



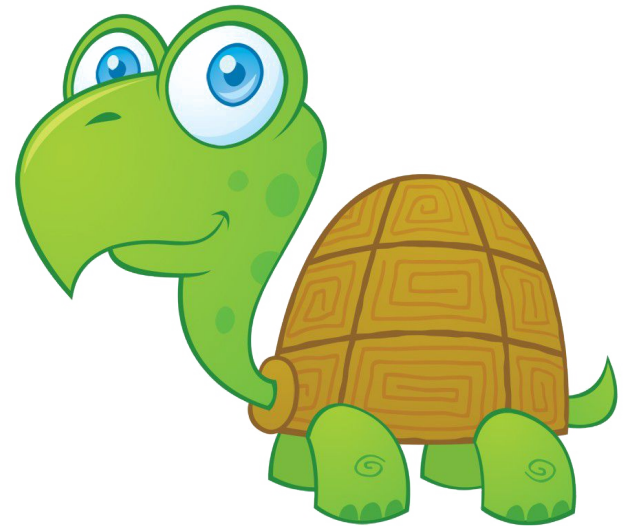
алгоритмика

PYTHON

Занятие 2

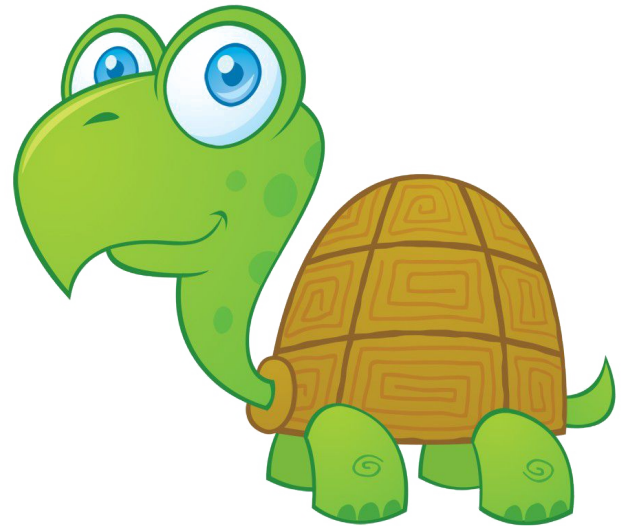
Сегодня на уроке

- Переменная – указатель на значение;
- Условие и условный оператор;
- Полная и неполная форма условного оператора;
- Операторы сравнения;
- Диалоговое окно.



Повторение

1. Для чего используется модуль Turtle?
2. Что делает черепаха при выполнении команды `forward`?
3. Что делает черепаха при выполнении команды `right`?
4. Как выполнить поворот фигуры?
5. Какая команда отвечает за цвет пера?



Задание на повторение

Напишите программу для рисования следующих фигур (длина стороны = 100):



Задание на повторение. Решение

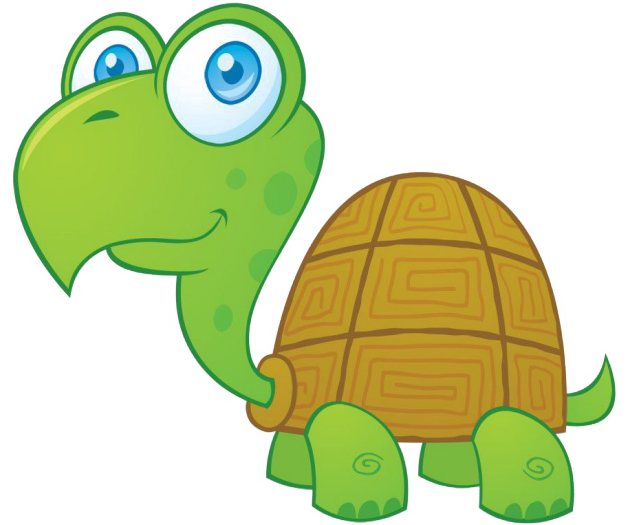
Напишите программу для рисования следующих фигур (длина стороны = 100):

```
from turtle import*           color("blue")
left(30)                       left(30)

forward(100)                   forward(100)
left(90)                       left(90)
forward(100)                   forward(100)
left(90)                       left(90)
forward(100)                   forward(100)
left(90)                       left(90)
forward(100)                   forward(100)
exitonclick()
```

Вопрос

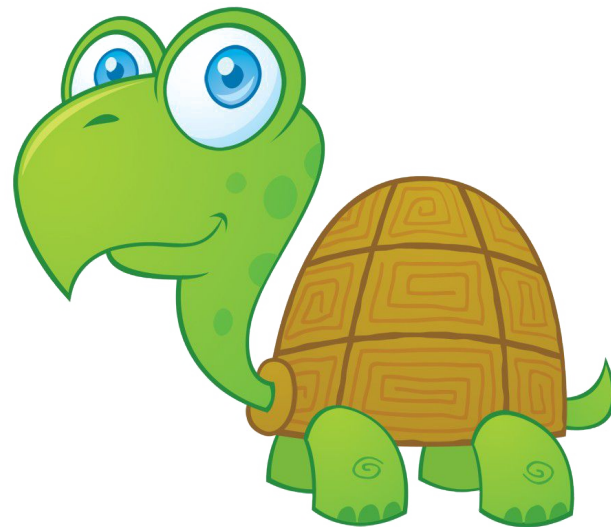
В каком количестве строк
нужно поменять значение,
чтобы получился рисунок с
квадратами 150×150 ?



