сегодня на уроке

---

- Отслеживание мыши
- Очистка Canvas
- Переключатели
- Создание Paint



# Вопросы

---

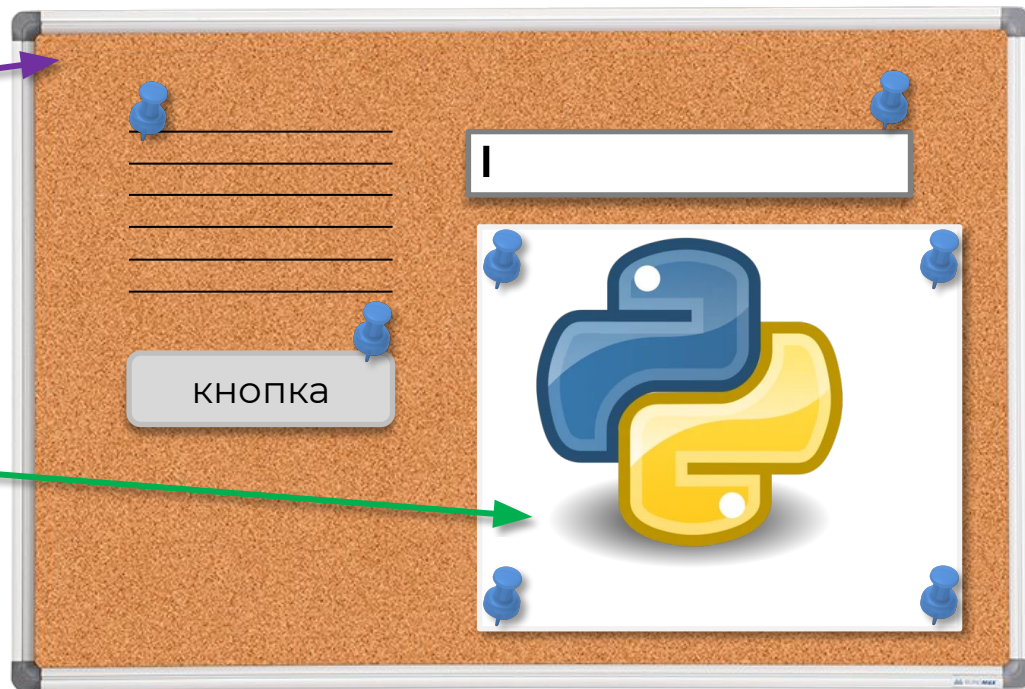
1. Как создаётся окно приложения?
2. Как задаётся размер окна?
3. Как разместить объект в определённом месте окна?
4. Как создать кнопку?
5. Что такое действие?
6. Что такое функция?
7. Что такое событие?
8. Как связать кнопку и функцию?

# Элементы интерфейса

**Окно** – основа  
интерфейса

**Холст** – основа  
для графических  
элементов

**canvas.pack()**



# Событие

---

Событие – это ответное действие, которое происходит после действия пользователя.

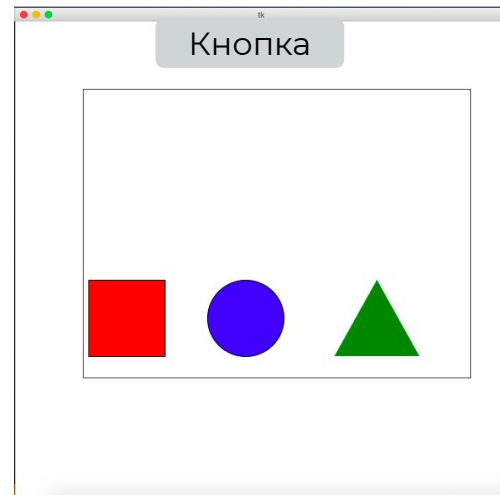
**Действие**

Пользователь нажал кнопку



**Событие**

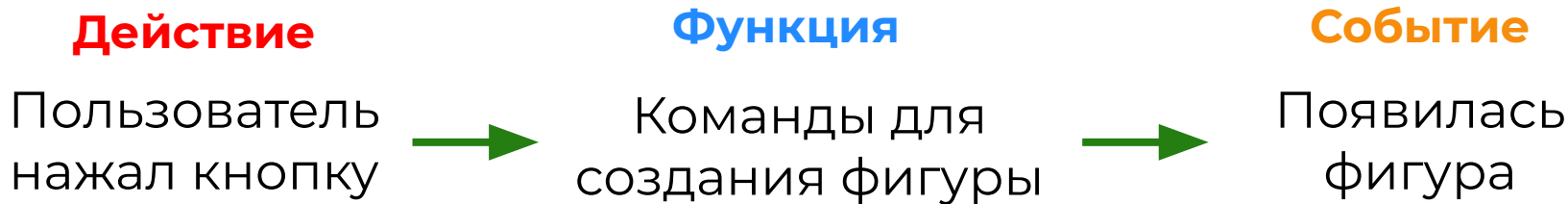
Появилась фигура



# Функция

---

Функция – это правило, которое описывает событие. Функция вызывается столько раз, сколько раз совершит действие пользователь.



# Функция в Python

---

**def** имя\_функции **(event):**

↔ тело функции

**Tab**

**def – definition** – правило. Сокращение, с которого начинается каждая функция.

имя\_функции – отражает назначение функции (печать, квадрат и т.п.).

**(event)** – обозначает ожидание события.



# Связка кнопки и функции

---

Метод bind – связывает виджет и функцию.

виджет.bind("<действие\_пользователя>", имя\_функции).  
(event) – обозначает ожидание события.

## Действие пользователя:

- <Button-1> - щелчок левой кнопкой мыши.
- <Button-2> - щелчок правой кнопкой мыши.
- <Button-3> - щелчок средней кнопкой мыши.
- <Double-Button-1> - двойной клик левой кнопкой мыши
- <Motion> - движение мыши.



# Задание размера окна

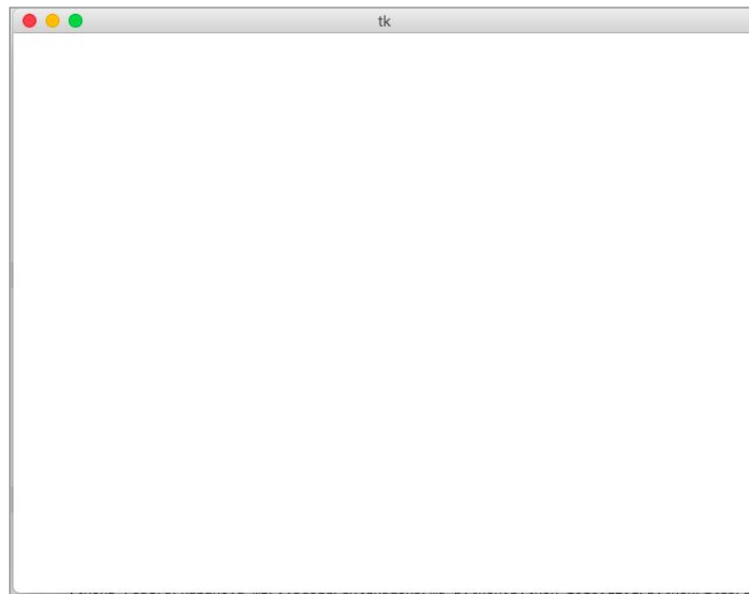
---

**geometry** - метод класса Tk(), который позволяет задать размер окна. **Размер окна должен быть больше размера холста.**

```
.geometry('ширинаxвысота')
```

```
root=Tk()
```

```
root.geometry('800x800')
```

