

СЕТЬ КЛУБОВ
ПРОГРАММИРОВАНИЯ
И РОБОТОТЕХНИКИ



Урок №10

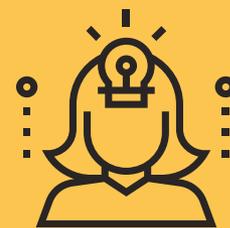
Тема урока:

Бинарная логика.

Группа 9-10 лет

ЦЕЛЬ УРОКА

- познакомиться с основами алгебры логики (бинарная логика),
- научиться применять логические операторы на практике



ОБСУЖДЕНИЯ

1. Как организовать развилку в алгоритме?
2. Какие варианты оператора условия бывают?
3. В каких ситуациях в игре можно применить оператор условия?

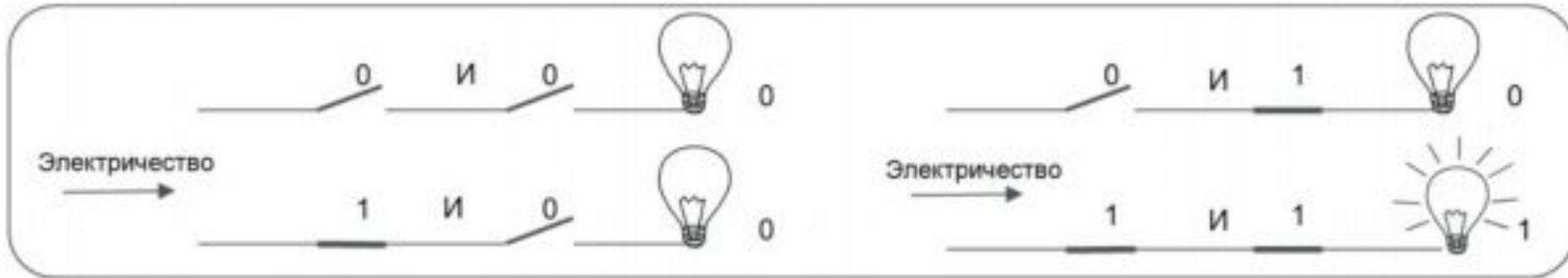


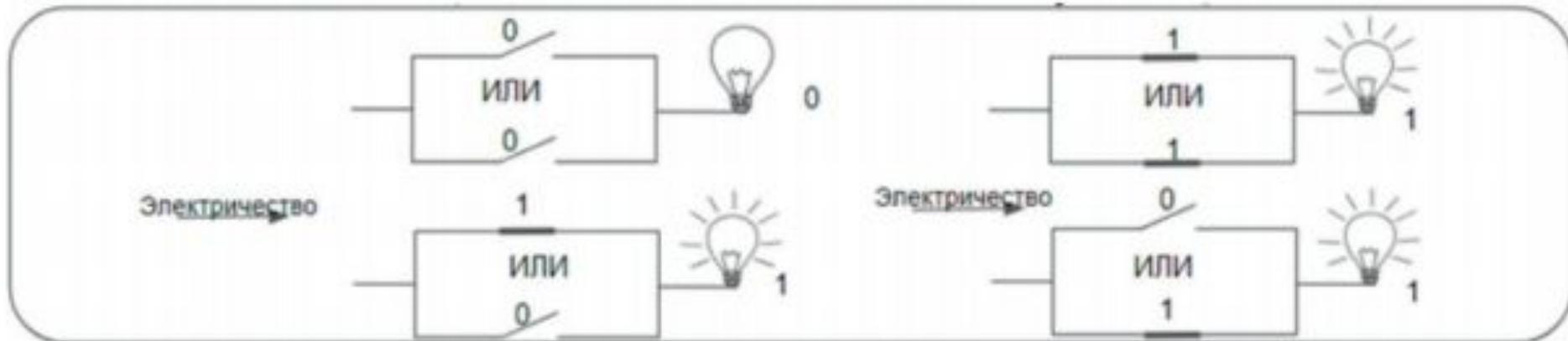
Условие - это некоторое предложение, для которого можно определить его истинность или ложность.

Такие предложения называют **-высказываниями**.

Принято, что **ноль - ложь**,
а **единица - истина**.

Т.е. Для любого условия можно записать его “статус” выполнения - 0 или 1.



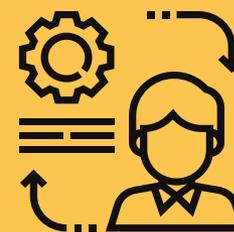


ОБСУЖДЕНИЯ



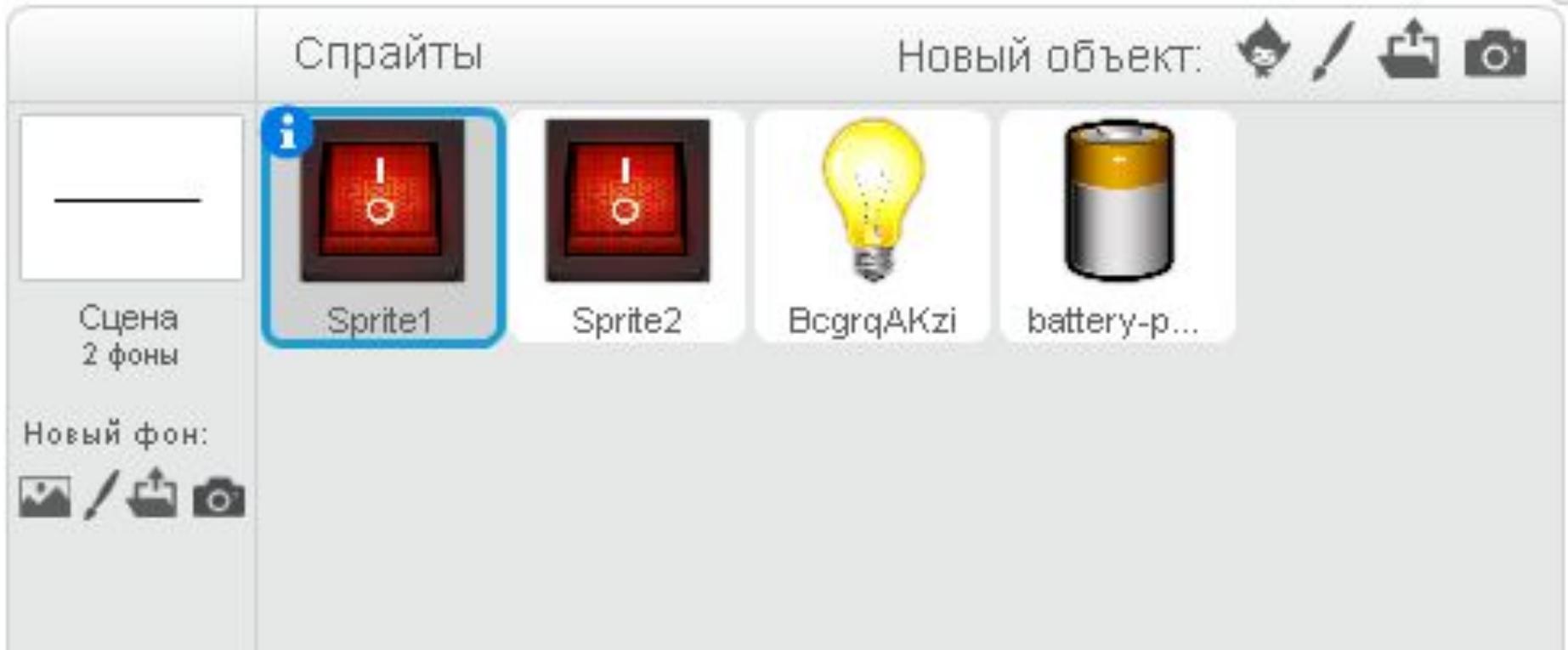
САМОСТОЯТЕЛЬНАЯ РАБОТА

Создадим проекты
демонстрирующие работу
логических операторов
И\ИЛИ\НЕ



Создадим виртуальную электроцепь.

Выберите фон фоне для последовательного подключения, так же нам понадобятся спрайты двух выключателей, лампочки и источника питания.