Вопрос

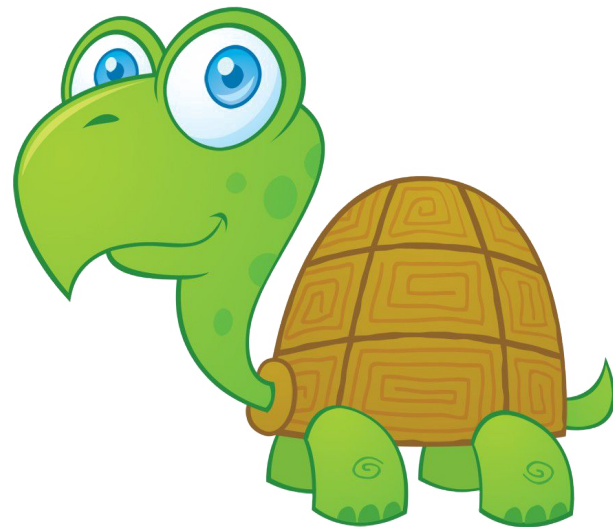
В каком количестве строк
нужно поменять значение,
чтобы получился рисунок с
квадратами 150×150 ?

8



Вопрос

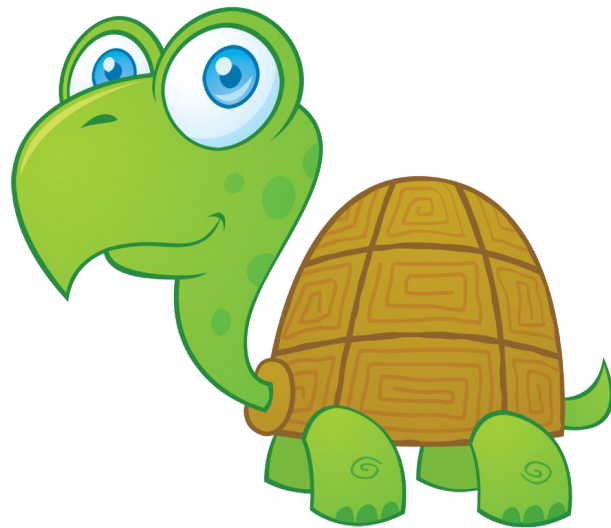
В какой команде необходимо изменить значение?



Вопрос

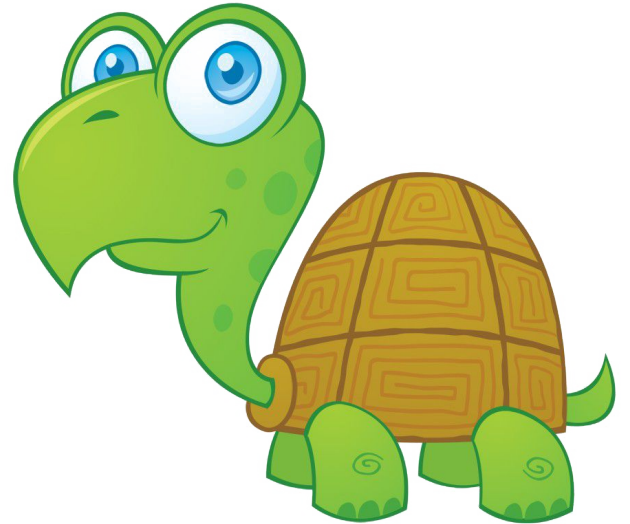
В какой команде необходимо изменить значение?

forward



Вопрос

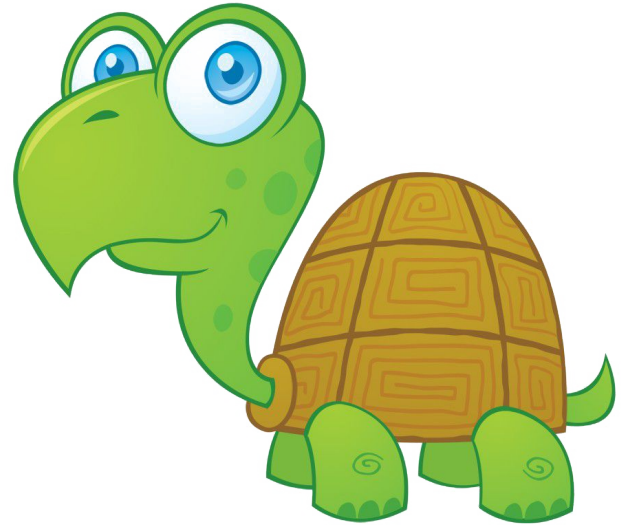
Возможно ли сделать данное изменение в другом месте с тем же эффектом для всей программы?



Вопрос

Возможно ли сделать данное изменение в другом месте с тем же эффектом для всей программы?