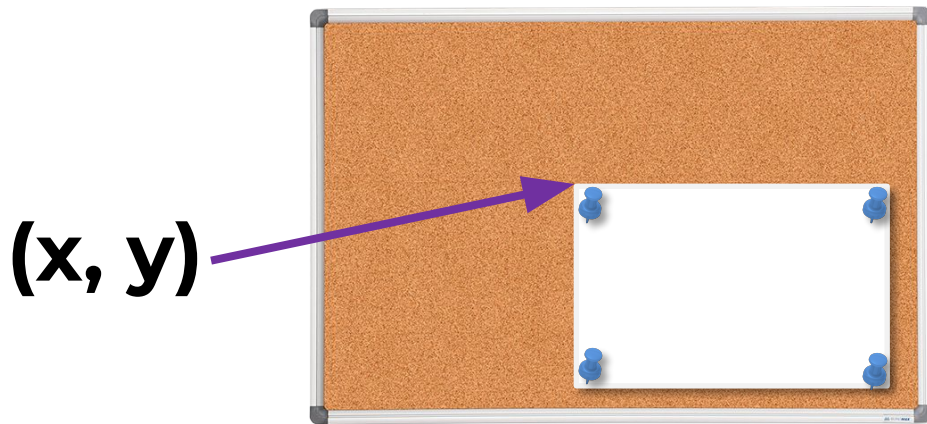


# Перемещение холста

---

**place** - метод, позволяющий закрепить объект в указанных координатах.

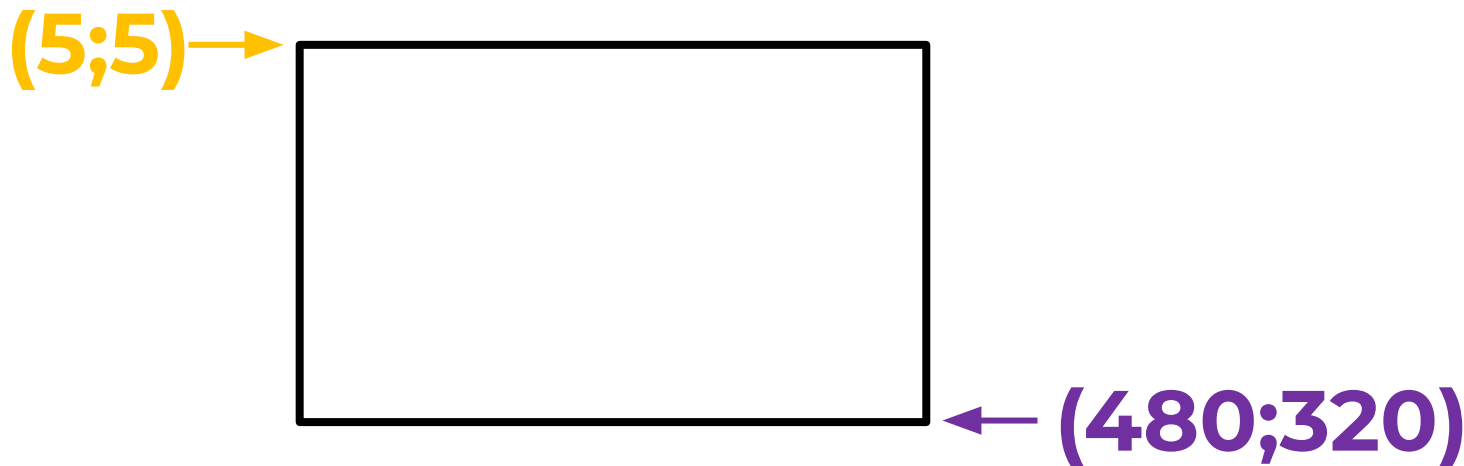
объект.place (x=100, y=100)



# Задание

---

Составьте программу для создания окна размером 600x600 с холстом размером 480x320, с началом в точке (100;200). На холсте расположите прямоугольник (см. рисунок).

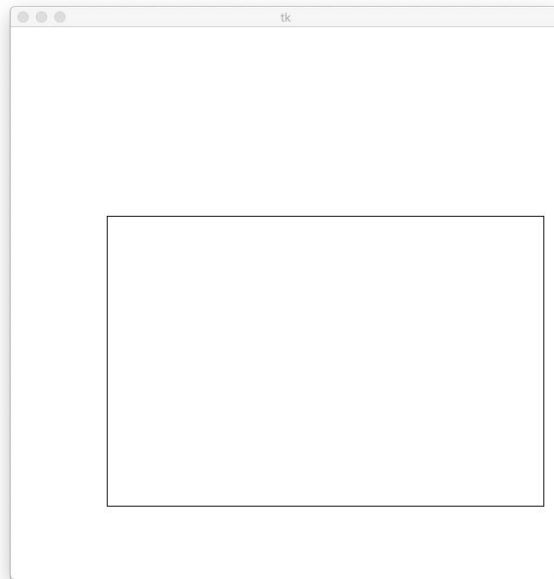


# Задание. Решение

---

Составьте программу для создания окна размером 600x600 с холстом размером 480x320, с началом в точке (100;200). На холсте расположите прямоугольник (см. рисунок).

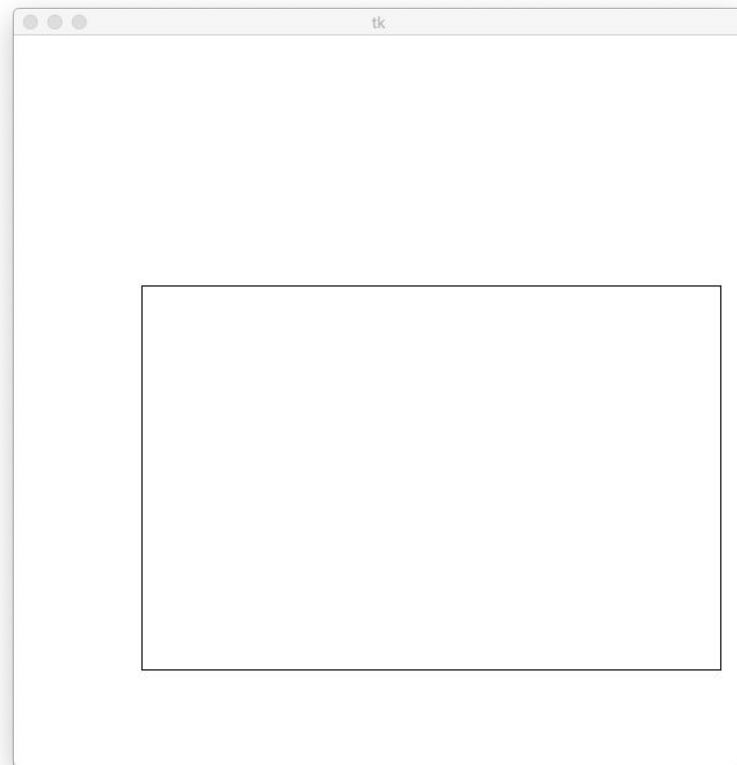
```
from tkinter import*  
root = Tk()  
root.geometry('600x600')  
  
canvas = Canvas(root,width=480,height=320)  
canvas.pack()  
canvas.place(x=100,y=200)  
canvas.create_rectangle(5,5,480,320)  
  
root.mainloop()
```



# Вопрос

---

Что обозначает  
прямоугольник, созданный  
на холсте?

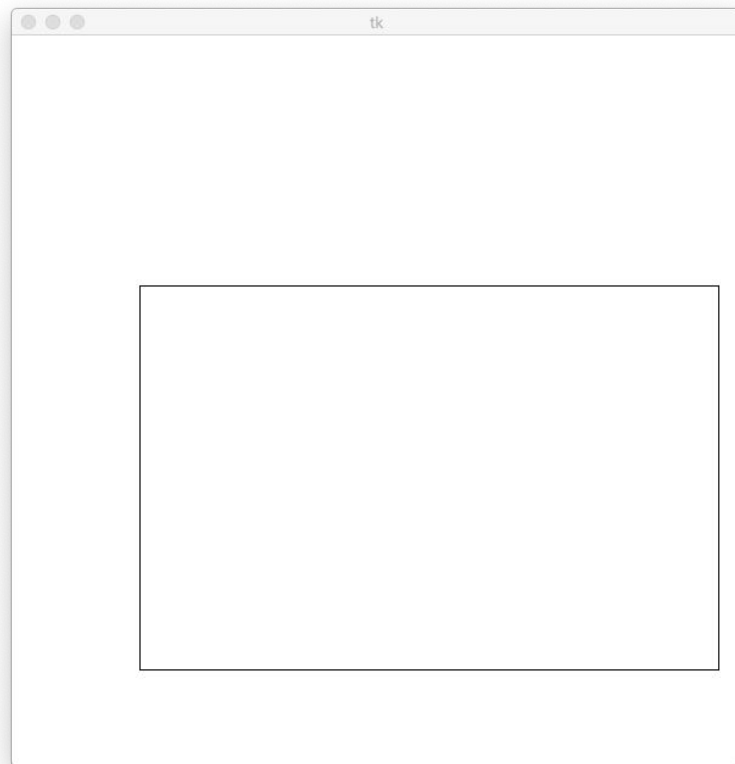


# Вопрос-ответ

---

Что обозначает  
прямоугольник, созданный  
на холсте?

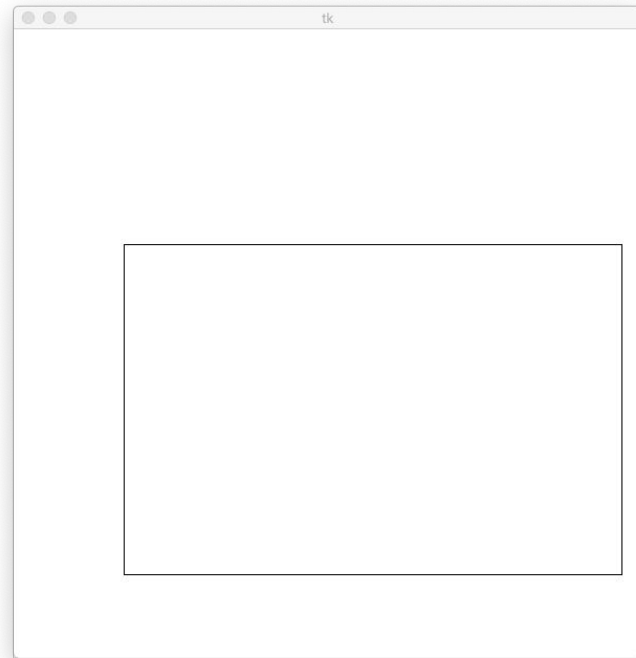
**Границы холста**



# Вопрос

---

Как и где мы указываем место расположения фигуры на холсте?