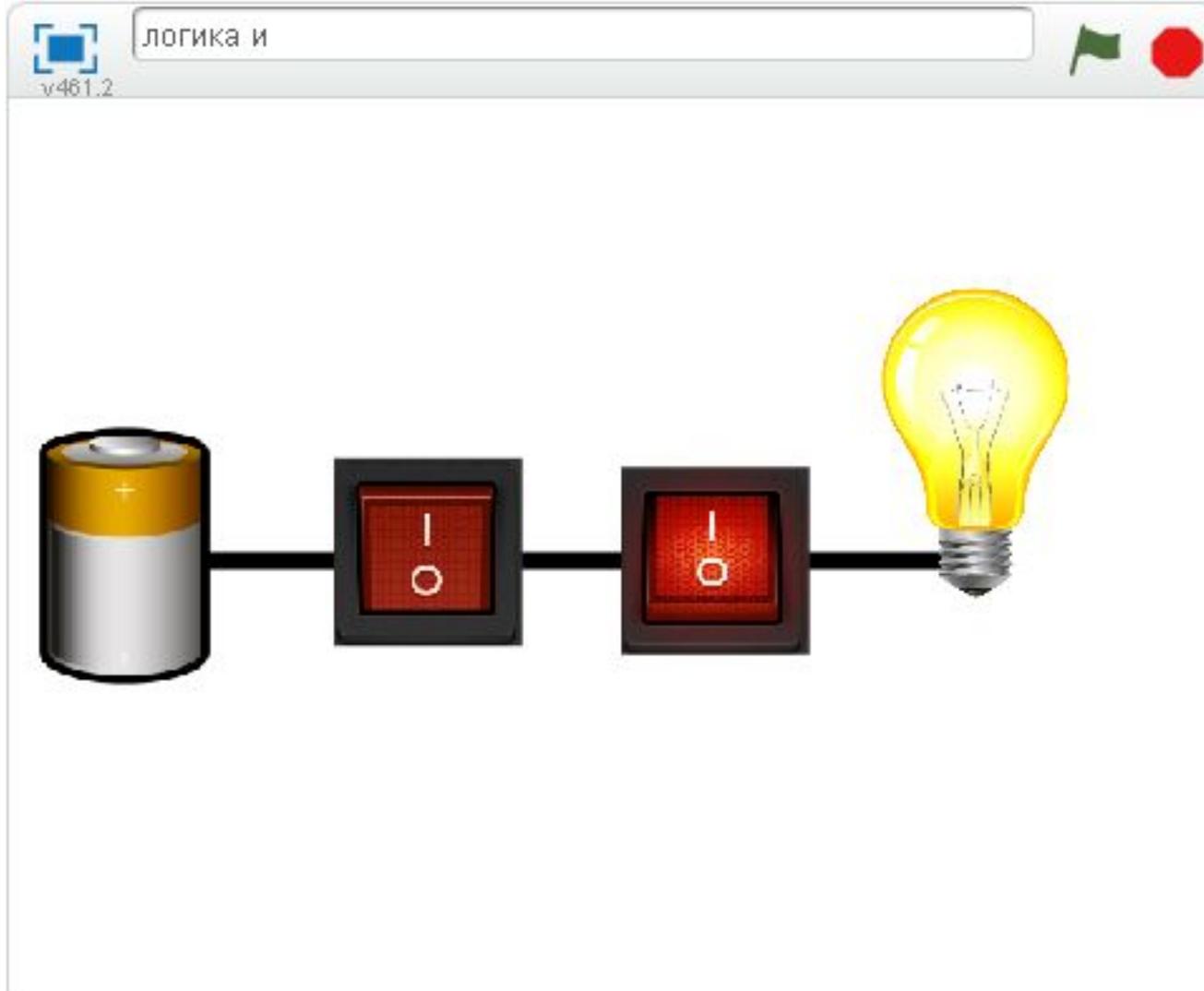
У спрайта “выключатель”
два костюма -
включен/выключен.



У спрайта “лампочка” то же
два костюма - светится/не
светится



Выстроим из спрайтов электрическую цепь с последовательным подключением.



Создадим две переменные, что хранить в них состояние выключателя вкл-1. выкл-0.

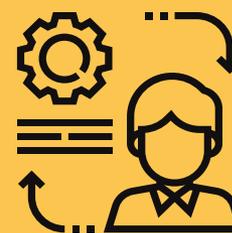


Напишем стартовые скрипты для выключателей. для первого:



САМОСТОЯТЕЛЬНАЯ РАБОТА

Составьте стартовый скрипт для второго выключателя самостоятельно.



Стартовый скрипт для второго выключателя:



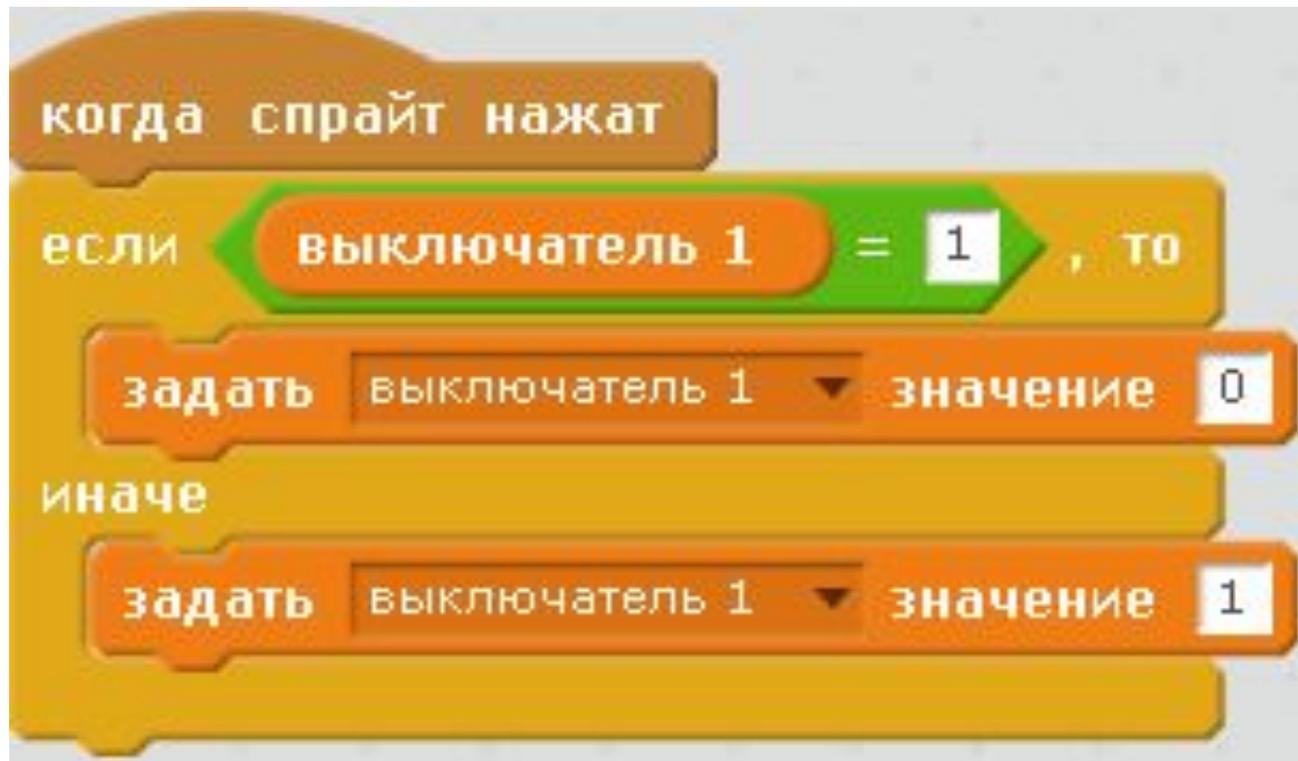
```
когда щелкнут по [флаг]
  задать [выключатель 2] значение [0]
  сменить костюм на [выкл]
  всегда
    если [выключатель 2 = 1], то
      сменить костюм на [вкл]
    иначе
      сменить костюм на [выкл]
```

The image shows a Scratch script for a switch. It starts with a 'when green flag clicked' event block. The first block is 'set switch 2 to 0'. The second block is 'change costume to выкл'. This is followed by a 'forever' loop containing an 'if-else' conditional block. The 'if' part checks 'switch 2 = 1' and changes the costume to 'вкл'. The 'else' part changes the costume to 'выкл'.

При клике на выключатель, он должен либо включаться, либо выключаться, при этом меняется значение переменной, отвечающей за состояние выключателя.

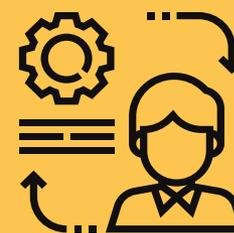
С 0 на 1, с 1 на 0.

Запишем это через условие.