Да

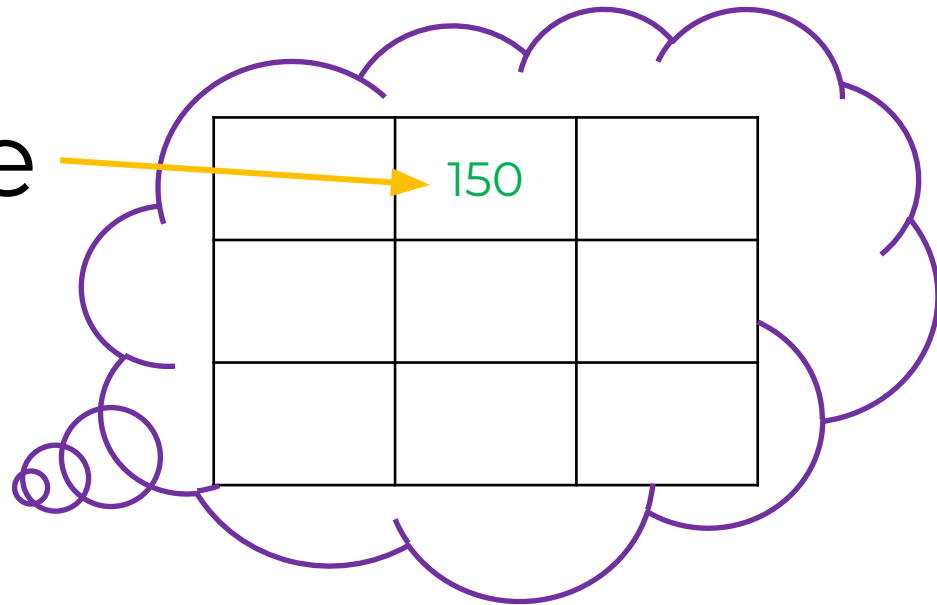


Переменная

Переменная – это имя указателя на область памяти.

`line` = `150`
имя указателя значение

line

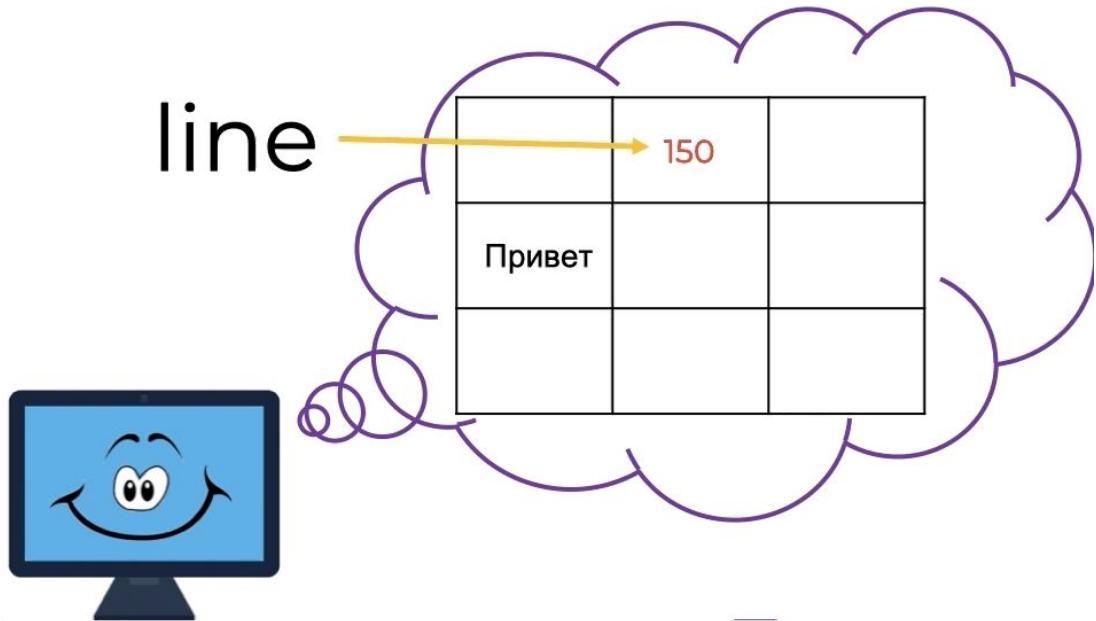


Переменная

Переменная – это имя указателя на область памяти.

```
line = 150
```

```
line = "Привет"
```



Использование переменной

Вместо прямого указания значения (числа) используется имя переменной.

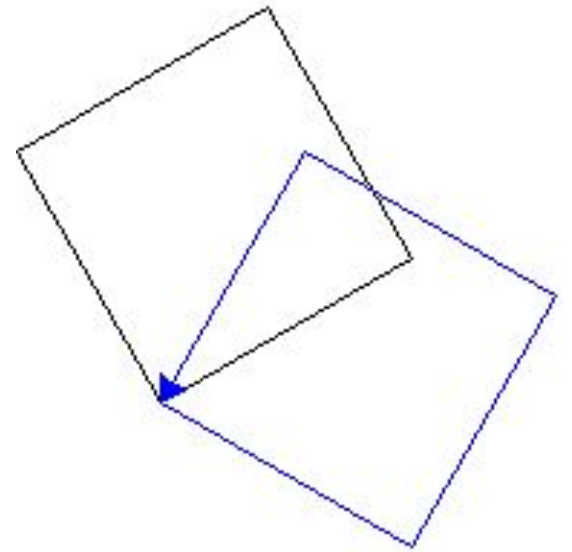
`line = 150` – создание указателя на значение

`forward(line)` – передача значения в команду



Задание

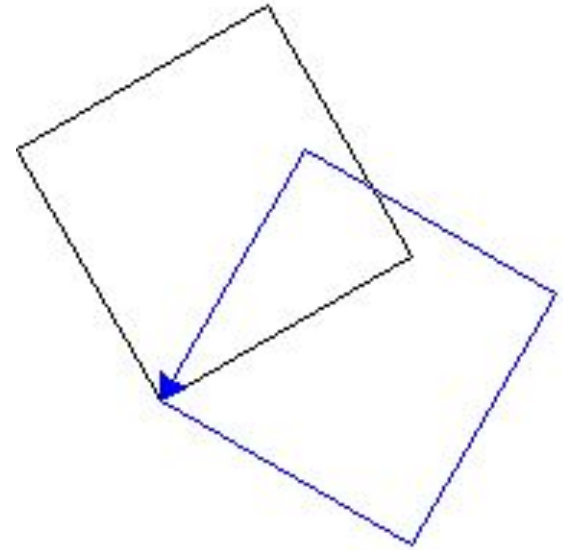
Измените программу для рисования указанной фигуры с использованием переменной `line`:



