

# 1 УРОК В SCRATCH

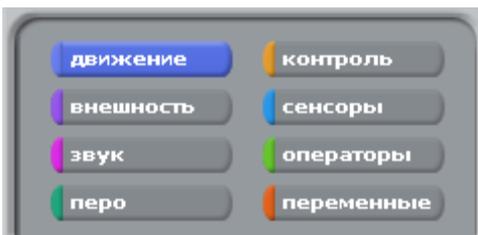
Обновление	Просмотров	Рейтинг
24 апреля 2015	59	• • 0

## Интерфейс Scratch и основы работы в нем

Будем знакомиться с интерфейсом Scratch постепенно, разбирать работу лишь того, что нам потребуется на текущий момент.

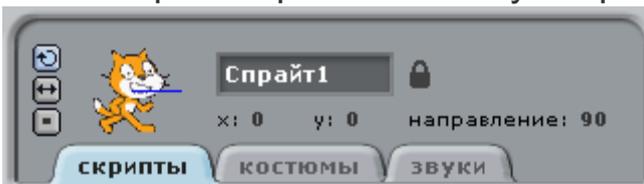
После того, как программа запустилась, перед нами появляется окно, в котором можно выделить три части (три столбца). В свою очередь каждый столбец состоит как бы из двух ячеек: верхней и нижней. Все ячейки разные и каждая из них предназначена для своей конкретной цели.

Верхняя ячейка в первом столбце содержит восемь кнопок, которые называются **движение**, **контроль**, **внешность**, **сенсоры**, **звук**, **операторы**, **перо**, **переменные**. При включении одной кнопки все остальные выключаются. Включенная кнопка вся окрашивается в соответствующий ей цвет. При этом в нижней ячейке первого столбца появляются команды, связанные с включенной кнопкой. У всех кнопок разные привязанные к ним команды. Попробуйте нажимать кнопки и посмотрите, как меняются команды в нижней ячейке.



На этом уроке нас будут интересовать только команды, связанные с кнопками **движение** и **контроль**.

Рассмотрим верхнюю ячейку второго столбца.



Здесь показаны свойства объекта, которым мы будем управлять (писать программы для него). Сейчас — это кот. Его имя написано в поле — Спрайт1. Странное имя для кота, не так ли? Спрайтами в

Scratch называются все объекты по умолчанию, отличаются лишь их номера. Обычно предполагается, что мы сами их будем переименовывать и называть более адекватными именами. Например, для кота лучше в поле имени вписать кот или Вася или что-нибудь еще. Сделайте это.

Ниже поля имени отображены три свойства объекта (в данном случае, кота) — это его положение (координаты  $x$  и  $y$ ) и направление. Если вы уже изучили работу в среде KТurtle, то вспомните, что направление, когда черепашка «смотрит» вверх равно  $0$  градусам, когда точно направо — равно  $90$  градусам, а полная окружность равна  $360$ -ти градусам. Здесь то же самое. Кот смотрит направо, поэтому его направление равно  $90$ .

Внизу мы видим три кнопки-вкладки — **скрипты**, **костюмы** и **звук**.

От того какая из них нажата, зависит ячейка внизу. Если нажата кнопка **скрипты**, то нижняя ячейка второго столбца покажет программы (скрипты) для объекта, который отображен в верхней ячейке. Сейчас нажата кнопка **скрипты** и мы видим пустую ячейку внизу. Это значит, что для кота пока нет никакой программы. Мы составим ее чуть позже. Кнопки **костюмы** и **звук** позволяют настраивать и менять соответственно внешний вид объекта и издаваемые им звуки. Оставьте включенной кнопку **скрипты**.

Перейдем к третьему столбцу и опишем в нем только верхнюю ячейку, где видим белое поле и кота на нем. Белое поле — это холст. Многие действия, которые задает программист на вкладке **скрипты**, объект выполняет именно на холсте. Так, если мы запрограммируем ходьбу кота, то он будет перемещаться по холсту. Если зажать левую кнопку мыши над котом, а затем не отпуская ее перемещать мышь, то кот будет перемещаться. Таким образом, мы можем менять его положение на холсте. Переставьте кота в другое место и посмотрите в ячейку свойств объекта (второй столбец). Заметьте, что координаты  $x$  и  $y$  изменили свои значения.

Над холстом находятся две вот такие кнопки:



Когда для объектов (кота и других) будут составлены какие-нибудь скрипты (программы) на вкладке **скрипты**, то чтобы объекты начали их выполнять, надо нажать зеленый флажок. Чтобы остановить выполнение, надо нажать красный круг. Не забывайте останавливать свои скрипты!