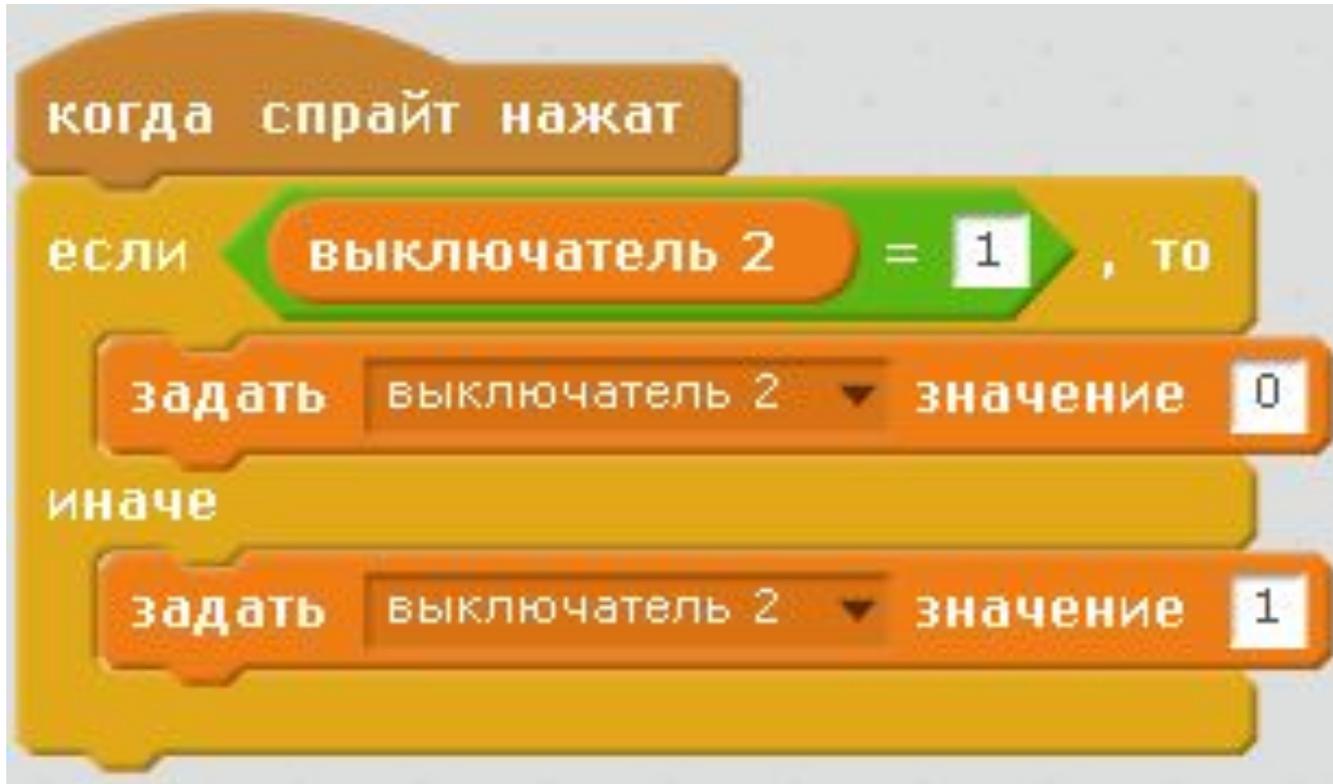


САМОСТОЯТЕЛЬНАЯ РАБОТА

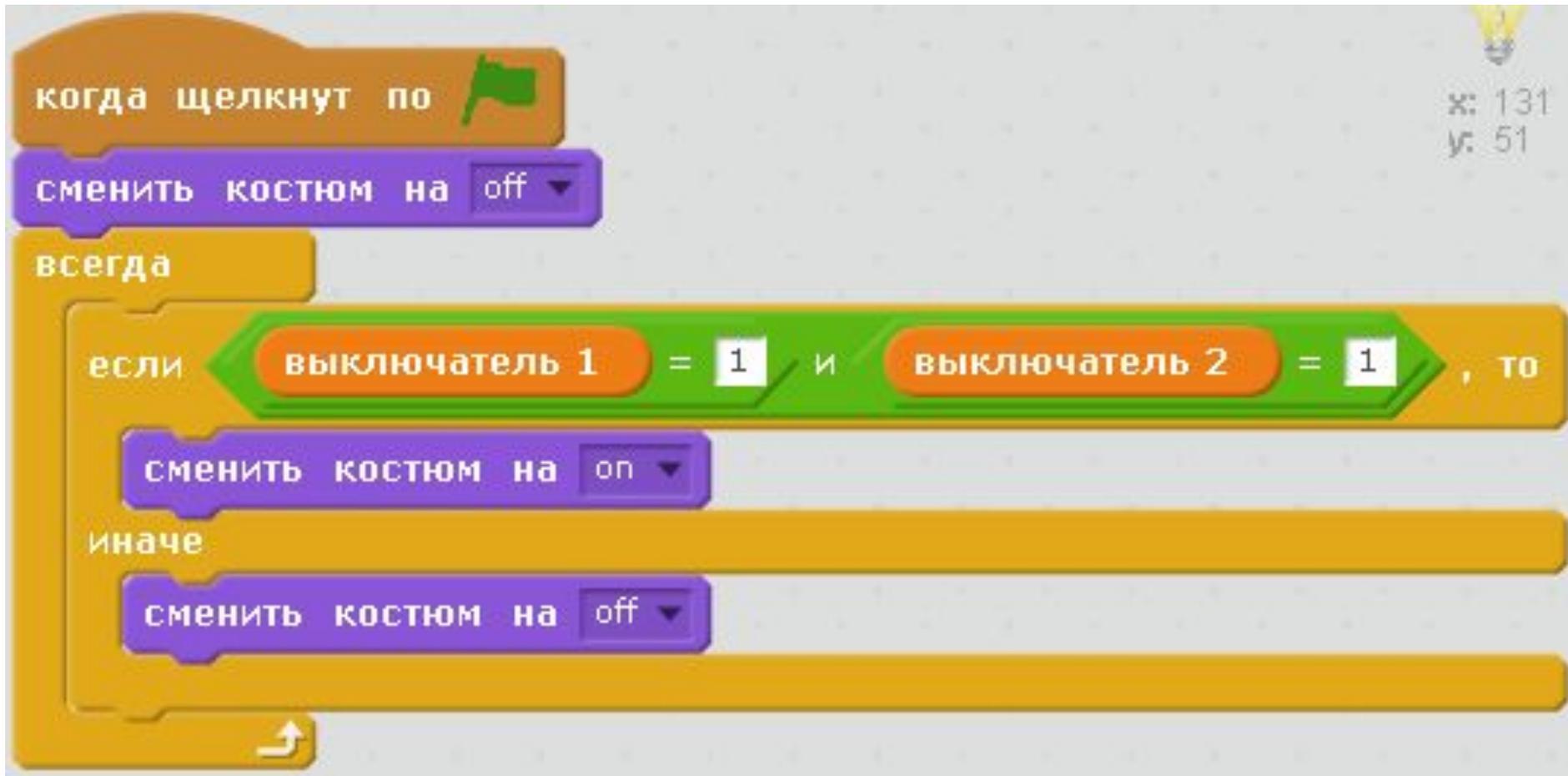
Составьте скрипт для второго выключателя самостоятельно.



Скрипт для второго выключателя.



Стартовый скрипт для лампочки.



The image shows a Scratch script for a light bulb. The script starts with a 'when green flag clicked' event block. This is followed by a 'set costume to off' block. Then, there is a 'forever' loop block. Inside the loop, there is an 'if' block with the condition 'switch 1 = 1 and switch 2 = 1'. If the condition is true, the script sets the costume to 'on'. If the condition is false, it sets the costume to 'off'. The script ends with a return arrow block.