Задание. Решение

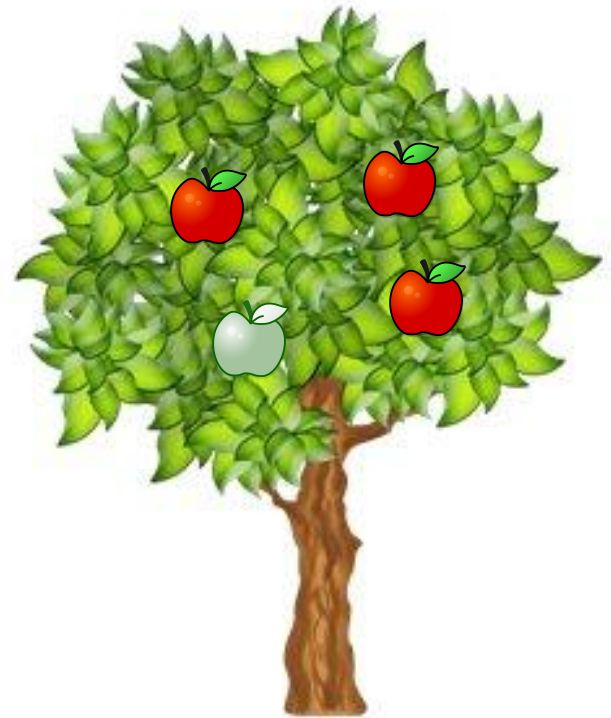
Измените программу для рисования указанной фигуры с использованием переменной `line`:

```
from turtle import*           color("blue")
left(30)                       left(30)
line = 150
forward(line)                 forward(line)
left(90)                      left(90)
forward(line)                 forward(line)
left(90)                      left(90)
forward(line)                 forward(line)
left(90)                      left(90)
forward(line)                 forward(line)
exitonclick()
```



Задача

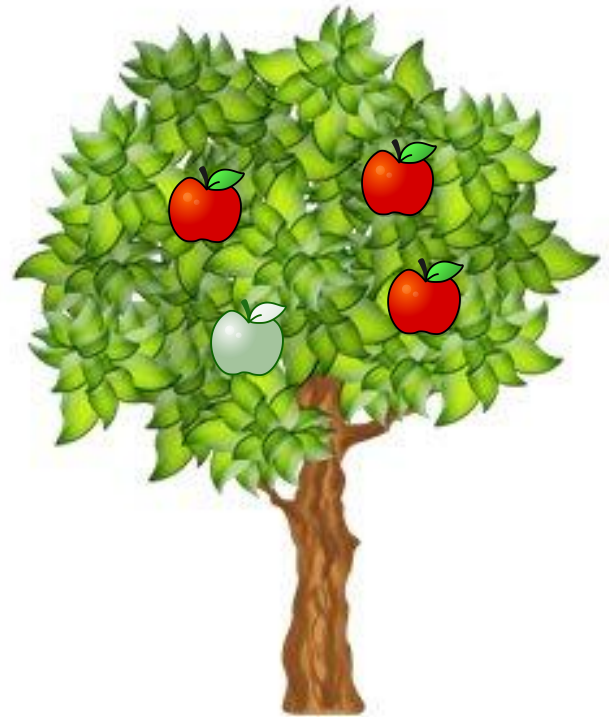
Соберите только созревшие яблоки.



Задача

Соберите только созревшие яблоки.

Чем отличаются созревшие плоды?

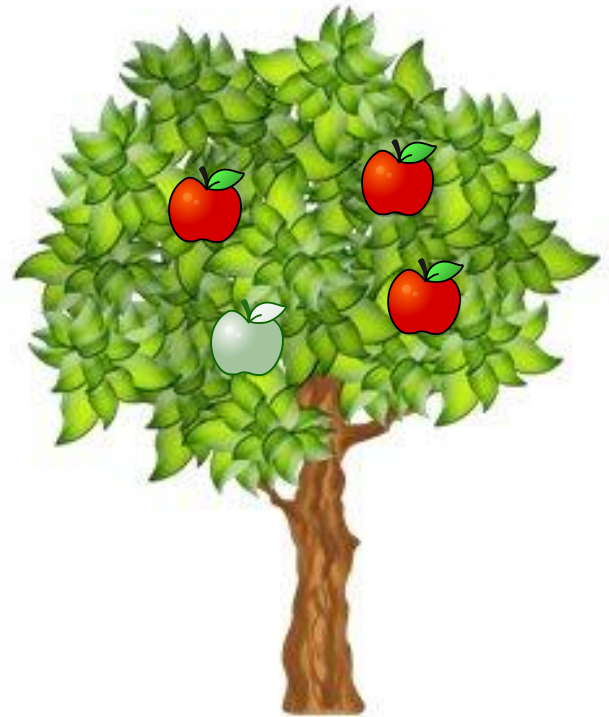


Задача

Соберите только созревшие яблоки.

Чем отличаются созревшие плоды?