Над флагом и кругом мы видим еще три кнопки.



Они предназначены для изменения всего окна приложения Scratch. Сейчас у нас включена вторая кнопка, поэтому окно Scratch примерно поровну разделено на три столбца. Однако если мы составляем какую-нибудь очень сложную программу и у нас получается очень большой скрипт, то лучше нажать первую кнопку. При этом второй столбец займет существенную часть окна. Третья кнопка предназначена исключительно для просмотра созданной программы-анимации. Если ее нажать, то холст занимает центр экрана, и доступными остаются только три кнопки: запуск программы (зеленый флажок), остановка программы (красный круг) и выход из режима презентации (стрелка), осуществляющий возврат к предыдущему окну. Попробуйте переключаться между режимами работы.

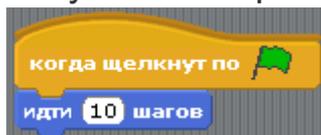
## Первая программа

Мы уже рассмотрели интерфейс достаточно, чтобы написать первую программу для кота.

Сейчас включена кнопка **движение** и видны команды, отвечающие за перемещение объекта. Если перетащить мышью любую из этих команд в ячейку **скрипты**, то она станет командой для кота.

Например, иди 10 шагов заставит животное переместиться на 10 точек экрана. После того как команда перемещена в скрипты, можно посмотреть, как она работает, совершив двойной клик по ней. Кот шагнет на холсте. Задайте команду иди 10 шагов для объекта и проверьте, как она работает.

У нас есть программа для кота, состоящая из одной команды. По идее при нажатии на кнопку запуска (зеленый флаг) она должна срабатывать. Но на самом деле этого не произойдет. Чтобы программа запускалась при нажатии флажка, надо в ее начало поместить специальную команду, которая связана с кнопкой контроль. На ней написано когда щелкнут по и изображен зеленый флажок. Если взять эту команду, перетащить в поле скриптов и соединить с командой иди 10 шагов, то программа станет запускаться при нажатии на кнопку запуска. Сделайте это.



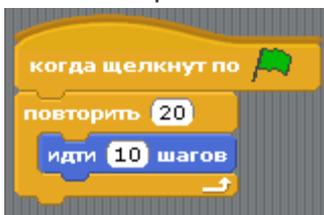
Обратите внимание на то, как соединяются между собой команды: как блоки в конструкторе Лего. В каких-то случаях это может служить своего рода подсказкой. Какие-то команды можно соединять между собой, а какие-то не соединяются, и, значит, мы составляем не очень правильную программу.

Чтобы разорвать блоки, надо потянуть за нижнюю команду, чтобы переместить весь блок — за самую верхнюю. Попробуйте и затем верните все на место.

Сделаем некоторые выводы. Наш код даже не шагает, а просто прыгает на 10 точек вперед и останавливается. Конец программы. Хорошо бы было, чтобы он все-таки шагнул и проделывал путь несколько больше. Самое первое, что может прийти в голову, — это увеличить количество шагов, например, до 100. Для этого надо в команде иди 10 шагов щелкнуть по числу 10 и вписать в поле число 100. Переместите кота ближе к левой границе холста, поменяйте число шагов и запустите программу.

Теперь кот шагает больше. Но разве это похоже на шаги? Он просто скачет на большее расстояние. Поэтому вернем число 10 и подумаем над другим решением.

В командах контроля есть такая команда как повторить 10. Это цикл, который прокручивает то, что в него вложено такое количество раз, какое число написано в его поле. По умолчанию — это 10. Если команду иди 10 шагов поместить внутрь этого цикла, то он будет ее «крутить». Поместите кота ближе к левой границе холста и составьте вот такую программу для него. Посмотрите, как теперь наш объект будет двигаться. Согласитесь, что у нас получилась настоящая анимация объекта.



Продолжим улучшать и видоизменять нашу первую программу. Допустим, мы не знаем, сколько раз надо повторять шаги. Мы хотим, чтобы объект двигался всегда, пока мы не нажмем кнопку остановки (красный круг). Для этого надо поступить так: вытащить команду иди 10 шагов из циклаповтори ..., убрать цикл повтори ... (перетащить его в левый столбец), взять команду всегда и собрать такую программу:

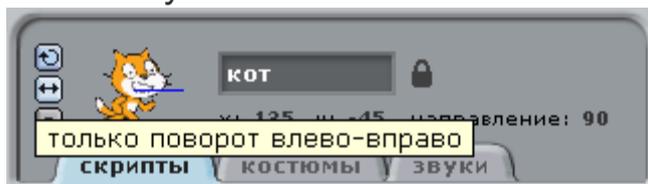


Теперь программа должна работать вечно, пока не будет остановлена. Так и происходит. Если нажать кнопку запуска, то зеленый флажок будет гореть до тех пор, пока не будет нажат красный. Это значит, что программа не может сама остановиться.