когда щелкнут по 

сменить костюм на

всегда

если = и = , то

сменить костюм на

иначе

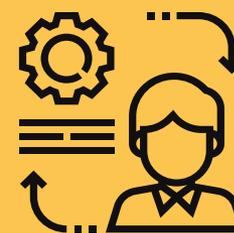
сменить костюм на



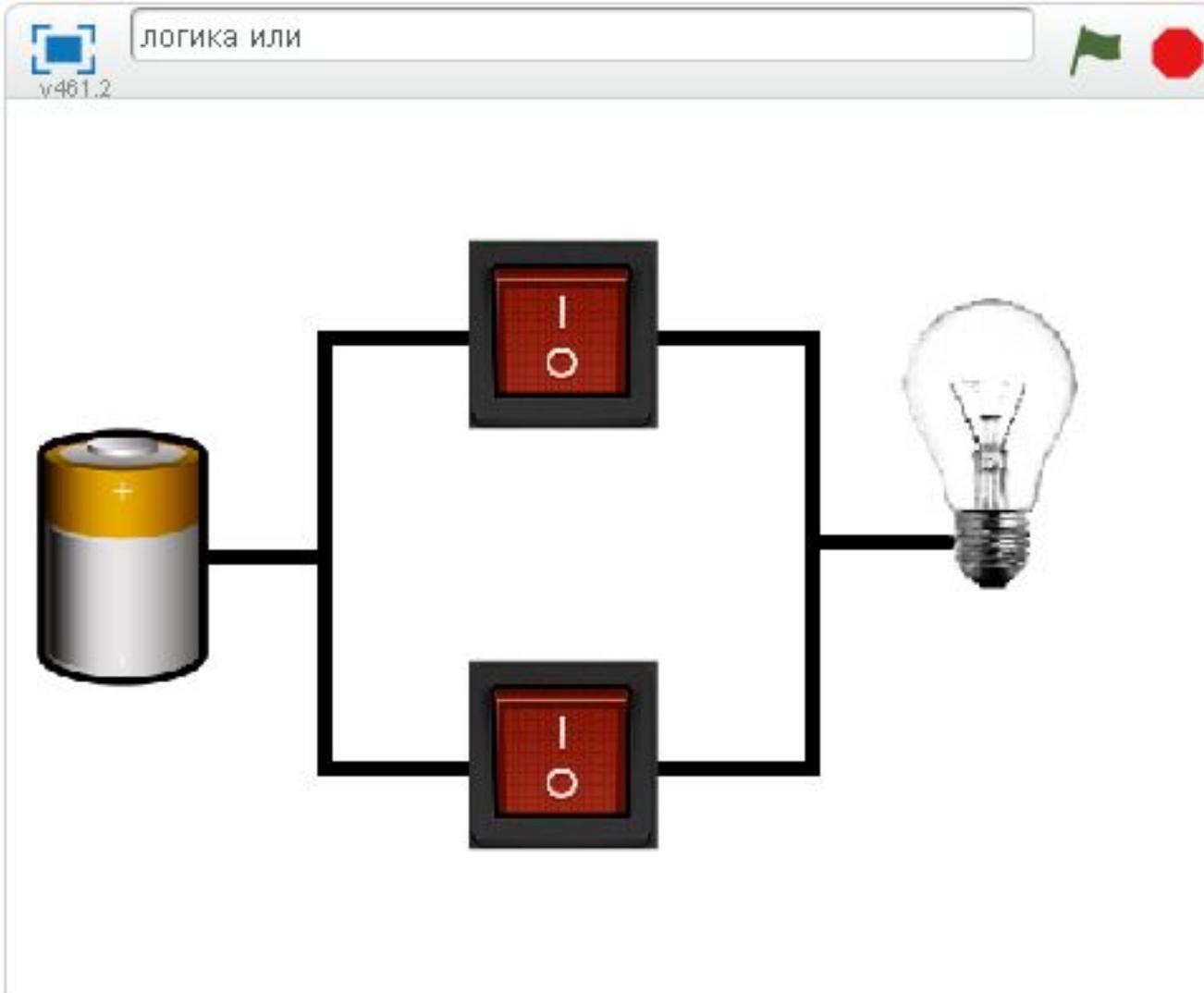
x: 131
y: 51

САМОСТОЯТЕЛЬНАЯ РАБОТА

Составим схему для оператора ИЛИ



Выберем фон с параллельным подключением выключателей и расставим спрайты.



Составим скрипт для лампочки, используем оператор или:



```
когда щелкнут по [флаг]
  сменить костюм на [off]
  всегда
    если [выключатель 1 = 1] или [выключатель 2 = 1], то
      сменить костюм на [on]
    иначе
      сменить костюм на [off]
```

The image shows a Scratch script for a light bulb. It starts with a 'when clicked' event block. The first block is 'change costume to off'. This is followed by a 'forever' loop. Inside the loop, there is an 'if' block with the condition 'switch 1 = 1 or switch 2 = 1'. If the condition is true, the costume is changed to 'on'. If the condition is false, the costume is changed to 'off'.

ОБСУЖДЕНИЯ

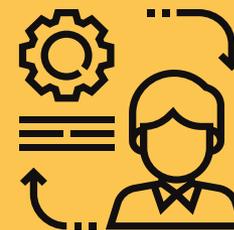
Что изменилось по сравнению с оператором И?

В каком случае лампочка будет гореть?