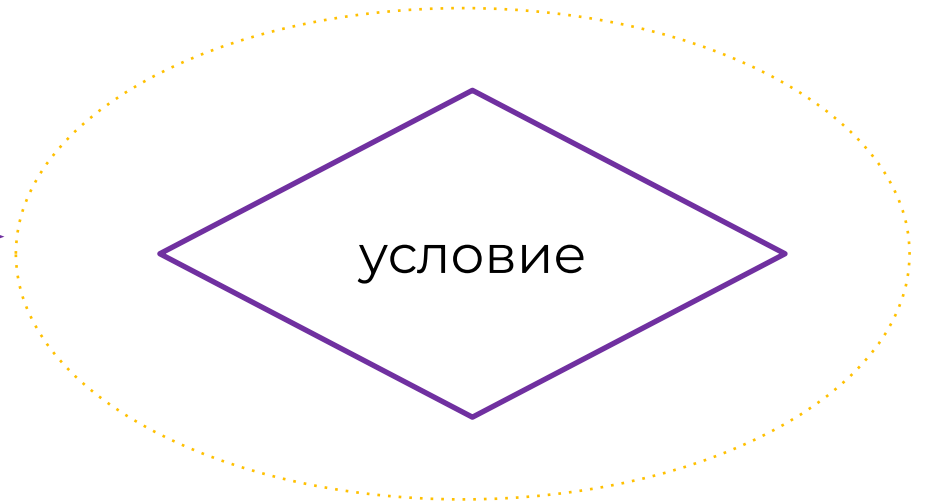
**Если яблоко красное –
сорвать**



Простое условие

Условие – логическое выражение, принимающее значение истинности или ложности.

Блок проверки
условия



Операторы сравнения

Простое условие состоит из 2 операндов: левого и правого, отношения между которыми необходимо проверить.

Оператор	Описание	Пример	Результат
<code>==</code>	Проверяет одинаково ли значение операндов, если одинаково – то условие является истинным	<code>5==5</code> <code>"hello"=="hello"</code> <code>text=="hello"</code>	
<code>></code>	Проверяет значение левого операнда, если он больше, чем правый – то условие является истинным	<code>5>2</code> <code>a>5</code> <code>a>b</code>	
<code>>=</code>	Проверяет больше или равно значение левого операнда, чем значение правого. Если да, то условие становится истинным	<code>5>=5</code> <code>2>=3</code>	

Операторы сравнения

Оператор	Описание	Пример	Результат
$<$	Проверяет значение левого операнда, если он меньше, чем правый – то условие является истинным	$3 < 2$ $a < 1$ $a < b$	
$<=$	Проверяет меньше или равно значение левого операнда, чем значение правого. Если да, то условие становится истинным.	$3 <= 3$ $a <= b$	
$!=$	Проверяет равны ли оба операнда. Если нет, то условие становится истинным.	$a != b$ $5 != 2$	

Условный оператор

Алгоритмическая конструкция, выполняющая действия в зависимости от выполнения или невыполнения условия.

Истина – условие выполнилось.

Ложь – условие **не** выполнилось.