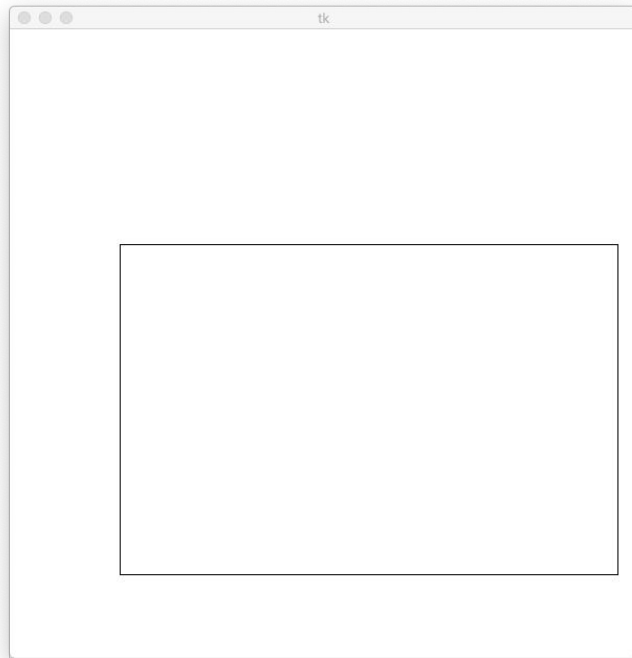


# Вопрос-ответ

---

Как и где мы указываем место расположения фигуры на холсте?

**В программе, в  
методе создания  
фигуры**

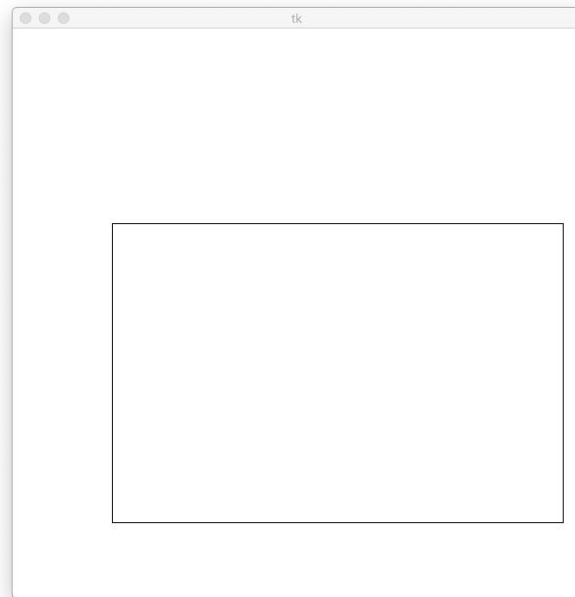




# Вопрос

---

Может ли сейчас пользователь изменить или указать место появления фигуры в момент работы приложения?

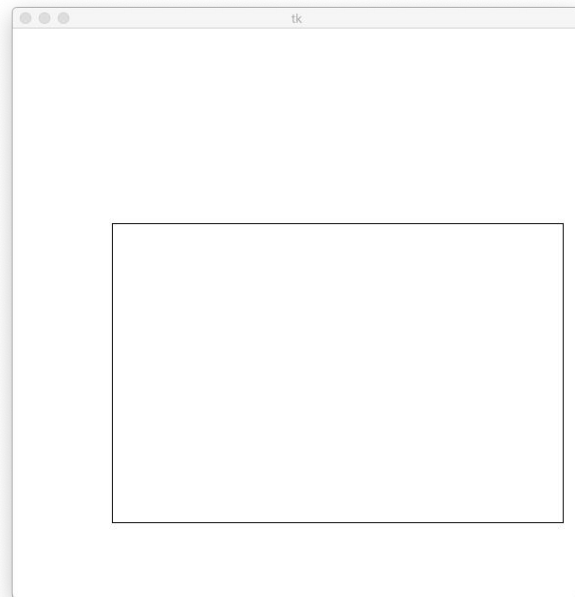


# Вопрос-ответ

---

Может ли сейчас пользователь изменить или указать место появления фигуры в момент работы приложения?

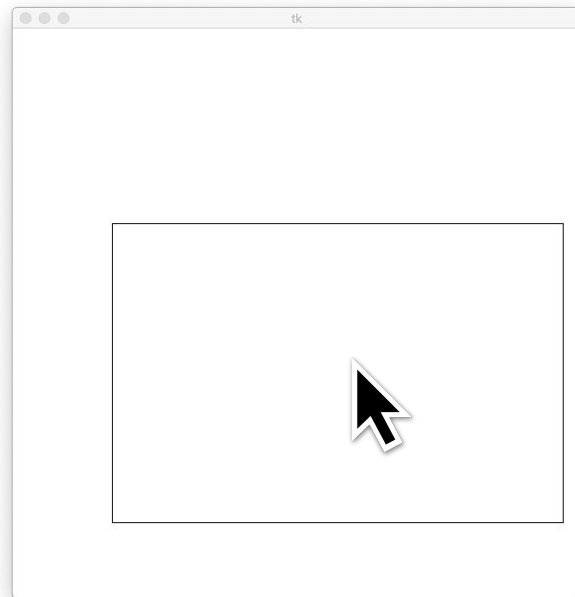
**Сейчас нет. Нужно  
останавливать программу  
и изменять координаты.**



# Вопрос

---

С помощью чего пользователь сможет указывать, где нарисовать фигуру?

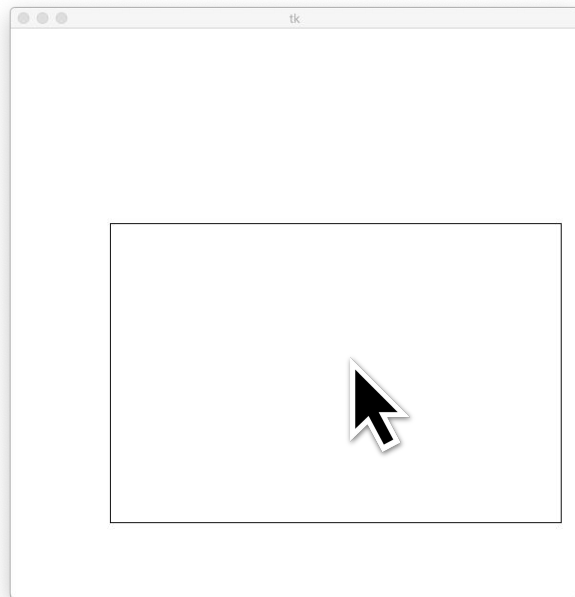
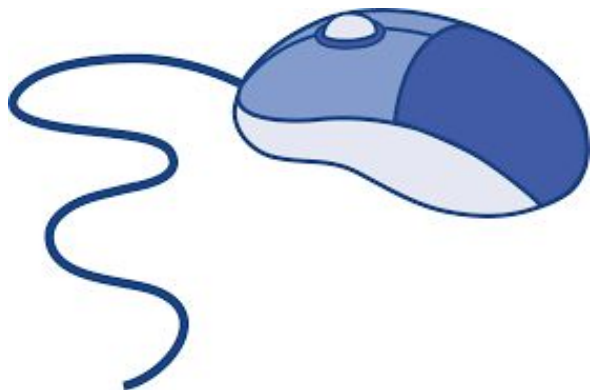


# Вопрос-ответ

---

С помощью чего пользователь сможет указывать, где нарисовать фигуру?