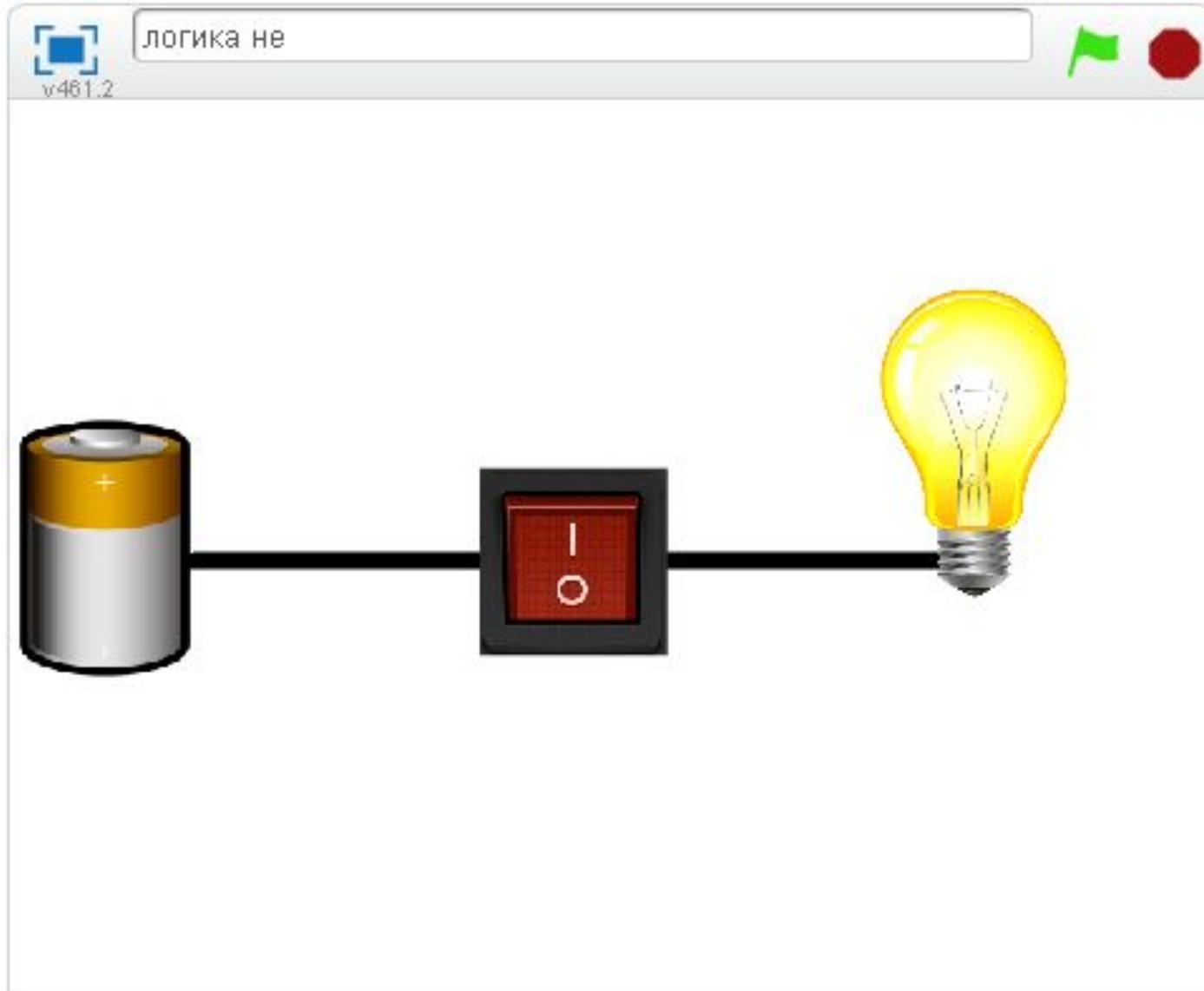


САМОСТОЯТЕЛЬНАЯ РАБОТА

Составим схему для оператора НЕ



Нам понадобится только один выключатель

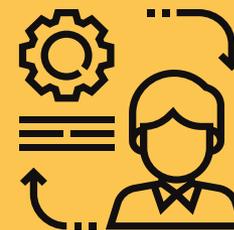
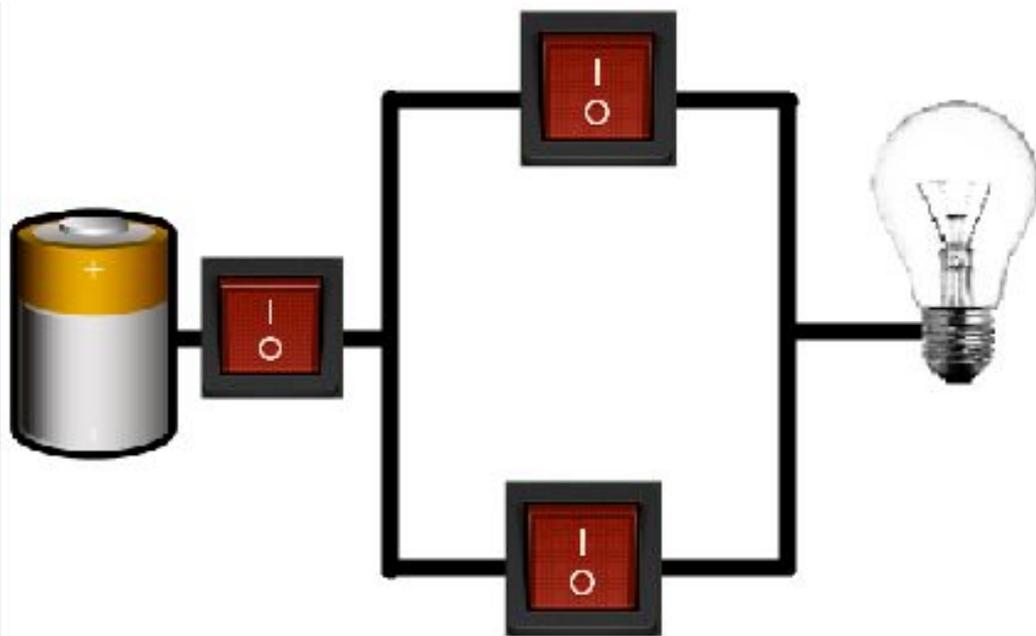


Составим скрипт для лампочки

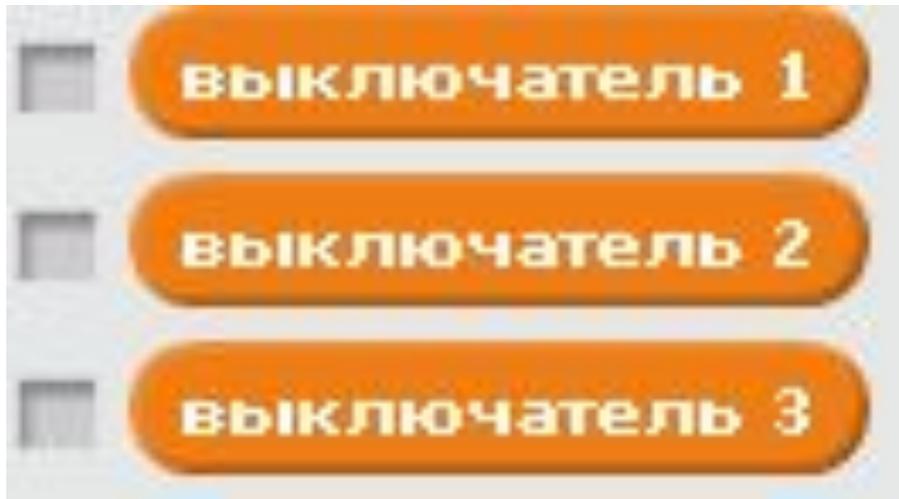


САМОСТОЯТЕЛЬНАЯ РАБОТА

Составьте самостоятельно проект для следующей схемы:



Так как в схеме три выключателя, нужны три переменных



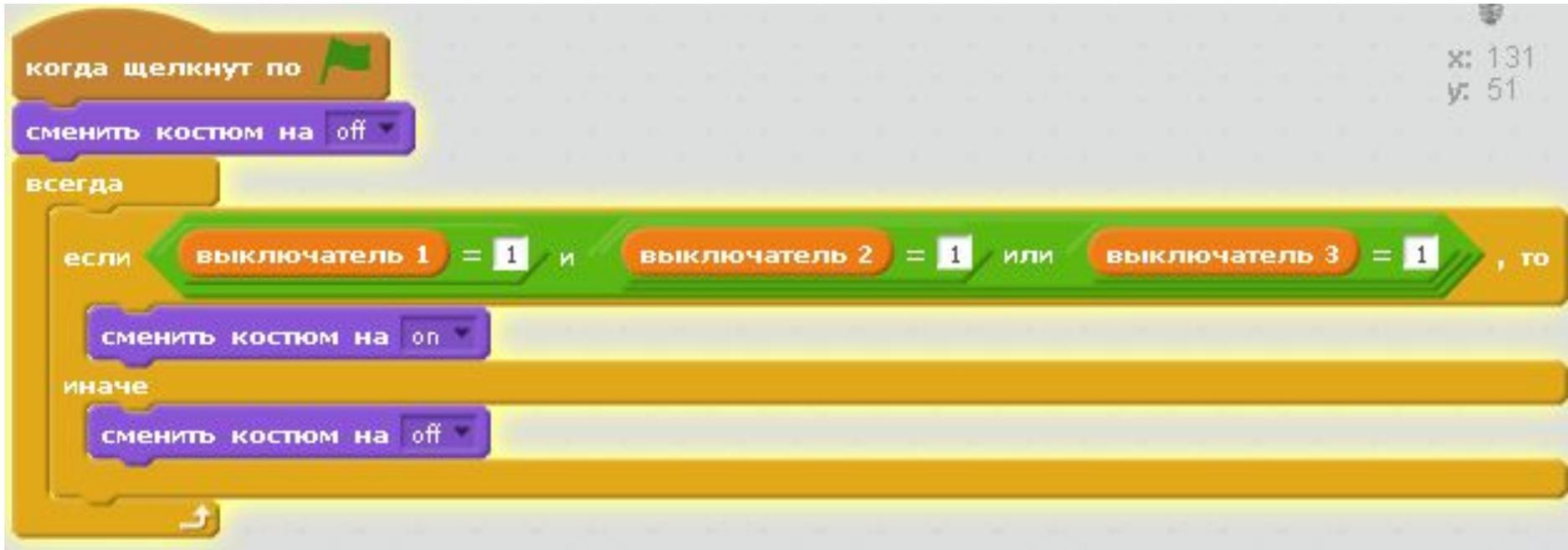
Выключатели работают так же как и в трех предыдущих проектах, например скрипт для первого выключателя:



```
когда щелкнут по [флаг]
  задать [выключатель 1] значение [0]
  сменить костюм на [выкл]
  всегда
    если [выключатель 1] = [1], то
      сменить костюм на [вкл]
    иначе
      сменить костюм на [выкл]
```

```
когда спрайт нажат
  если [выключатель 1] = [1], то
    задать [выключатель 1] значение [0]
  иначе
    задать [выключатель 1] значение [1]
```

Скрипт для лампочки



The image shows a Scratch script for a light bulb. The script starts with a 'when green flag clicked' event block. This is followed by a 'set costume to off' block. Then, there is a 'forever' loop block. Inside the loop, there is an 'if' block with the condition: 'switch 1 = 1 and switch 2 = 1 or switch 3 = 1'. If this condition is true, the script sets the costume to 'on'. If the condition is false, it sets the costume to 'off'. The script also includes a small coordinate display in the top right corner showing 'x: 131' and 'y: 51'.

```
когда щелкнут по [флаг]
сменить костюм на [off]
всегда
  если [выключатель 1 = 1 и выключатель 2 = 1 или выключатель 3 = 1], то
    сменить костюм на [on]
  иначе
    сменить костюм на [off]
```

ОБСУЖДЕНИЯ

В каких случаях лампочка будет светиться?