Задание

Найдите условный оператор

Если на улице
дождь, возьми
зонт

$5==7$

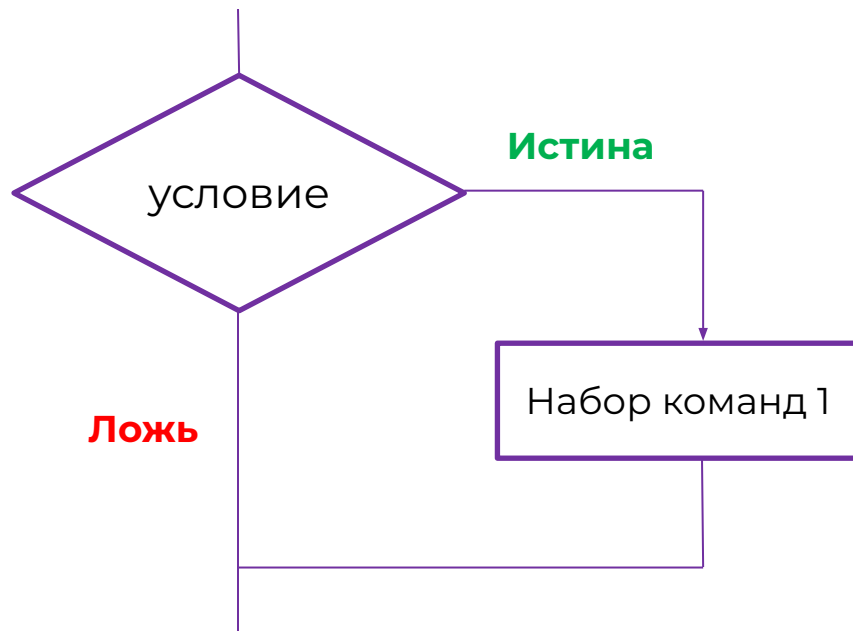
Собери все
фрукты

Если сегодня
вторник, то
завтра среда

Если $2=2$, ты
молодец, иначе
нет

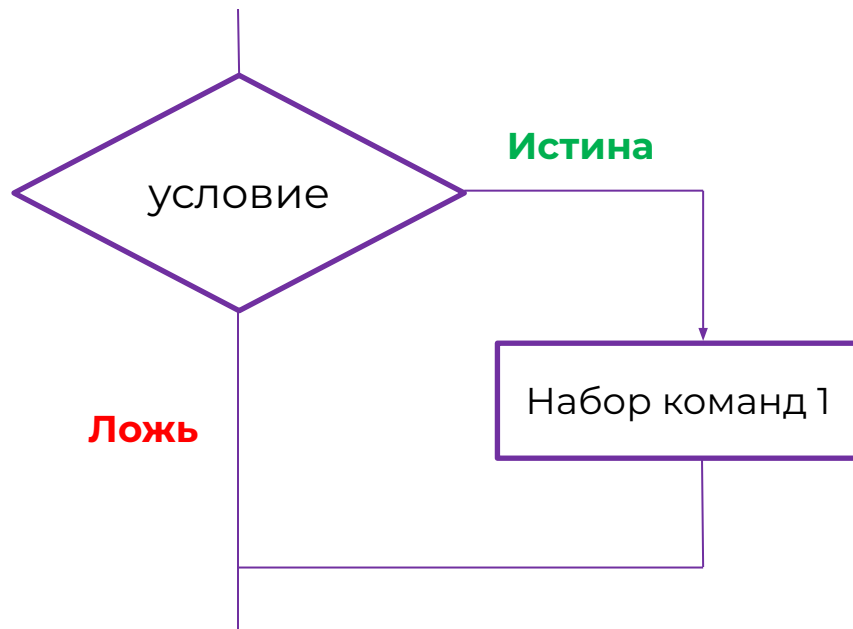


Условный оператор в неполной форме



Условный оператор в неполной форме

if <условие> :
набор команд 1



Табуляция в Python

`if` <условие> :

↔ набор команд 1

Важно! Блок команд, который должен выполняться в случае истинности условия, должен быть отделён знаком табуляции (Tab)

Задание

Найдите ошибку в программе.

```
from turtle import*  
text="красный"  
if text=="красный":  
    color("red")  
forward(100)  
exitonclick()
```

Если text указывает на «красный», нарисовать красную линию в 100 пикселей.



Задание. Решение

Найдите ошибку в программе.

```
from turtle import*
text="красный"
if text=="красный":
    color("red")
    ↔ forward(100)
exitonclick()
```

Если text указывает на «красный», нарисовать красную линию в 100 пикселей.

Задание

```
from turtle import*
text="красный"
if text=="красный":
    color("red")
    forward(100)
if text=="синий":
    color("blue")
    back(100)
exitonclick()
```

Что нужно изменить в программе, чтобы появилась синяя линия?



Задание. Решение

```
from turtle import*
text="синий"
if text=="красный":
    color("red")
    forward(100)
if text=="синий":
    color("blue")
    back(100)
exitonclick()
```

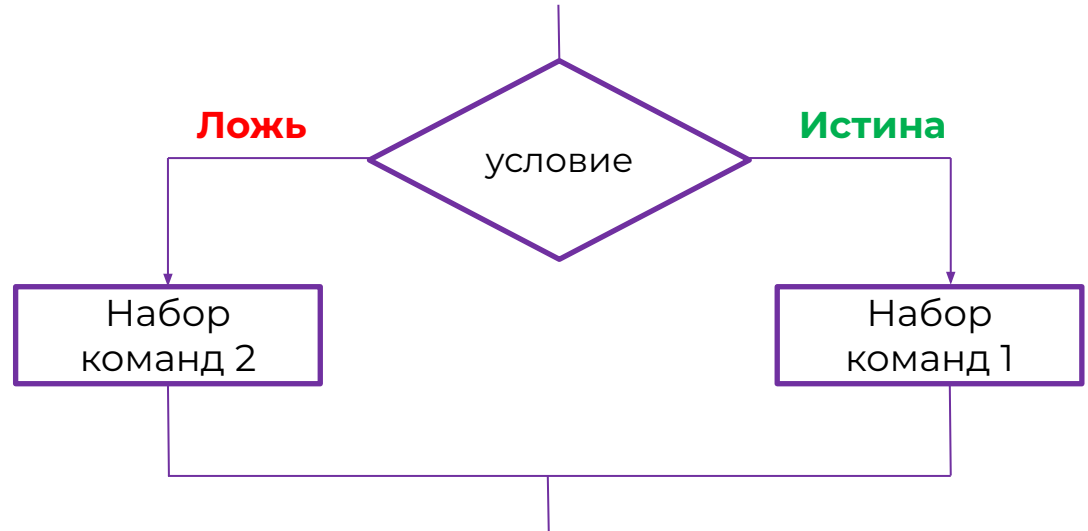
Что нужно изменить в программе, чтобы появилась синяя линия?



Условный оператор в полной форме



Условный оператор в полной форме



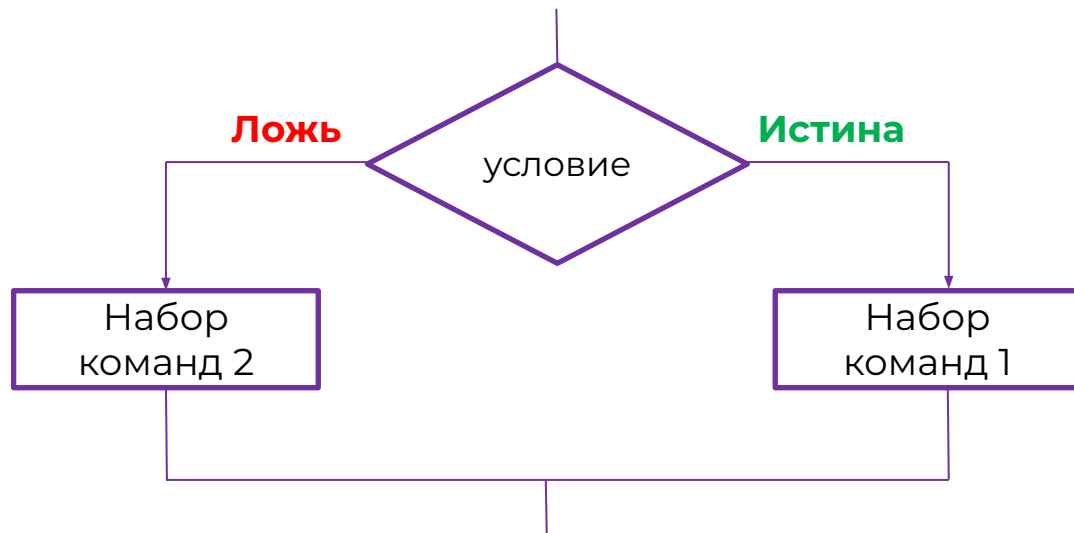
Условный оператор в полной форме

if <условие> :

набор команд 1

else:

набор команд 2



Задание

Измените предыдущую программу так, что, если ввели красный, рисуем красную линию. Во всех остальных случаях (например, если ввели жёлтый) синюю.



