Strukturierte Programmierung

① Gib zu den folgenden Befehlen aus Python die Struktogramm-Elemente an:

Python

```
1.) forward(100)

2.) for i in range(6):
        forward(50)
        left(60)
```

Python

```
3.) if i == 0:
        color(„blue")
    else:
        color(„yellow")

4.) while size < 50:
        circle(size)
        size = size + 10
```

② Erstelle vom Unterprogramm muster2() ein Struktogramm.

Python

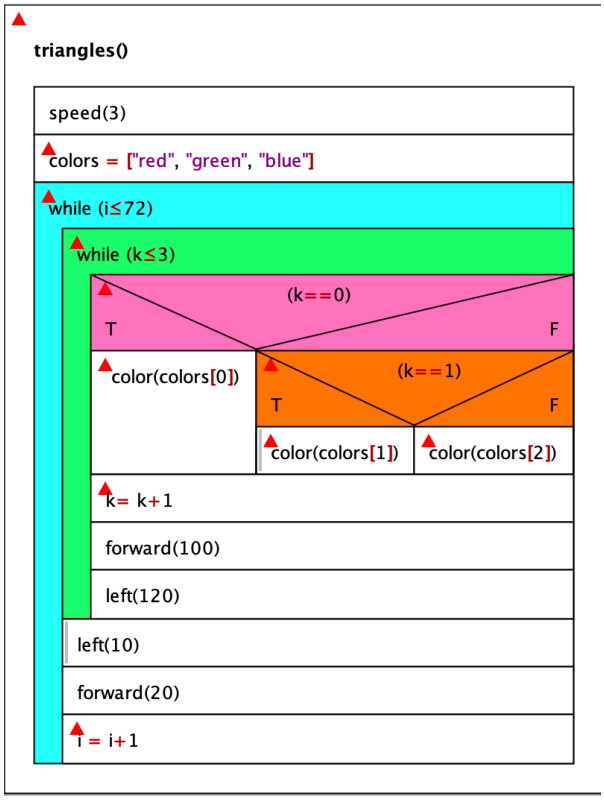
```
from turtle import *

def muster2():
    speed(30)
    bgcolor("black")  # Background color
    colors = ["blue", "red"]  # Alternative color choice
    for i in range(12):
        color(colors[i % 2])  # Alternating colors
        circle(50)
        right(30)
    color("yellow")
    penup()
    goto(-10, -10)
    pendown()
    size = 20
    for i in range(6):
        size = 20
        while size < 50:
            circle(size)
            size += 10
        left(60)
        penup()
        goto(0,0)
        pendown()

muster2()
done()
```


Angaben zu den Urhebern und Lizenzbedingungen der einzelnen Bestandteile dieses Dokuments finden Sie unter
https://www.tutory.de/entdecken/dokument/python-strukturen

③ Erstelle aus dem Struktogramm eine Programm. Denk daran zuerst from turtle import * zu schreiben, das Programm mit def zu beginnen und und zuletzt triangles() aufzurufen.



④ Hier gibt es noch je eine Übung.

Achtung!
Nur **def boxes()** soll ins Struktogramm:

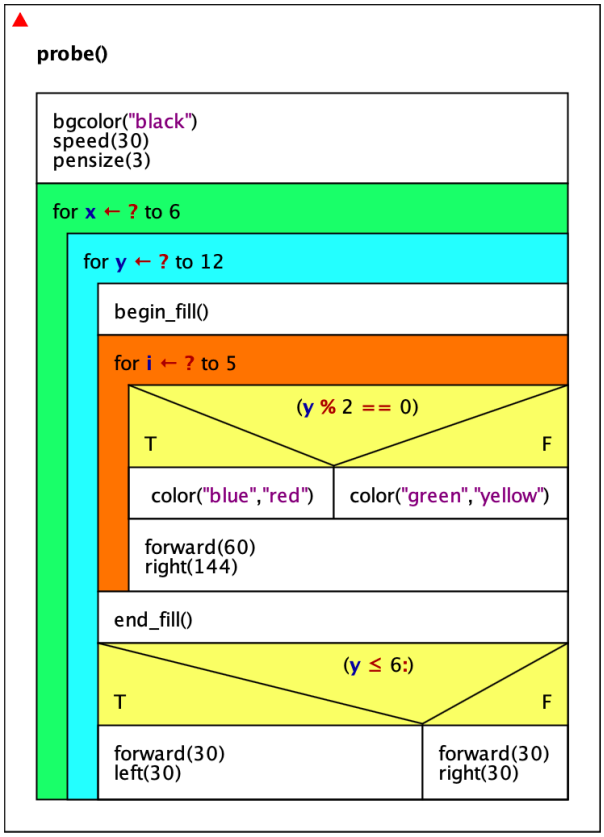
```python
from turtle import *

def boxes():
    speed(0)
    bgcolor("white")
    for y in range(3):
        for x in range(3):
```

Achtung!
Nur def soll ins Struktogramm.

```python
            if (x + y) % 2 == 0:
                color("red")
            else:
                color("blue")
            for _ in range(4):
                forward(50)
                right(90)

boxes()
done()
```



Angaben zu den Urhebern und Lizenzbedingungen der einzelnen Bestandteile dieses Dokuments finden Sie unter
https://www.tutory.de/entdecken/dokument/python-strukturen