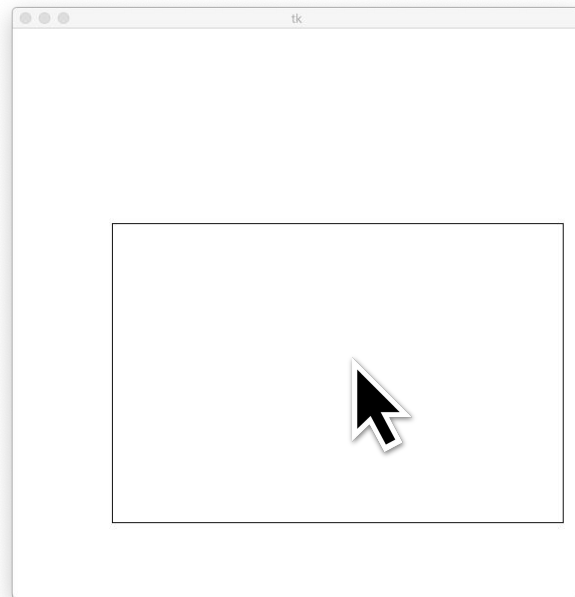
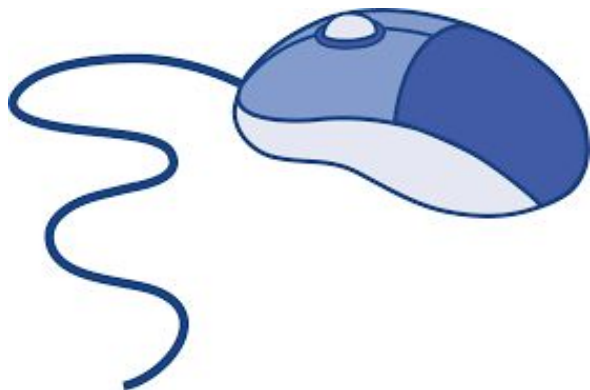
**С помощью курсора мыши**



# Вопрос

---

Какое действие пользователю нужно совершить, чтобы произошло событие – появилась фигура?

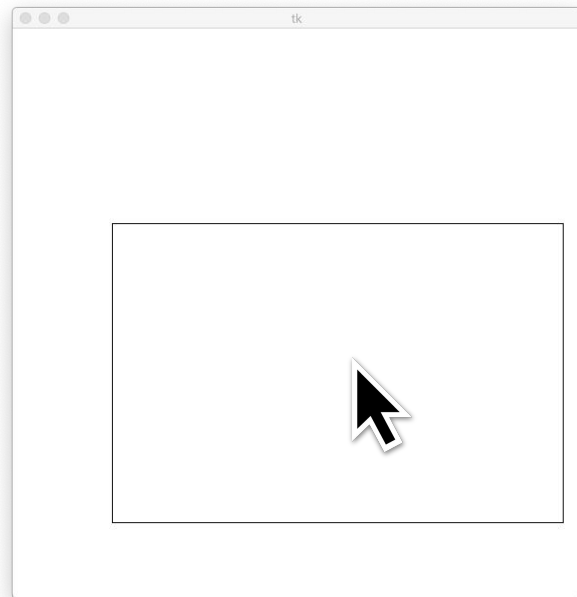
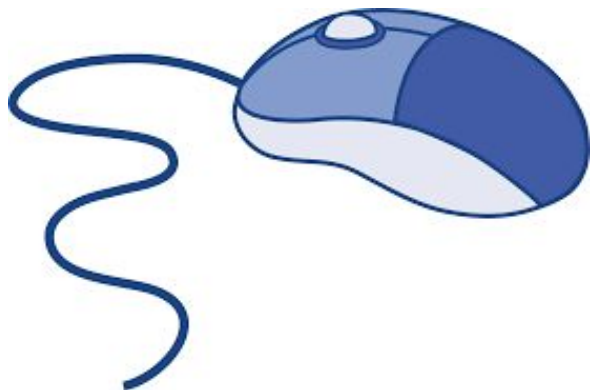


# Вопрос-ответ

---

Какое действие пользователю нужно совершить, чтобы произошло событие – появилась фигура?

**Нажать на кнопку мыши**



# Объект - действие

---

Объект `event` позволяет отследить координаты, в котором произошло действие.

**event.x** - Отслеживает координату **x**

**event.y** - Отслеживает координату **y**

`x = event.x` – **переменная x** указывает на значение координаты **x**

`y = event.y` – **переменная y** указывает на значение координаты **y**



# Задание

---

Добавьте в программу функцию `click_left`, в тело функции добавьте следующие команды:

```
x = event.x
```

```
y = event.y
```

```
print("x:", x)
```

```
print("y:", y)
```

1. Какое событие будет происходить после действия пользователя?
2. Где мы увидим результаты команды `print`?



# Решение

---

Добавьте в программу функцию `click_left`, в тело функции добавьте следующие команды:

```
from tkinter import*
```

```
root = Tk()
```

```
root.geometry('600x600')
```

```
def click_left(event):
```

```
    x = event.x
```

```
    y = event.y
```

```
    print("x:", x)
```

```
    print("y:", y)
```

```
canvas=Canvas(root,width=480,height=320)
```

```
canvas.pack()
```

```
canvas.place(x=100,y=200)
```

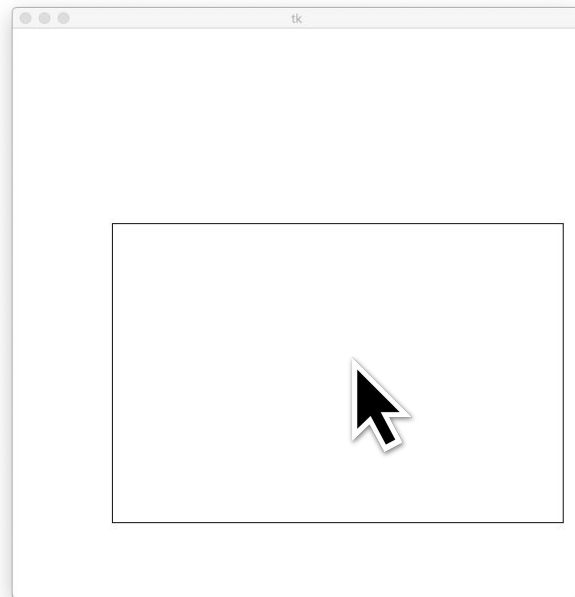
```
canvas.create_rectangle(5,5,480,320)
```

```
root.mainloop()
```

# Вопрос

---

В какой части окна пользователь должен нажать левую кнопку мыши, чтобы появлялась фигура?

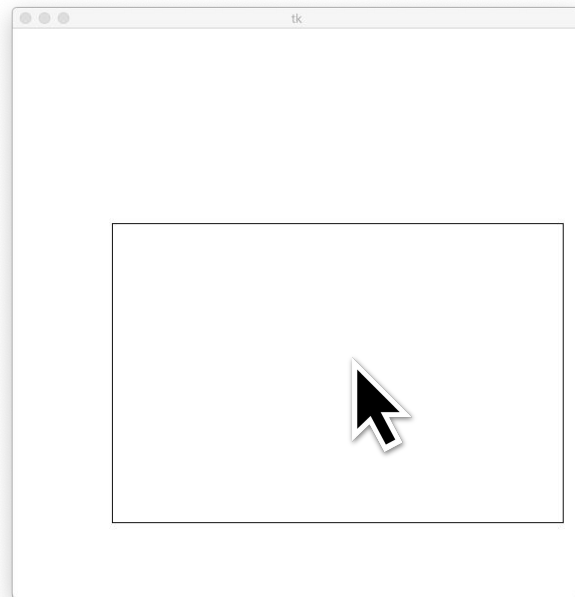


# Вопрос-ответ

---

В какой части окна пользователь должен нажать левую кнопку мыши, чтобы появлялась фигура?

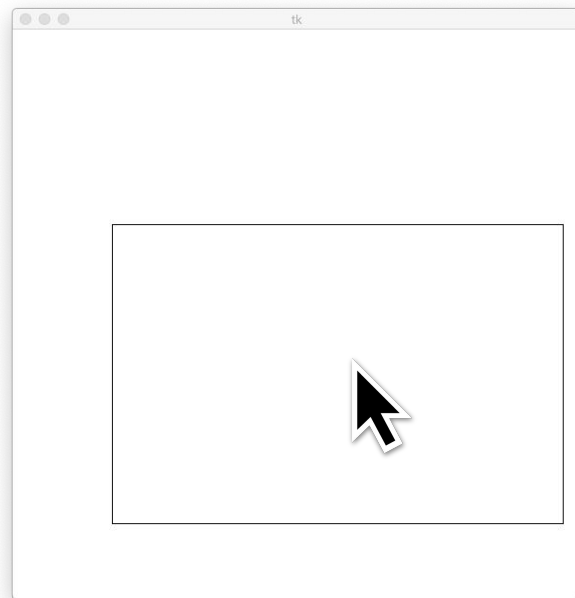
**По холсту**



# Вопрос

---

С каким объектом в программе мы должны связать функцию `click_left`?



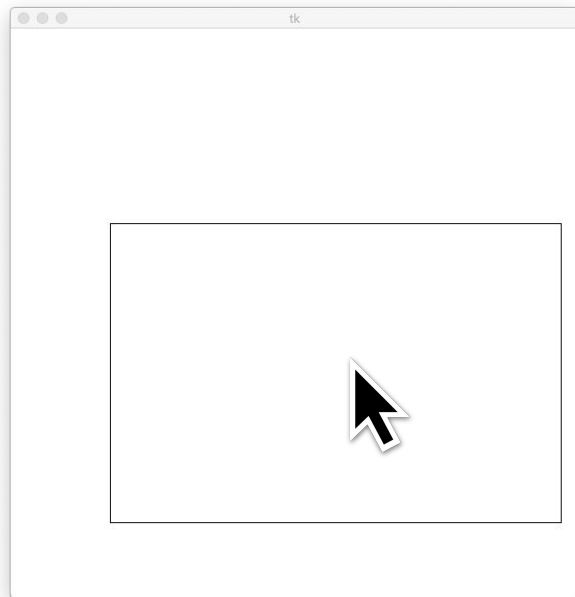
# Вопрос-ответ

---

С каким объектом в программе мы должны связать функцию `click_left`?