Задание

Измените предыдущую программу так, что, если ввели красный, рисуем красную линию. Во всех остальных случаях (например, если ввели жёлтый) синюю.

```
from turtle import*
text="красный"
if text=="красный":
    color("red")
    forward(100)
else:
    color("blue")
    back(100)
exitonclick()
```

Конструкция if – elif - else

if условие_1 :

набор команд_1

elif условие_2 :

набор команд_2

else:

набор команд_3

elif позволяет указать альтернативное условие, которое требуется проверить.



Задание

Если значение переменной равно 3 – нарисовать треугольник, если 4 – квадрат, иначе вывести – sorry.



Задание. Решение

Если значение переменной равно 3 – нарисовать треугольник, если 4 – квадрат, иначе вывести – sorry.

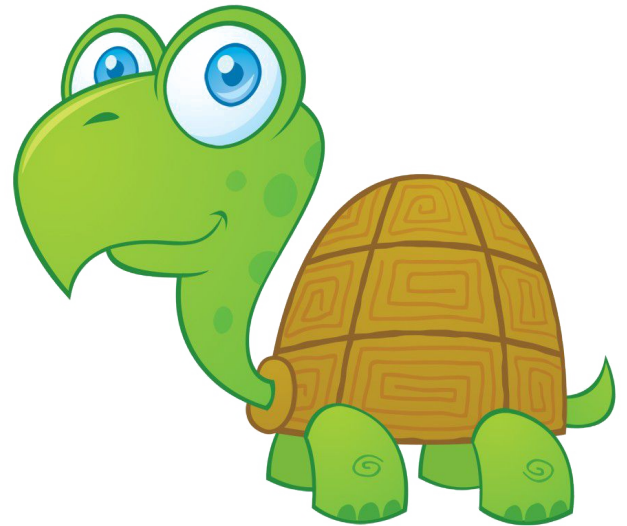
```
from turtle import*
text=3
if text==3:
    forward(100)
    left(135)
    forward(140)
    left(135)
    forward(105)
```

```
elif text==4:
    forward(100)
    left(90)
    forward(100)
    left(90)
    forward(100)
    left(90)
    forward(100)
else:
    print("sorry")
exitonclick()
```



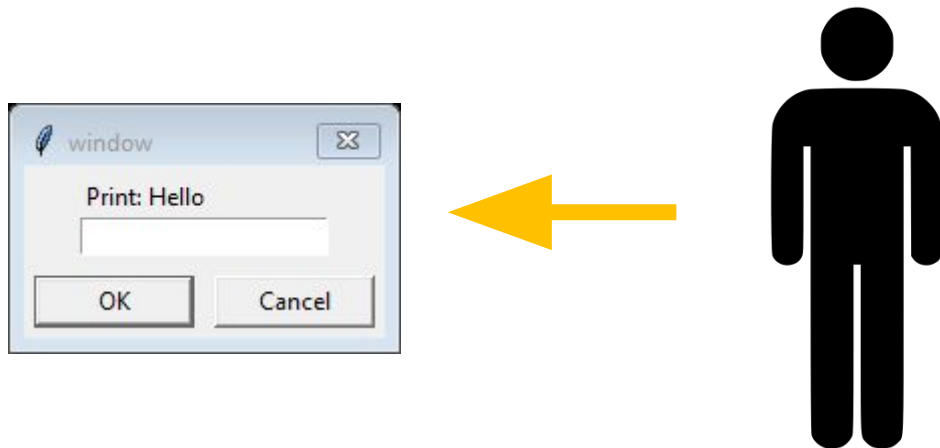
Вопрос

Как можно передавать значения в программу, не изменяя каждый раз код программы?



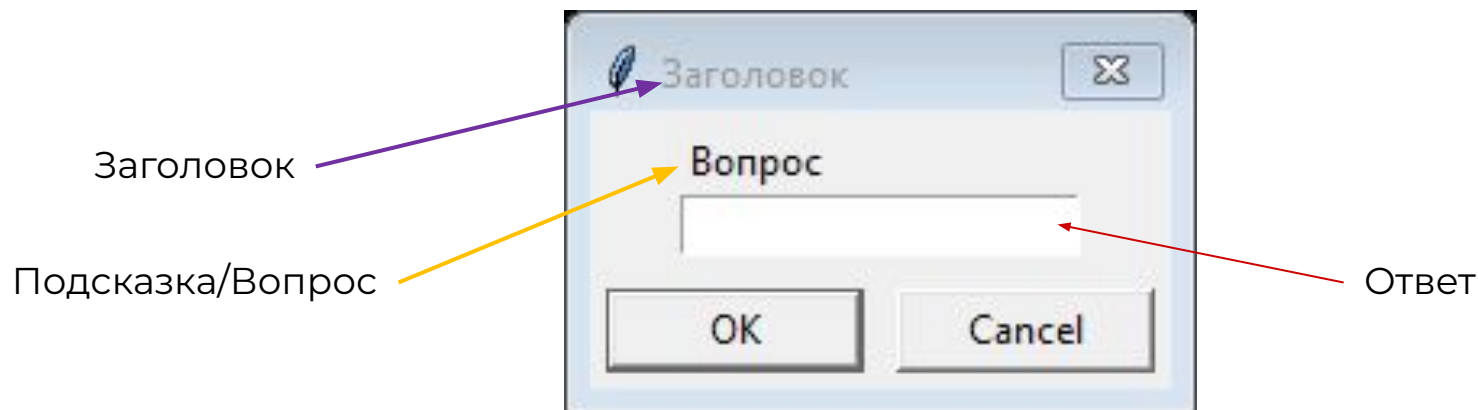
Диалоговое окно

Диалоговое окно – окно, предназначенное для вывода информации и (или) получения ответа от пользователя.