ПЕРЕМЕНА

5 
МИНУТ



**Отдохните
от компьютера**



попейте воды



сходите в туалет

САМОСТОЯТЕЛЬНАЯ РАБОТА

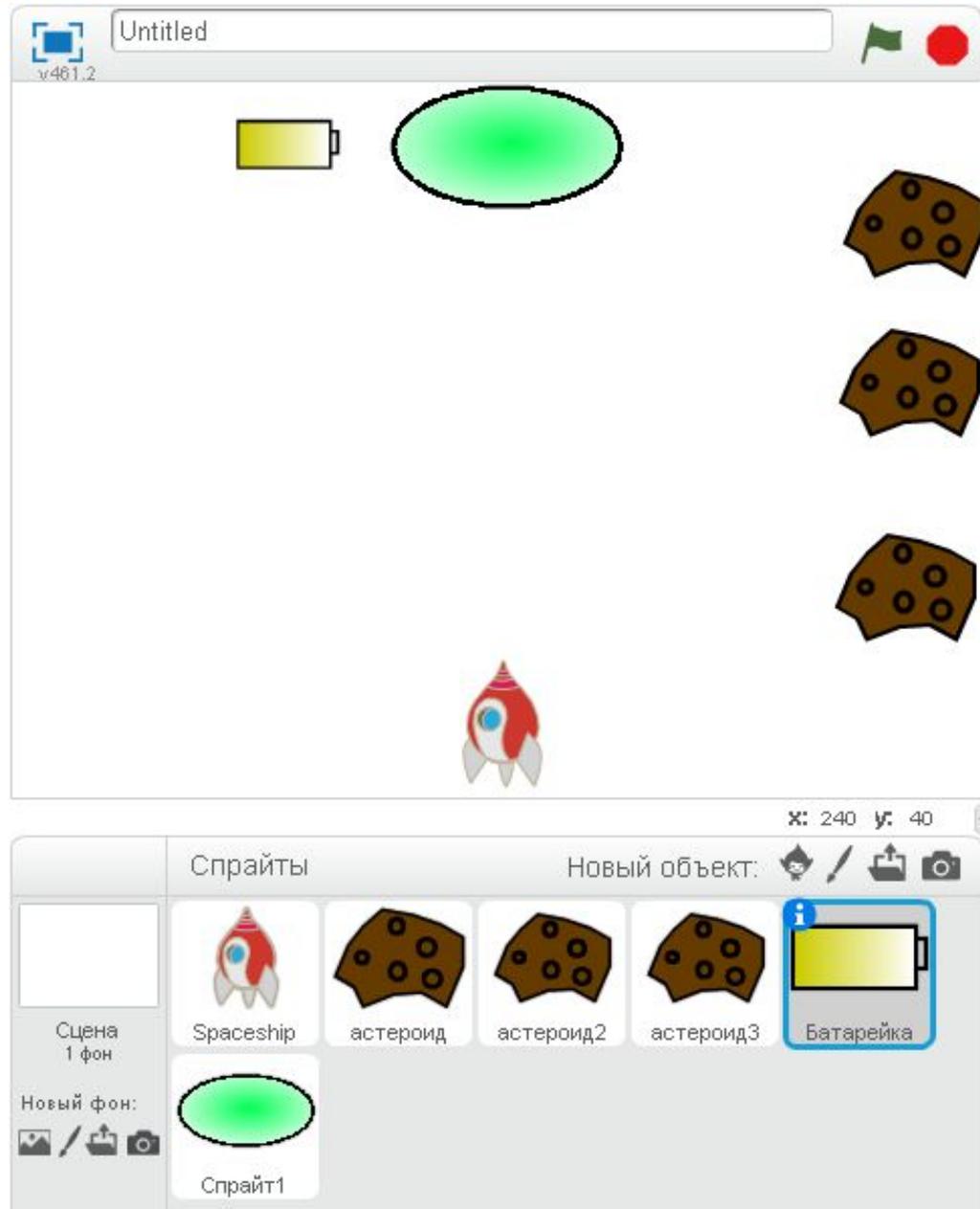
Заряди ракету

Создадим игру, в которой необходимо зарядить ракету. Для этого ракете необходимо долететь до зарядки не коснувшись астероидов и одновременно касаться места зарядки и батарейки. Ракета управляется с клавиатуры.

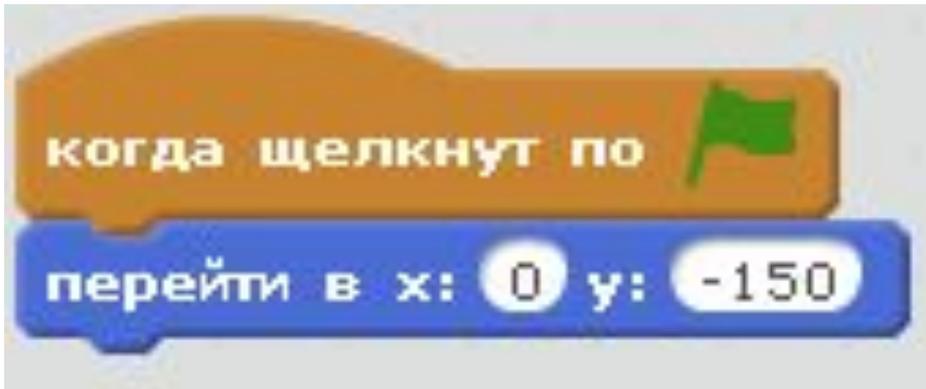


Создадим спрайты:

- Ракета
- Астериды
- Зарядка
- Батарейка.



Стартовый скрипт для ракеты:



Скрипты для управления ракетой с клавиатуры:



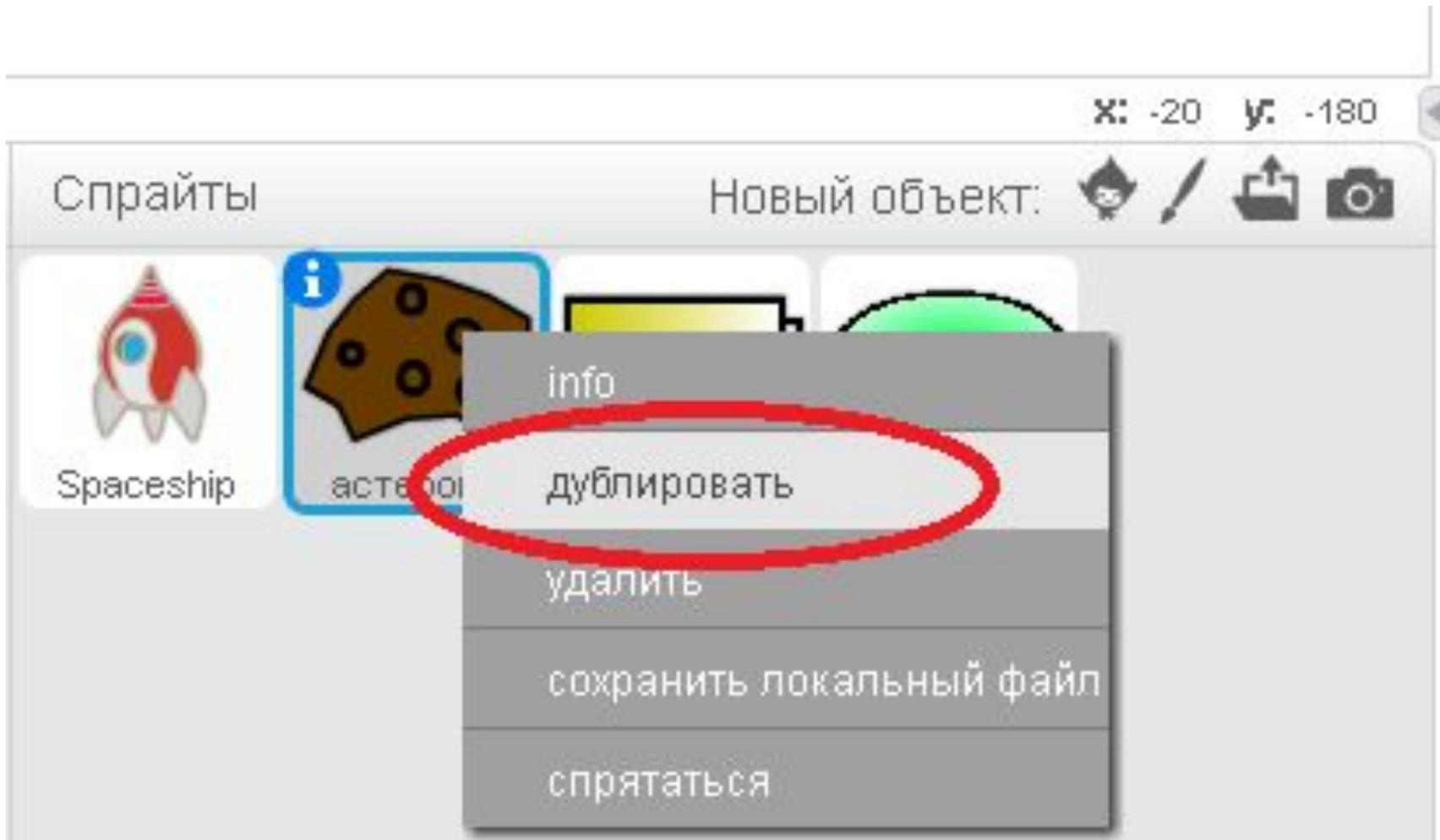
Составим скрипт для астероида.