**canvas**

```
canvas.bind("<Button-1>", click_left)
```



# Задание

---

Добавьте в программу привязку функции click\_left к canvas

```
from tkinter import*
root = Tk()
root.geometry('600x600')
def click_left(event):
    x = event.x
    y = event.y
    print("x:", x)
    print("y:", y)
canvas=Canvas(root,width=480,height=320)
canvas.pack()
canvas.place(x=100,y=200)
canvas.create_rectangle(5,5,480,320)
root.mainloop()
```

← canvas.bind("<Button-1>",click\_left)

# Задание. Решение

Добавьте в программу привязку функции click\_left к canvas

```
from tkinter import*
root = Tk()
root.geometry('600x600')
def click_left(event):
    x = event.x
    y = event.y
    print("x:", x)
    print("y:", y)
canvas=Canvas(root,width=480,height=320)
canvas.pack()
canvas.place(x=100,y=200)
canvas.create_rectangle(5,5,480,320)
canvas.bind("<Button-1>",click_left)
root.mainloop()
```



# Вопрос

---

Как работает метод `create_oval`?



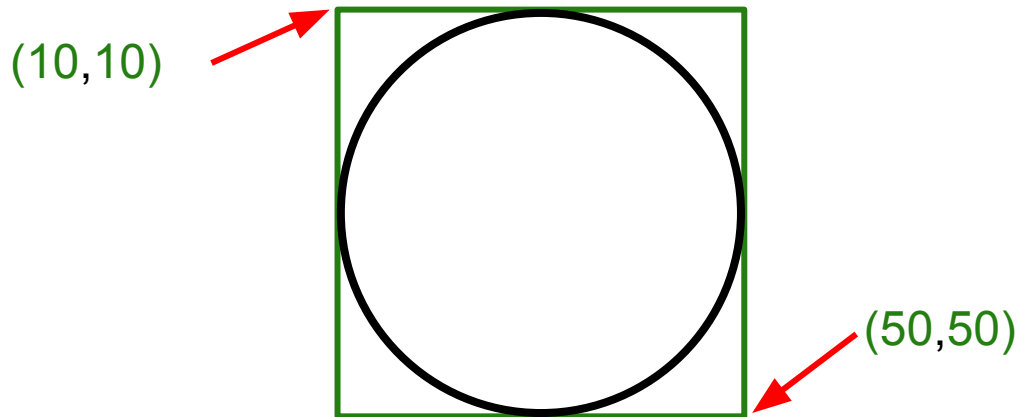


# Вопрос-ответ

Как работает метод `create_oval`?

**Создаёт круг в прямоугольнике,  
координаты которого указаны в методе**

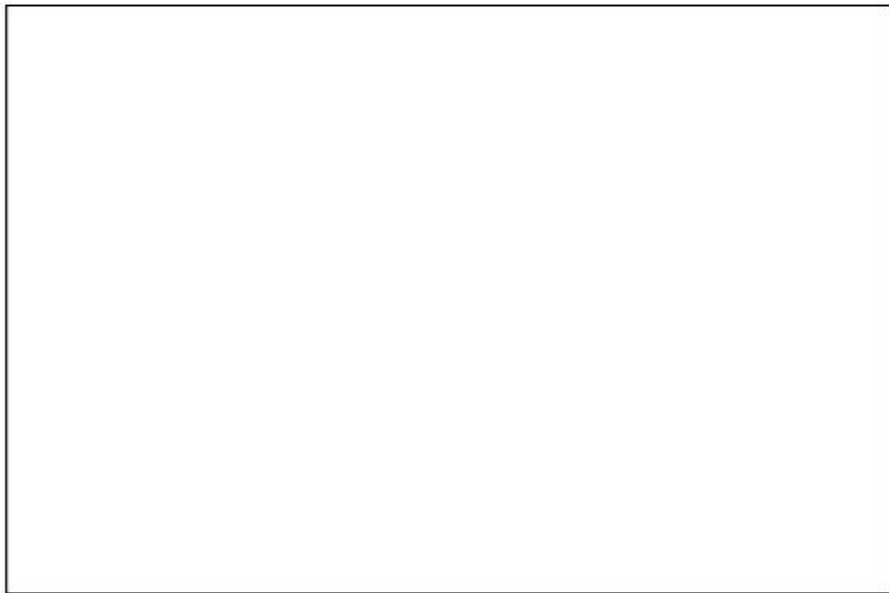
`canvas.create_oval(10,10,50,50)`



# Вопрос

---

Какому месту круга соответствует положение курсора мыши?

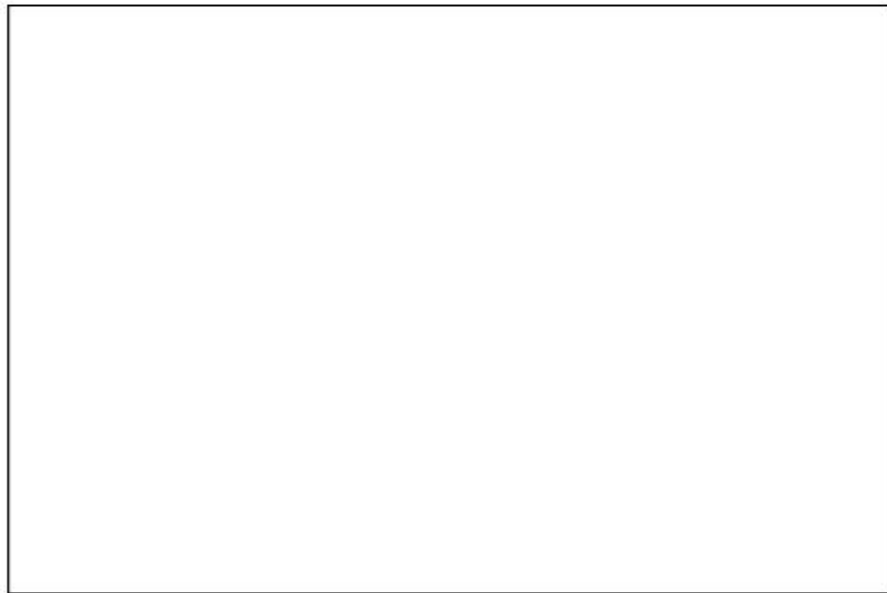


# Вопрос-ответ

---

Какому месту круга соответствует положение курсора мыши?

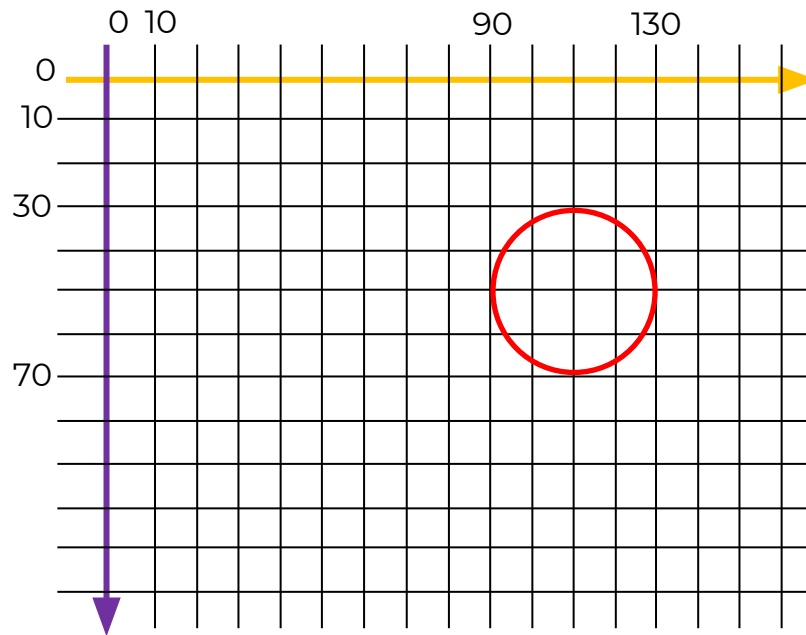
**Центру**



# Задание на листе

По рисунку добавьте значение координат в метод `create_oval`.