Но посмотрите, что происходит с котом. Он останавливается, достигнув границы холста. Программа работает, а кот стоит. Тут что-то не так. Останавливаем программу! Возвращаем кота на место. С кнопкой **движение** связана такая команда как если край, оттолкнуться, оттолкнуться. Если ее поместить в цикл всегда, то объект уже не остановится достигнув края, а оттолкнется от него и пойдет в другую сторону. Составьте вот такой скрипт и посмотрите, как он работает.



Скрипт работает замечательно, но кот не совсем нормален при движении справа налево. Он идет вниз головой. Для исправления этого недочета остановим программу и снова обратим свой взгляд на ячейку свойств объекта.



В левой части этой ячейки есть три маленькие кнопки: со скругленной стрелкой, двуголовной стрелкой и квадратной точкой. По умолчанию нажата первая кнопка и поэтому объект при столкновении поворачивается так, как мы наблюдали. Если нажать вторую кнопку, то он будет поворачиваться так, как нам надо в данный момент (слева направо). Третья кнопка вообще запрещает какие-либо повороты. Нажмите вторую кнопку и посмотрите, как ходит кот. Не забудьте после этого остановить программу. Посмотрите на изображение кота в ячейке свойств. У него есть какой-то синий отрезок. Зажав на нем мышью, его можно поворачивать. При этом меняется значение направления. Это означает, что объект при запуске программы будет двигаться в установленном таким образом направлении. Поэкспериментируйте с выбором направления движения кота.

## Самостоятельная работа

Составьте вот такую программу, испытайте (протестируйте) ее и попытайтесь объяснить, как она работает.



## Задания по теме "Знакомство с интерфейсом"

Задания к данному разделу предназначены для того, чтобы мы смогли научиться создавать и сохранять проекты, открывать созданные ранее проекты, то есть научиться основам работы в Scratch.

Успехов Вам!

P.S.: Самое интересное еще впереди! До встречи в следующей теме!

**Задание 1.** Редактирование проекта.

1. Запустите программу.
2. Откройте проект **Celebration**, выбрав пункт меню **File – Открыть проект-Примеры-Greetings- Celebration**
  1. Воспроизведите скрипт (программу), нажав на зеленый флажок.
  2. В нижней ячейке правой части рабочего поля выберите объект **Student**, перейдите на вкладку **Скрипты** и измените значение в поле **Ждать** на 0,5.
1. В нижней ячейке правой части рабочего поля выберите объект **screen1**, перейдите на вкладку **Скрипты** и измените значение в поле **Изменить** на Рыбий глаз.
1. Сохраните проект под именем **Proekt**.
1. **Файл с проектом отправьте преподавателю, выбрав кнопку Загрузить внизу страницы.**

**Задание 2.**

1. Запустите программу.
2. Создайте новый проект, выбрав пункт меню **File – Новый**.
3. Выберите объект **CAT3** из коллекции **Animals**.

4. Составьте для объекта следующий скрипт, перетаскивая нужные блоки из левой части рабочего поля в центральную:
5. Воспроизведите скрипт, нажав на зеленый флажок.
6. Сохраните проект под именем **Proekt1**.
7. **Файл с проектом отправьте преподавателю.**

### Вопросы к самостоятельной работе:

1. Сколько циклов в программе? Назовите их.
2. Какой цикл является вложенным, а какой внешним?
3. Какие команды содержит цикл **повтори ...**?
4. Какие команды содержит цикл **всегда**?
5. Сколько всего шагов делает кот, прежде чем останавливается на 3 секунды

6. Составьте такую программу, испытайте (протестируйте) ее и попытайтесь объяснить, как она работает.

### Вопросы к самостоятельной работе:

1. Сколько циклов в программе? Назовите их.
2. Какой цикл является вложенным, а какой внешним?
3. Какие команды содержит цикл **повтори ...**?
4. Какие команды содержит цикл **всегда**?
5. Сколько всего шагов делает кот, прежде чем останавливается на 3 секунды?