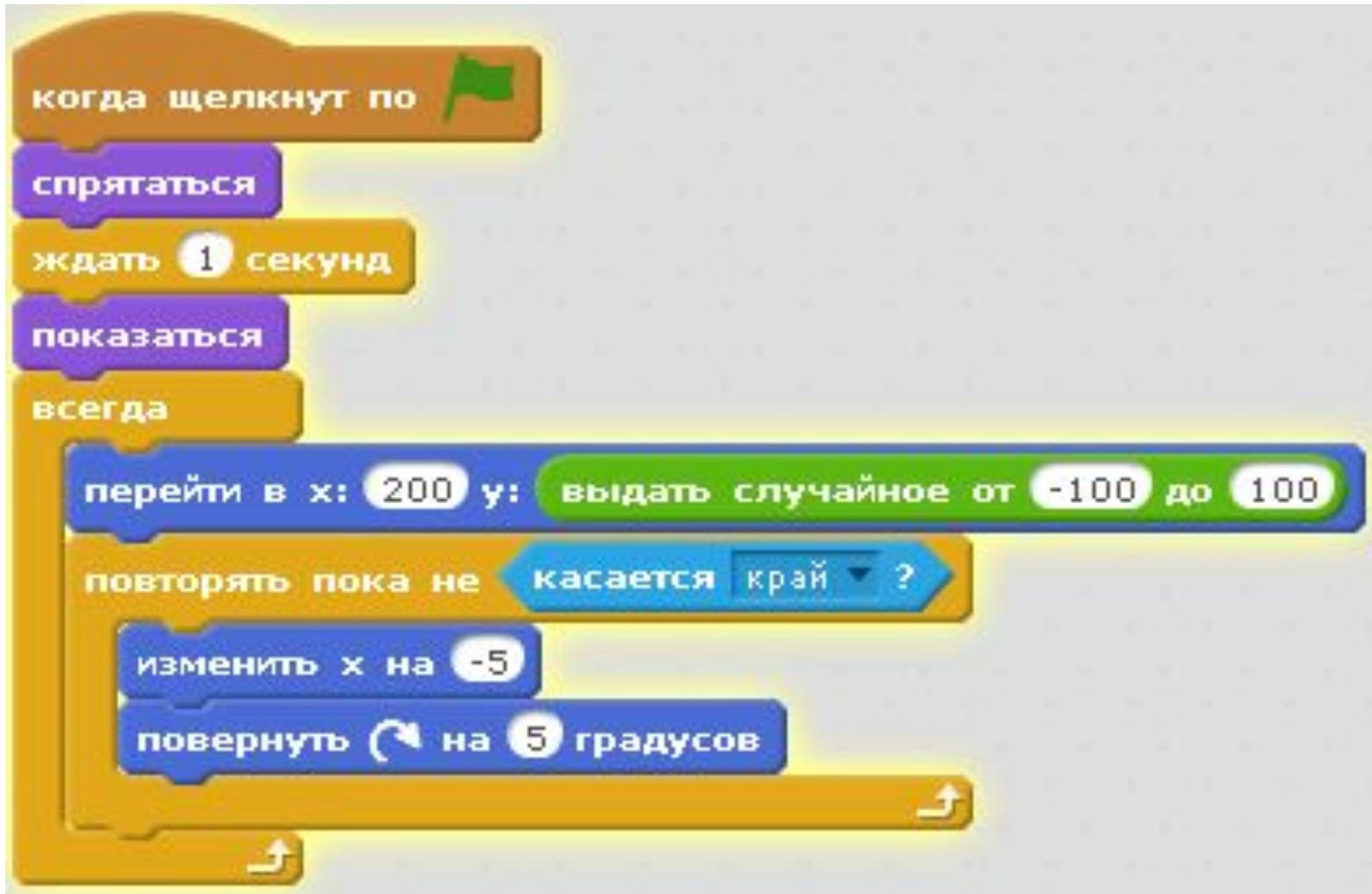
```
when green flag clicked
  always
    go to x: 200 y: random from -100 to 100
    repeat until not touching edge?
      change x by -5
      rotate 5 degrees
```

The image shows a Scratch script for an asteroid. It starts with a 'when green flag clicked' event block. This is followed by an 'always' loop block. Inside the loop, the first block is 'go to x: 200 y: random from -100 to 100'. The second block is a 'repeat until not touching edge?' loop. Inside this loop, there are two blocks: 'change x by -5' and 'rotate 5 degrees'. The script ends with a return arrow at the bottom of the 'repeat until' block.

Создадим еще два астероида, продублировав первый.



Добавим задержку для второго и третьего астероида.
Для второго 1 секунду, для третьего 2.

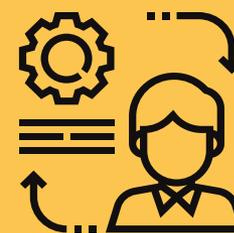


```
when green flag clicked
  hide
  wait 1 seconds
  show
  loop
    go to x: 200 y: random from -100 to 100
    repeat until touches edge?
      change x by -5
      turn 5 degrees
```

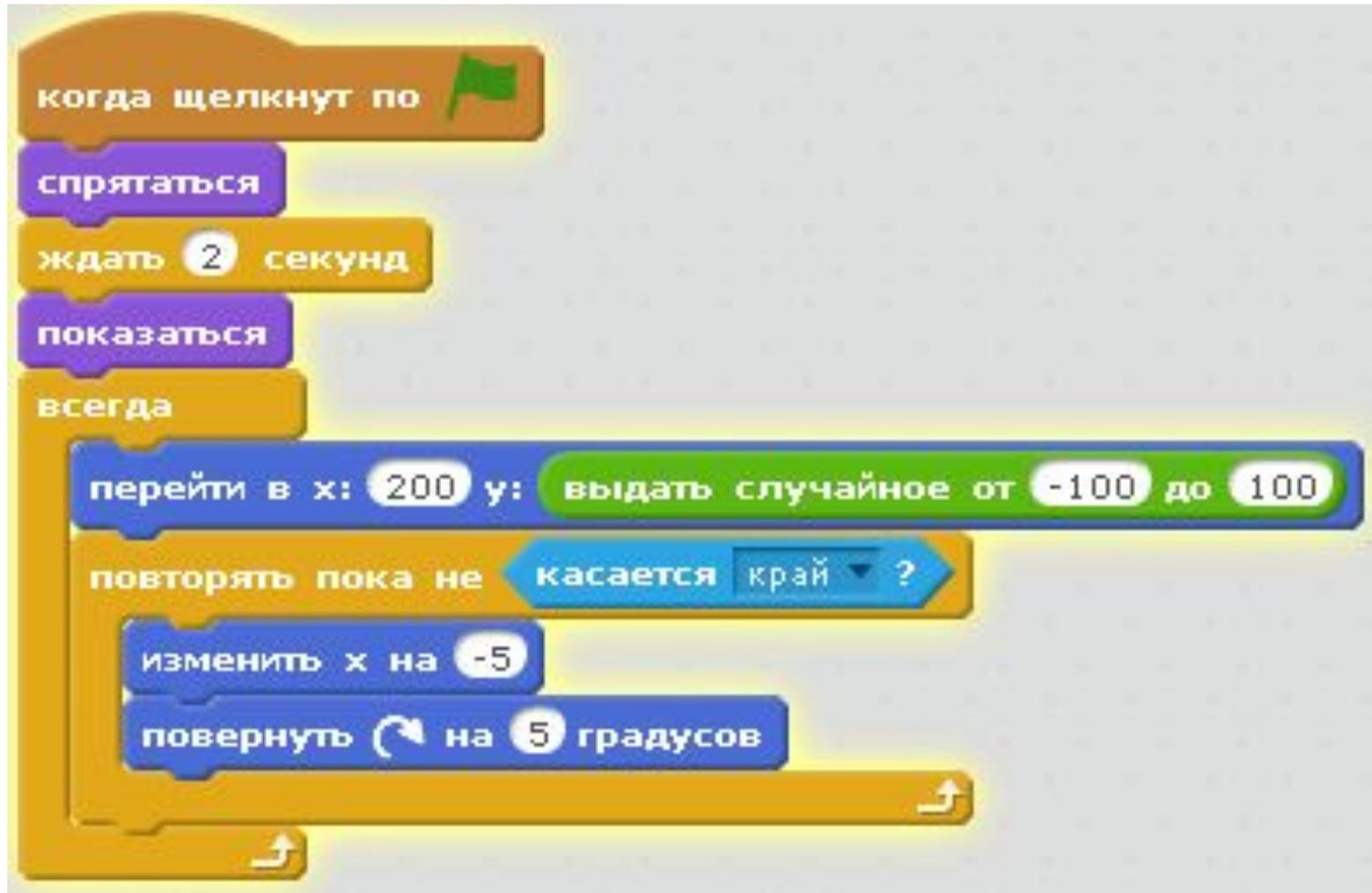
The image shows a Scratch script for moving an asteroid. It starts with a 'when green flag clicked' event block. The script then performs the following actions in order: 'hide', 'wait 1 seconds', 'show', and a 'loop' block. Inside the loop, it goes to the coordinates x: 200 and y: random from -100 to 100. It then enters a 'repeat until touches edge?' loop. Inside this loop, it changes the x coordinate by -5 and turns 5 degrees. The script ends with an upward arrow indicating the loop continues.