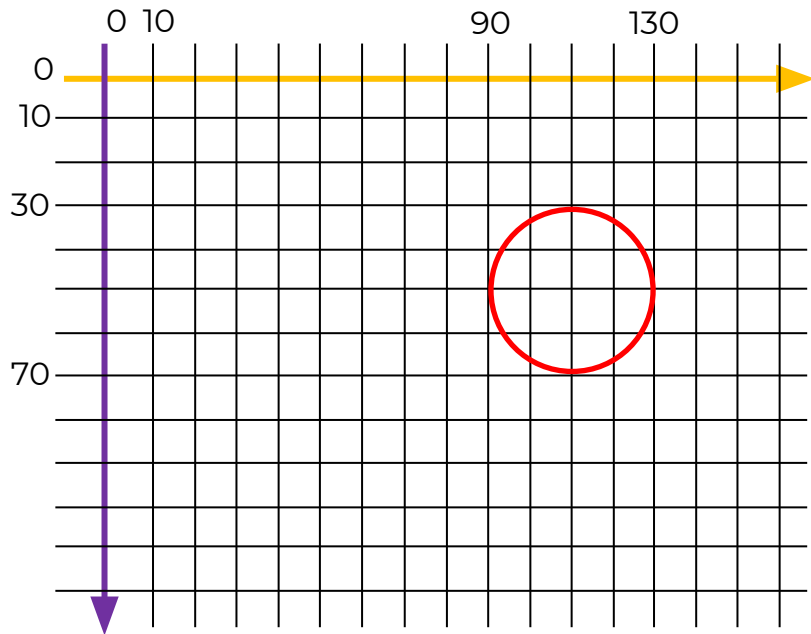
`canvas.create_oval(____,____,____,____)`



# Задание на листе. Решение

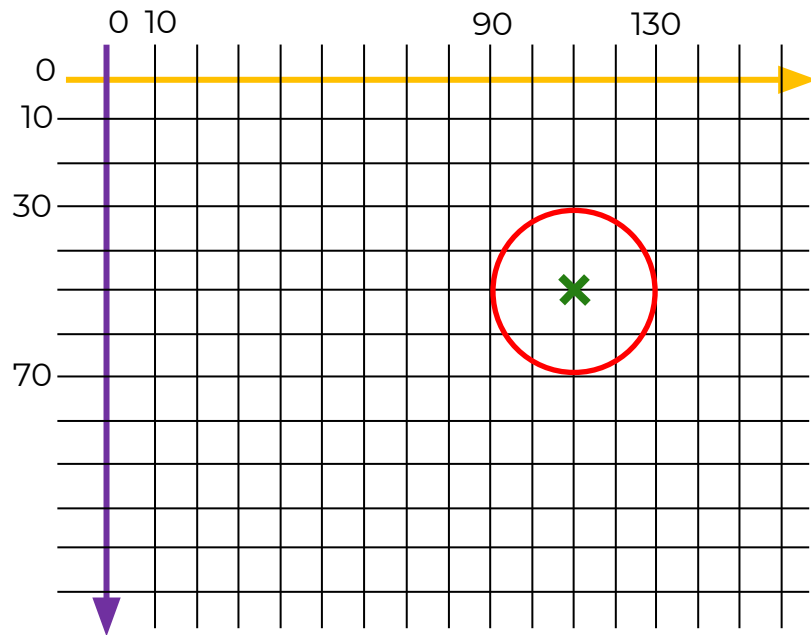
По рисунку добавьте значение координат в метод `create_oval`

```
canvas.create_oval(90,30,130,70)
```



# Задание на листе

Запишите координаты центра круга.

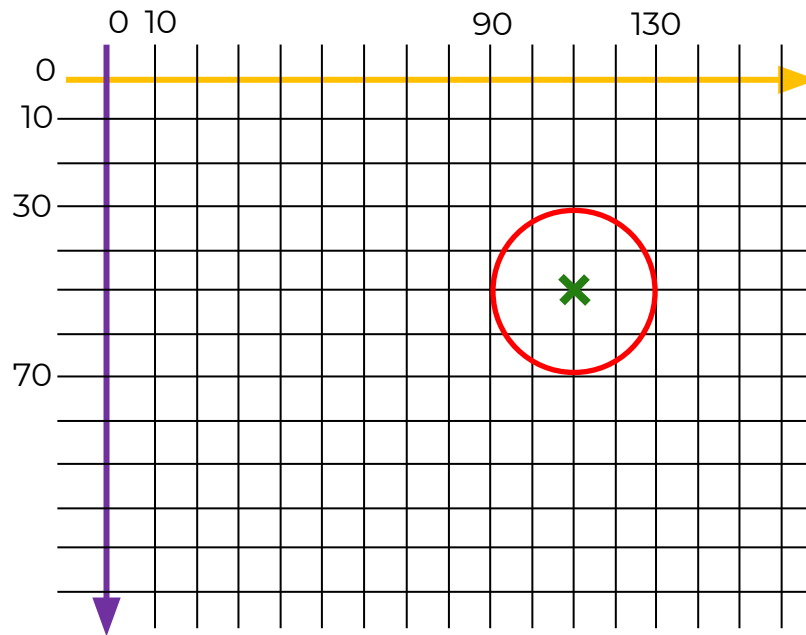


# Задание на листе. Решение

Запишите координаты  
центра круга.

**x=110**

**y=50**

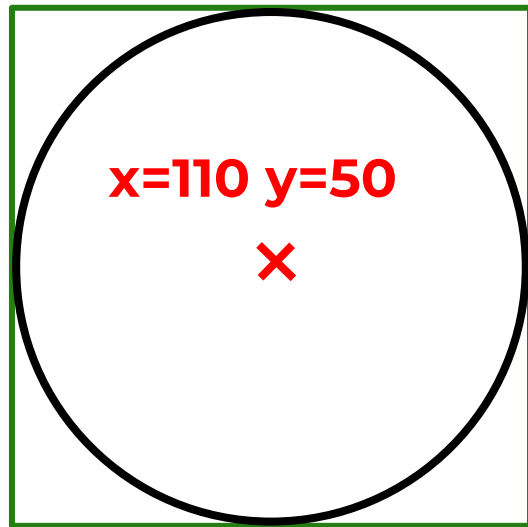


# Задание

---

Как, зная координаты центра круга, получить координаты квадрата, в котором его нужно нарисовать?

$x_1=90$   $y_1=30$



$x_2=130$   $y_2=70$

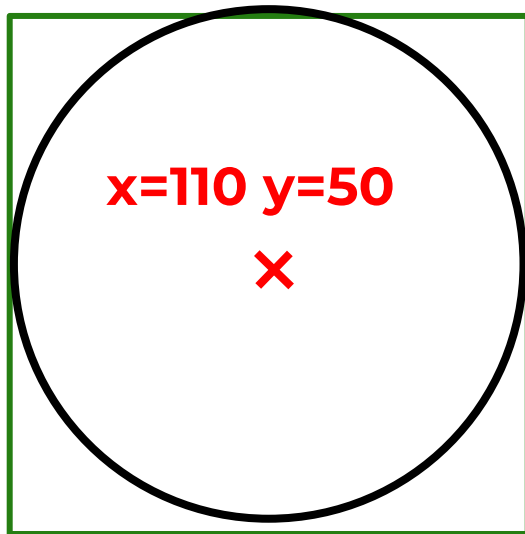




# Задание. Решение

Как, зная координаты центра круга, получить координаты квадрата, в котором его нужно нарисовать?

$x1=90$   $y1=30$



$$110 - 20 = 90$$

$$50 - 20 = 30$$

$$110 + 20 = 130$$

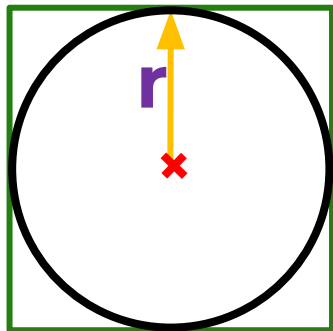
$$50 + 20 = 70$$

$x2=130$   $y2=70$



# Радиус

**20** – радиус круга, который создаётся внутри квадрата. **Радиус (r)** – расстояние от центра до любой точки круга.



$$110 - 20 = 90$$

$$50 - 20 = 30$$

$$110 + 20 = 130$$

$$50 + 20 = 70$$

$$x - r = x1$$

$$y - r = y1$$

$$x + r = x2$$

$$y + r = y2$$

# Задание

---

Какие координаты должны быть в методе `create_oval`, если пользователь создаёт круг с радиусом 30 и нажимает мышкой в координатах (40;50).

$$x - r = x1$$

$$y - r = y1$$

$$x + r = x2$$

$$y + r = y2$$

# Задание. Решение

---

Какие координаты должны быть в методе `create_oval`, если пользователь создаёт круг с радиусом 30 и нажимает мышкой в координатах (40;50).