Диалоговое окно в Turtle

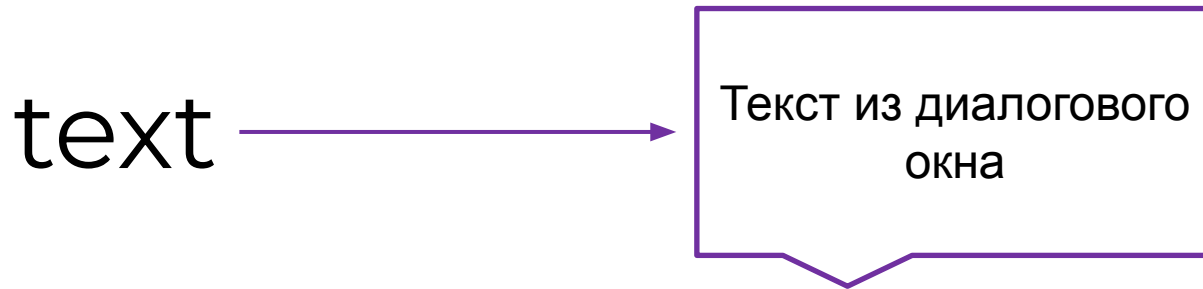
- **textinput(“заголовок_окна”, “подсказка/вопрос”)** – вызывает диалоговое окно для ввода строки.
- В команде указывается 2 параметра:
 1. “Заголовок_окна” – введённый текст, отображается в заголовке всплывающего окна.
 2. “Подсказка/вопрос” – описание, какую информацию необходимо ввести.



Использование ответа

- После получение ответа, мы можем использовать его в своей программе.
- Создадим указатель на ответ из окна.

```
text=textinput("Заголовок", "Вопрос")
```



Задание

Измените программу с использованием диалогового окна:
если значение переменной равно 3 – нарисовать
треугольник, если 4 – квадрат, иначе вывести – sorry.



Задание. Решение

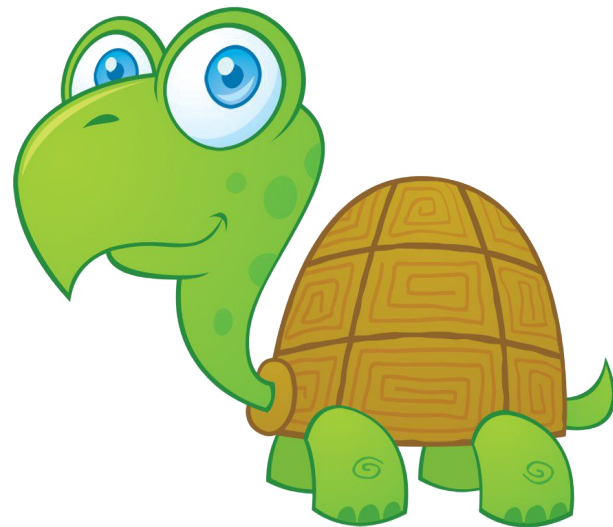
Если значение переменной равно 3 – нарисовать треугольник, если 4 – квадрат, иначе вывести – sorry.

```
from turtle import*
text=textinput("Фигура", "3-треугольник, 4-квадрат")
if text=="3":
    forward(100)
    left(135)
    forward(140)
    left(135)
    forward(105)
elif text=="4":
    forward(100)
    left(90)
    forward(100)
    left(90)
    forward(100)
    left(90)
    forward(100)
else:
    print("sorry")
exitonclick()
```



Вопрос

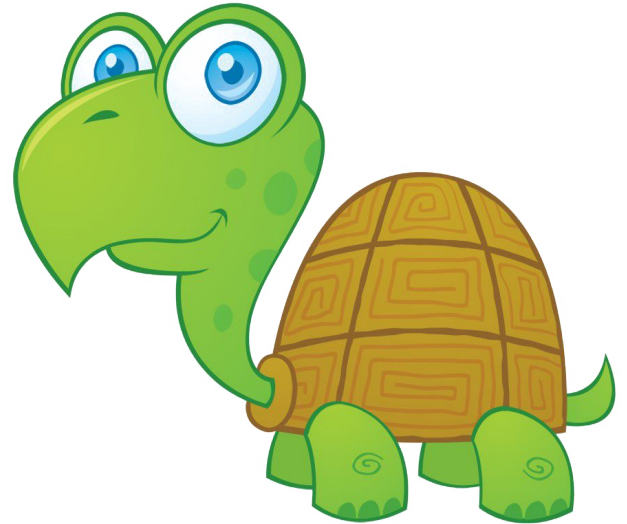
Какие вопросы мы можем задать в наших программах?



Вопрос

Какие вопросы мы можем задать в наших программах?

1. Какого размера фигуру нарисовать?
2. Каким цветом рисовать?
3. Какой толщины линии?
4. Сколько фигур нарисовать?
5. Какую фигуру нарисовать?



Задание

Создайте программу с запросом на количество фигур (от 1 до 3), цвет линии и толщину линии