САМОСТОЯТЕЛЬНАЯ РАБОТА

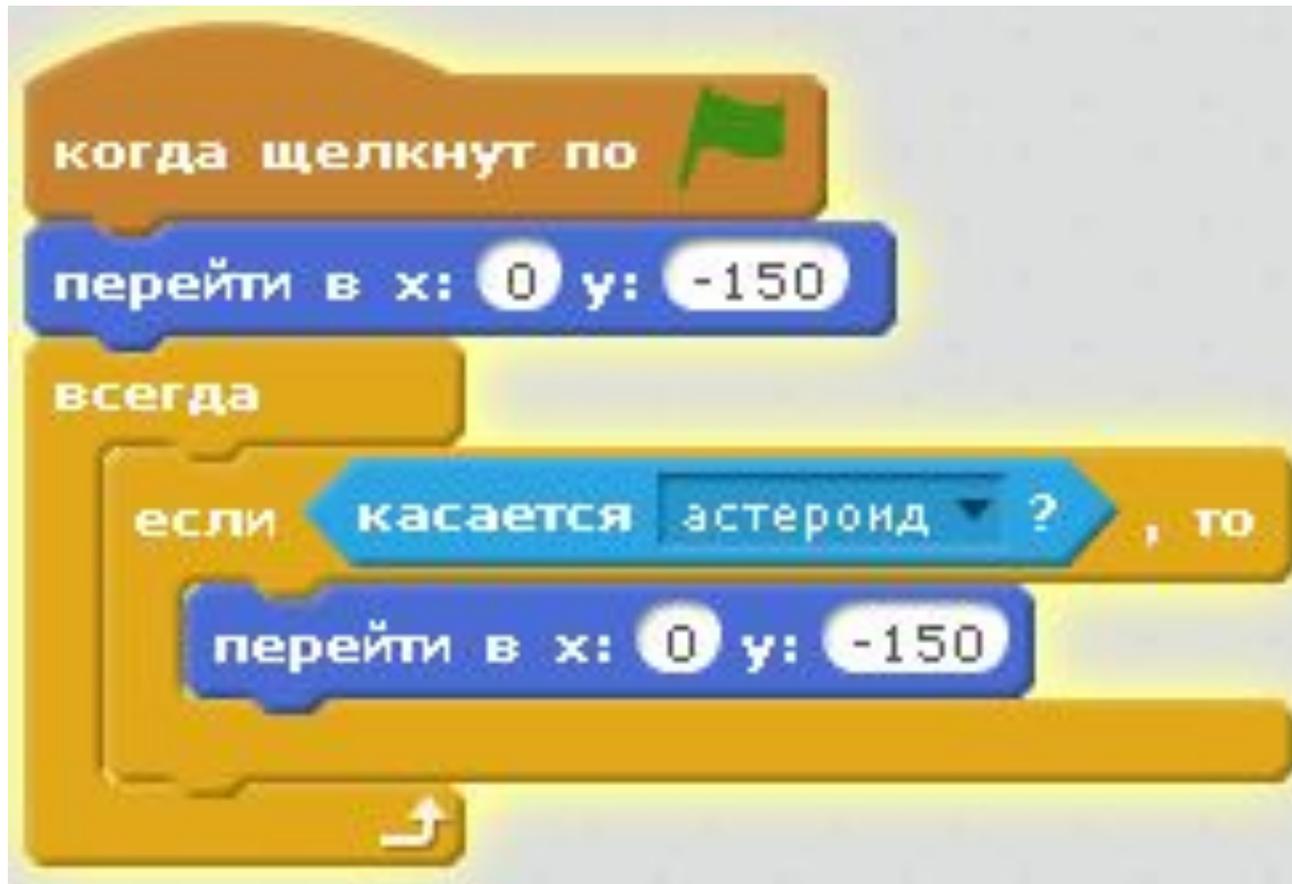
Доработайте скрипт для
третьего астероида
самостоятельно.



Скрипт для третьего астероида.



Ракете нельзя сталкиваться с астероидами, в случае касания астероида ракета возвращается в точку старта.



Астероидов у нас три, поэтому используем оператор ИЛИ



Запрограммируем ситуацию выигрыша, что бы выиграть- ракете нужно одновременно коснуться базы зарядки и батарейки, используем оператор И.



В случае победы Ракета передает сообщение “Победа”.

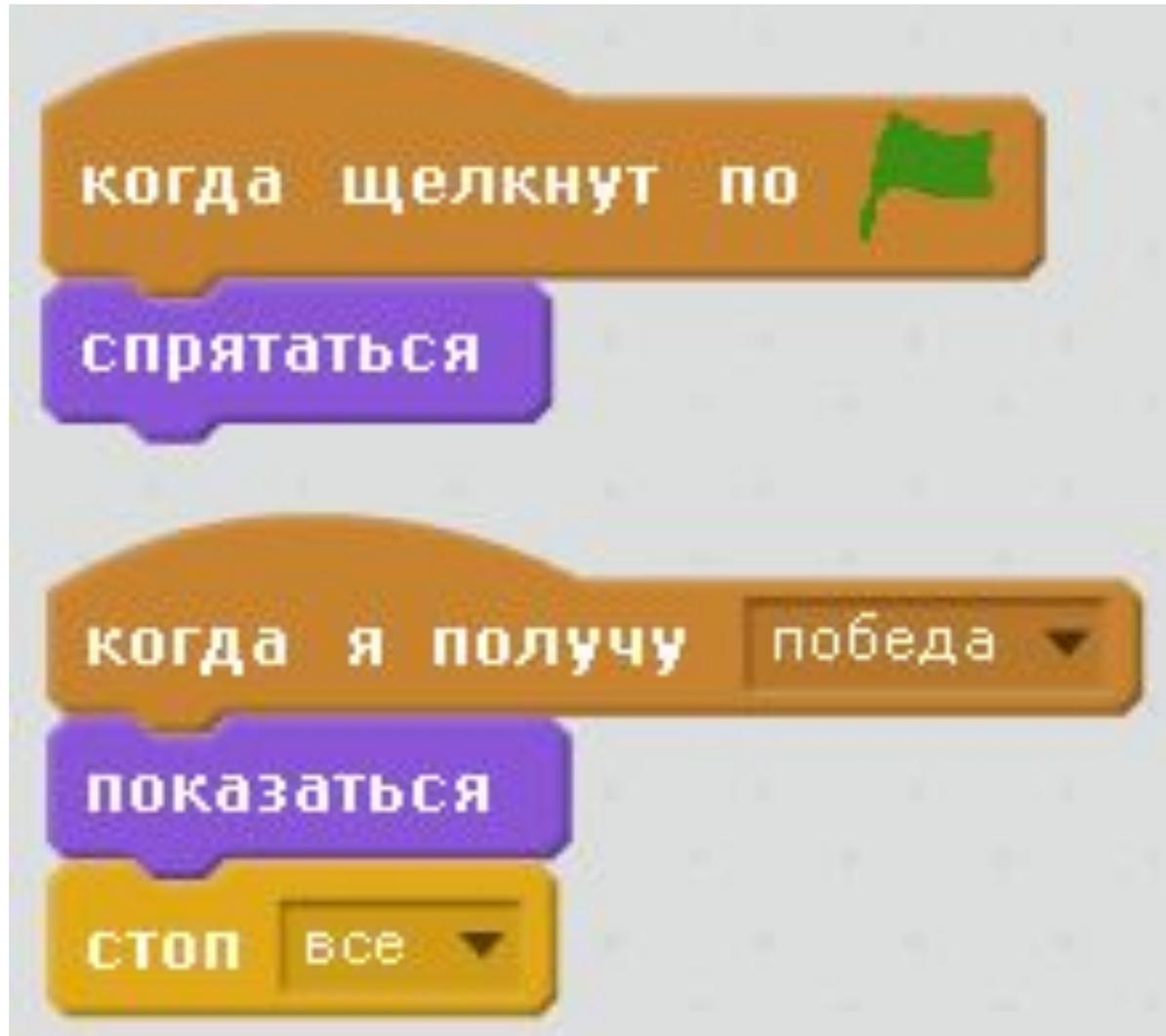
Составим скрипт для батарейки



Создадим спрайт победы



Составим скрипт для победы.



ЧТО ВЫ СЕГОДНЯ УЗНАЛИ

Логические операторы

1. И
2. ИЛИ
3. НЕ



ВОПРОСЫ ДЛЯ ПРОВЕРКИ

Приведите примеры:

- где в жизни применяют операторы (союзы) И/ИЛИ/НЕ,
- где можно применить в играх/программах.