`canvas.create_oval(10,20,70,80)`

$$x - r = x1$$

$$y - r = y2$$

$$x + r = x2$$

$$y + r = y2$$

---

## Задание

Измените функцию `click_left`. Добавьте задание радиуса и создание круга, там, где нажал пользователь.

```
def click_left(event):
```

```
    x = event.x
```

```
    y = event.y
```

```
    print("x:", x)
```

```
    print("y:", y)
```

```
        r=20
```

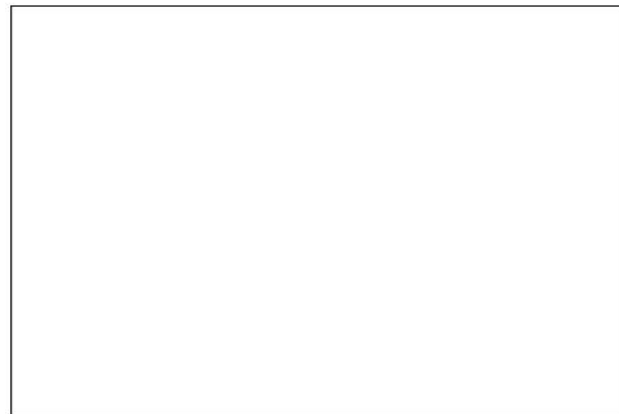
```
        canvas.create_oval(x-r,y-r,x+r,y+r)
```

# Задание. Решение

---

Измените функцию `click_left`. Добавьте задание радиуса и создание круга, там, где нажал пользователь.

```
def click_left(event):  
    x = event.x  
    y = event.y  
    print("x:", x)  
    print("y:", y)  
    r=20  
    canvas.create_oval(x-r,y-r,x+r,y+r)
```



# Переключатель

---

Виджет Radiobutton (переключатель) – позволяет пользователю выбрать 1 из нескольких вариантов.



Вариант 1

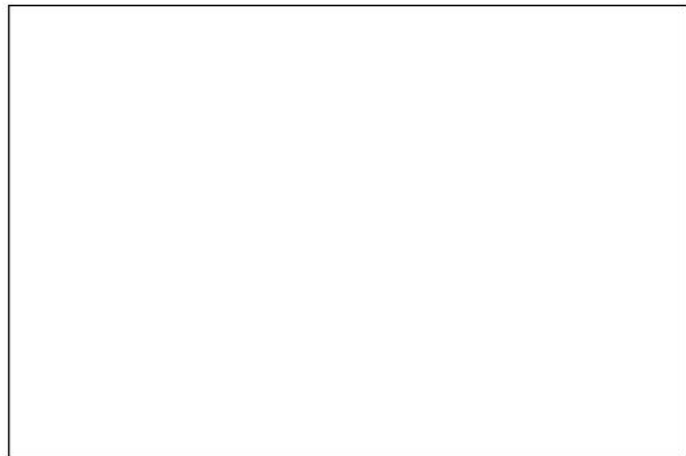
Вариант 2

Вариант 3

# Вопрос

---

Для чего мы можем использовать переключатель в нашей программе?



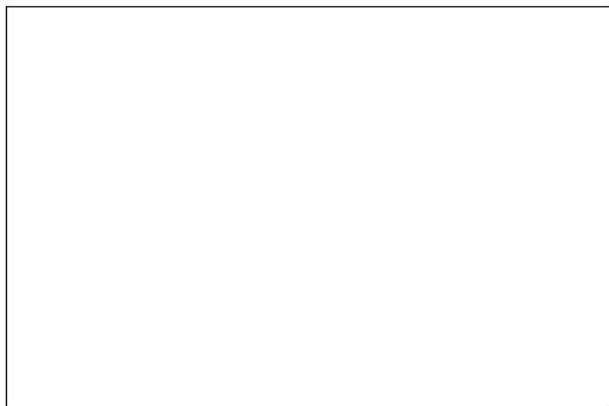
- ☒ Вариант 1
- ☐ Вариант 2
- ☐ Вариант 3



# Вопрос-ответ

---

Для чего мы можем использовать переключатель в нашей программе?



**Для  
изменения  
радиуса**



# Создание переключателя

---

`var=IntVar()` – переменная `var` указывает, что значения в переключателе будут целыми числами (`IntVar`).

`var.set(0)` – по умолчанию будет выбран первый переключатель.

```
r_1=Radiobutton(text="1",variable=var,value=0)
```

```
text="текст_рядом_с_кнопкой"
```

```
variable=имя_переменной
```

```
value=номер_кнопки
```

```
r_1.pack()
```

# Задание

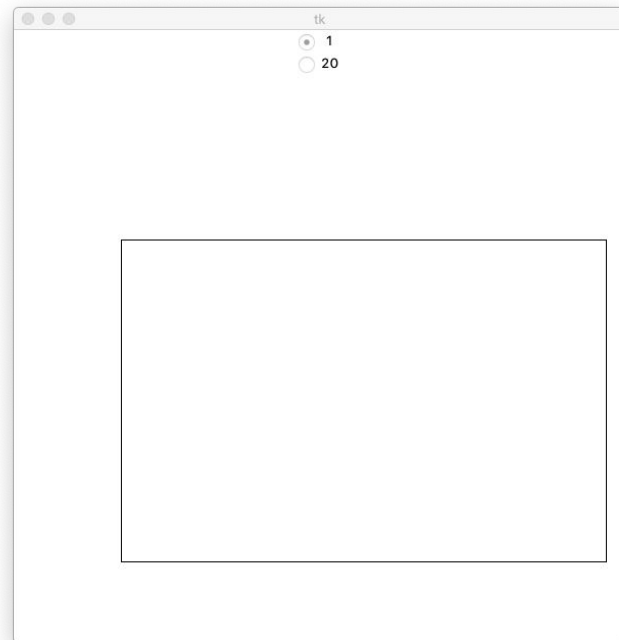
---

Добавьте в программу создание переключателя из 2 кнопок.

...

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()
```

```
canvas=Canvas(root,width=480,height=320)  
canvas.pack()  
canvas.place(x=100,y=200)  
canvas.create_rectangle(5,5,480,320)  
canvas.bind("<Button-1>",click_left)  
root.mainloop()
```



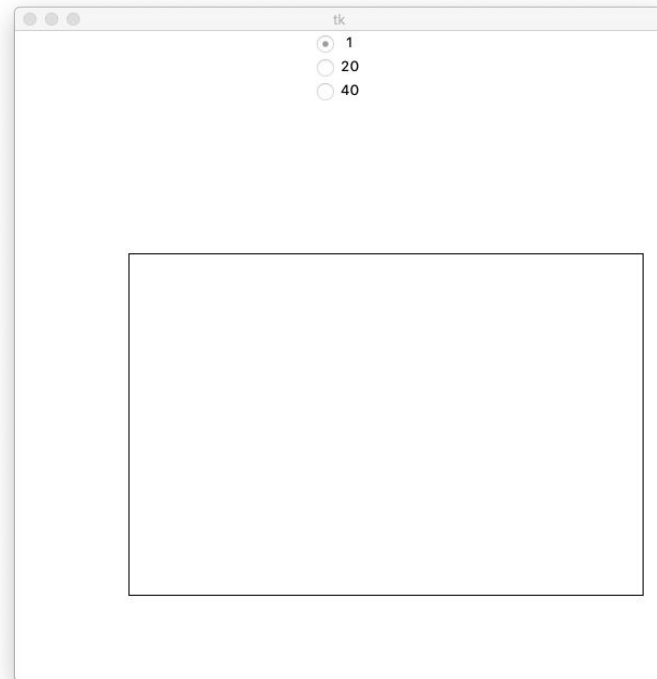
# Задание

Добавьте в переключатель ещё 1 кнопку для варианта 40.

...

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()
```

```
canvas=Canvas(root,width=480,height=320)  
canvas.pack()  
canvas.place(x=100,y=200)  
canvas.create_rectangle(5,5,480,320)  
canvas.bind("<Button-1>",click_left)  
root.mainloop()
```

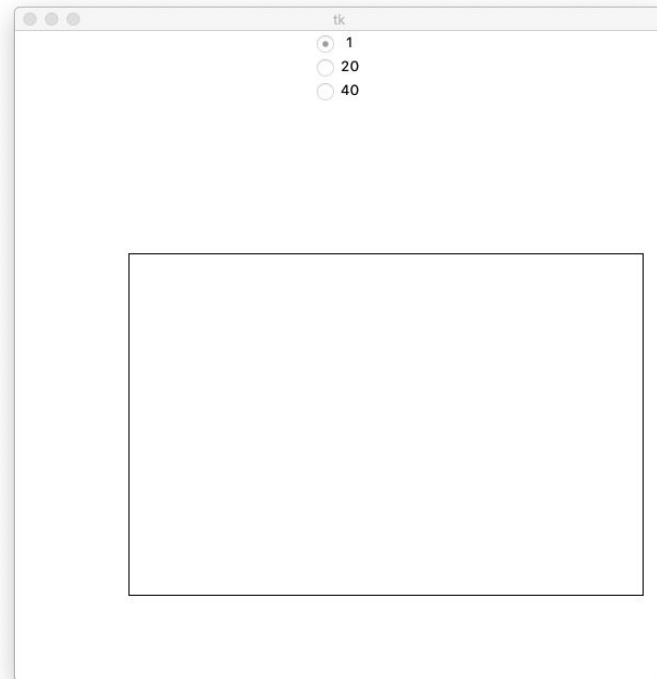


# Задание. Решение

Добавьте в переключатель ещё 1 кнопку для варианта 40

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()  
r_3=Radiobutton(text="40",variable=var,value=2)  
r_3.pack()
```

```
canvas=Canvas(root,width=480,height=320)  
canvas.pack()  
canvas.place(x=100,y=200)  
canvas.create_rectangle(5,5,480,320)  
canvas.bind("<Button-1>",click_left)  
root.mainloop()
```



# Вопрос

---

На какое значение будет указывать var, если будет выбрана 2 кнопка?

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()  
r_3=Radiobutton(text="40",variable=var,value=2)  
r_3.pack()
```

# Вопрос-ответ

---

На какое значение будет указывать var, если будет выбрана 2 кнопка?

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()  
r_3=Radiobutton(text="40",variable=var,value=2)  
r_3.pack()
```

**var=1**

# Вопрос

---

Как программа будет понимать, какая кнопка выбрана?





# Вопрос-ответ

---

Как программа будет понимать, какая кнопка выбрана?

Кнопка	Значение var
1	0
20	1
40	2

**По значению  
переменной var**

**var.get()** – получение значения  
переменной



# Вопрос

---

Какой оператор мы использовали, чтобы выполнять разные действия в зависимости от выполнения или невыполнения условия?

# Вопрос

---

Какой оператор мы использовали, чтобы выполнять разные действия в зависимости от выполнения или невыполнения условия?

if условие\_1 :

набор команд\_1

elif условие\_2 :

набор команд\_2

else:

набор команд\_3

**Условный  
оператор**



# Вопрос

---

В какой части программы будет проходить проверка значения переменной var?

```
from tkinter import*
root = Tk()
root.geometry('600x600')
def click_left(event):
    x = event.x
    y = event.y
    print("x:", x)
    print("y:", y)
    r=20
    canvas.create_oval(x-r,y-r,x+r,y+r)
```

```
var=IntVar()
var.set(0)
r_1=Radiobutton(text=" 1 ",variable=var,value=0)
r_1.pack()
r_2=Radiobutton(text="20",variable=var,value=1)
r_2.pack()
r_3=Radiobutton(text="40",variable=var,value=2)
r_3.pack()

canvas=Canvas(root,width=480,height=320)
canvas.pack()
canvas.place(x=100,y=200)
canvas.create_rectangle(5,5,480,320)
canvas.bind("<Button-1>",click_left)
root.mainloop()
```

# Вопрос

В какой части программы будет проходить проверка значения переменной var?

```
from tkinter import*  
root = Tk()  
root.geometry('600x600')  
def click_left(event):  
    x = event.x  
    y = event.y  
    print("x:", x)  
    print("y:", y)  
    r=20
```

canvas.create\_oval(x-r,y-r,x+r,y+r)

**Перед  
созданием круга**

```
var=IntVar()  
var.set(0)  
r_1=Radiobutton(text=" 1 ",variable=var,value=0)  
r_1.pack()  
r_2=Radiobutton(text="20",variable=var,value=1)  
r_2.pack()  
r_3=Radiobutton(text="40",variable=var,value=2)  
r_3.pack()
```

```
canvas=Canvas(root,width=480,height=320)  
canvas.pack()  
canvas.place(x=100,y=200)  
canvas.create_rectangle(5,5,480,320)  
canvas.bind("<Button-1>",click_left)  
root.mainloop()
```



# Задание

---

Добавьте в функцию проверку значения `var` перед созданием круга. Как изменится работа программы?

# Задание. Решение

---

Добавьте в функцию проверку значения var перед созданием круга. Как изменится работа программы?

```
def click_left(event):  
    x = event.x  
    y = event.y  
    print("x:", x)  
    print("y:", y)  
  
    if var.get()==0:  
        r=1  
    elif var.get()==1:  
        r=20  
    else:  
        r=40  
    canvas.create_oval(x-r,y-r,x+r,y+r)
```

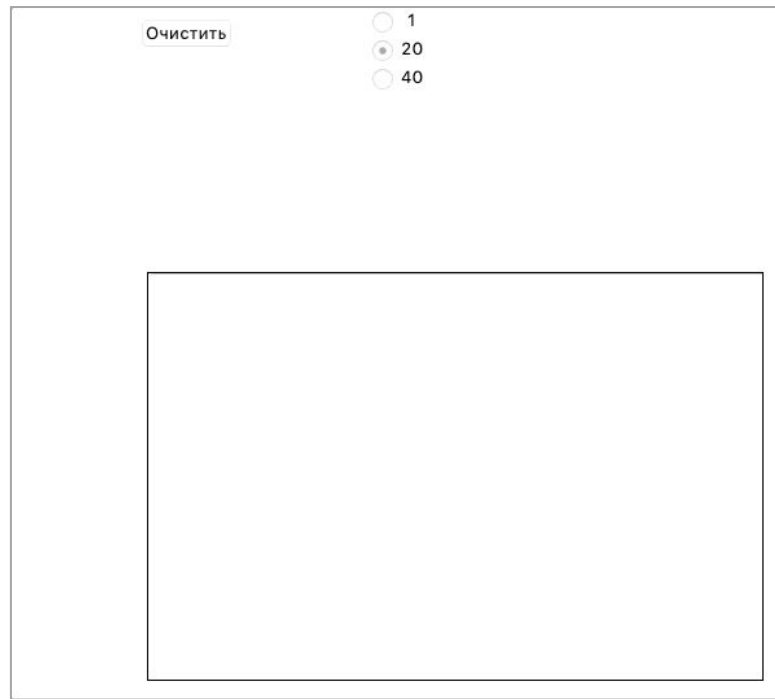


# Задание

---

Добавьте в приложение кнопку “Очистить” в точку  $x=100$ ,  $y=10$  и создайте для неё функцию `clean`. Тело функции приведено ниже.

```
canvas.delete('all')  
canvas.create_rectangle(5,5,480,320)
```



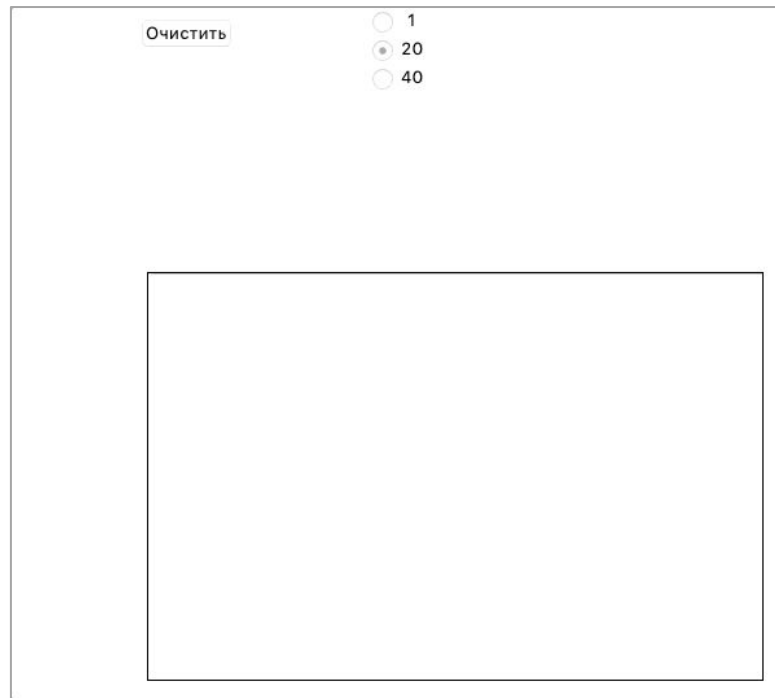


# Задание. Решение

---

Добавьте в приложение кнопку Очистить и создайте для неё функцию clean. Тело функции приведено ниже.

```
def clean(event):  
    canvas.delete('all')  
    canvas.create_rectangle(5,5,480,320)  
  
    btn_1 = Button(root,text="Очистить")  
    btn_1.pack()  
    btn_1.place(x=100,y=10)  
    btn_1.bind("<Button-1>",clean)
```



# Вопросы

---

1. Как получить значение положения мыши?
2. Что такое радиус?
3. Как работает переключатель?
4. Как создать переключатель?
5. В какой части программы происходит проверка значения переключателя?
6. Как очистить canvas?

# Задание на дом. Уровень 1

---

Добавьте еще один вариант размера кисти: 10.

- ☒ 5
- ☐ 10
- ☐ 20
- ☐ 40

## Задание на дом. Уровень 2

---

Добавьте еще один вариант размера кисти: 10.  
Разместите кнопки переключателя  
горизонтально.

☒ 5   ☐ 10   ☐ 20   ☐ 40