

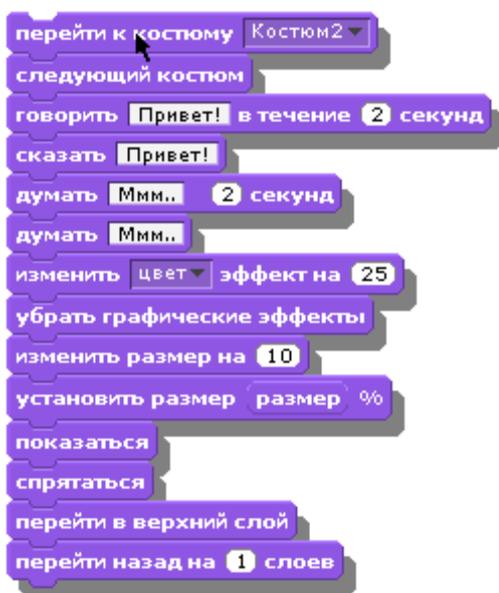
12 УРОК В SCRATCH

Обновление	Просмотров	Рейтинг
5 февраля 2015	7	0

Кирпичики контроля, движения и внешнего вида

Мы уже многому научились, судя по [работам](#). Теперь пришло время немного обобщить, привести в порядок. Этим мы займемся на этом и следующем уроке, а заодно и усложним задания

Внешний вид объекта (фиолетовый ящик)



Все команды однозначны. Чаще всего употребляются команды «Изменить()» и «Убрать графические эффекты». Переходы между слоями легче всего пояснить на примере 2-х одинаковых спрайтов котенка. У одного из них покрасим глаза в голубой цвет. Если мы теперь попросим одного из котят менять слой, то получится простая мультипликация – котенок будет менять цвет глаз.

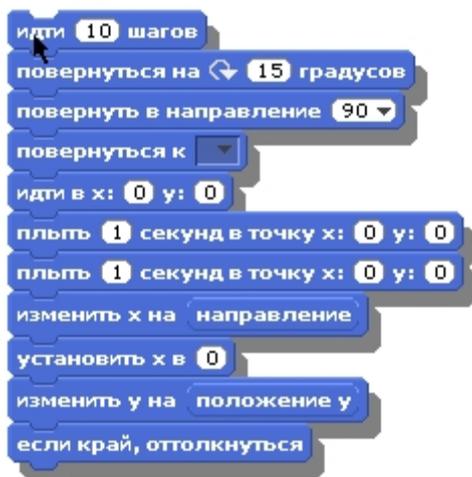


Здесь приведен самый простой пример, но мы можем увеличить число объектов.

Команда	Назначение
Перейти к костюму()	Перейти к костюму (можно выбрать костюм героя)
Выбрать костюм()	По умолчанию объект одет в костюм кошки, но мы можем выбрать и другие костюмы из библиотеки, пользуясь командой Импорт.
Изменить значение костюма	Изменить значение костюма на одну или несколько единиц. Имеет смысл только тогда, когда у нас несколько «импортных» костюмов.
Сказать	Сказать фразу, которую мы можем записать в окошечке команды. К сожалению, пока можно использовать только латинские буквы.
Думать	<p>Думать – отличается от «говорить» только графическим оформлением фразы</p> 
Убрать графические эффекты	Очистить все графические эффекты. Если мы производили над объектом видоизменение, то в результате этой команды все они отменяются
Изменить()	Изменить (цвет, округлость, яркость) () – видоизменить объект по одному из перечисленных параметров на указанную величину.
Показаться	Показать объект – он становится видимым
Спрятаться	Спрятать объект - он становится невидимым
Перейти в верхний слой	Объект перемещается в первый слой
Перейти назад на () слоев	объект уходит на несколько слоев внутрь
Значение костюма	Возвращает значение костюма, который в данный момент одет на нашем исполнителе.

Синий ящик - Движение

В синем ящике сложены строительные блоки, управляющие перемещением объектов по экрану.



Центральная точка экрана соответствует координатам $X=0$, $Y=0$. Вправо от нулевой точки X возрастает и это область положительных значений. Влево – область отрицательных значений.

Научимся использовать блоки, которые не являются командами, а возвращают значения – положение x , y и направление. Простой пример:

Мы хотим, чтобы котенок последовательно переходил из нижнего правого угла в левый верхний угол.



Команда	Назначение
идти	пройти вперед указанное число шагов. Например: Идти(10) – сделать десять шагов назад Идти (-30) – сделать 30 шагов назад
Повернуться на	повернуться - стрелочка указывает по часовой или против часов стрелки выполняется поворот. Например: повернуться на 45 градусов (стрелочка указывает по или против часовой стрелки)
Повернуться в направление()	Повернуться в указанном направлении. Например: повернуться в направление(180)
Повернуться к	Повернуться в направлении другого существа или координат мышки. После команды всегда существует перечень объектов, которые в данный момент присутствуют в системе и на которые можно реагировать. В самом простом случае, когда других объектов нет, предлагается повернуться в сторону, где находится указатель мышки.
Изменить X на()	Изменить положение по оси X на указанное число шагов. Например: Изменить X (-10) – объект смещается влево на 10 шагов
Установить x в ()	Установить объект положение по оси x. Например: Установить x в (50)
Идти в x () y()	Переместиться в точку с указанными координатами Например: Идти в x(70) y(-50) В начале проекта всегда стоит установить персонаж в конкретную точку, чтобы точно знать, где он находится.
Плыть () сек в точку	Плавное перемещение в точку с указанными координатами за указанное время. На перемещение затрачивается время указанное в секундах. Например: Плыть(20) сек в точку x(90) y(90) Объект медленно (в течение 20 секунд) будет переползать в указанную точку.
Идти в ()	Перейти в точку, где расположен указатель мыши или другая фигура.
Если край, оттолкнуться	Если попадаешь в край экрана, то отразись от него. Это очень полезно, если Вы не хотите потерять своего героя

Следующие несколько блоков нельзя использовать в качестве отдельных строительных блоков, но они могут быть вставлены внутрь строительных кирпичиков в качестве входных параметров

Положение x	Возвращает значение по оси X Например: Установить y (положение x)
Положение Y	Возвращает значение по оси y Например: Установить x (положение y)
Направление	Возвращает направление

Кирпичики контроля (желтый ящик)

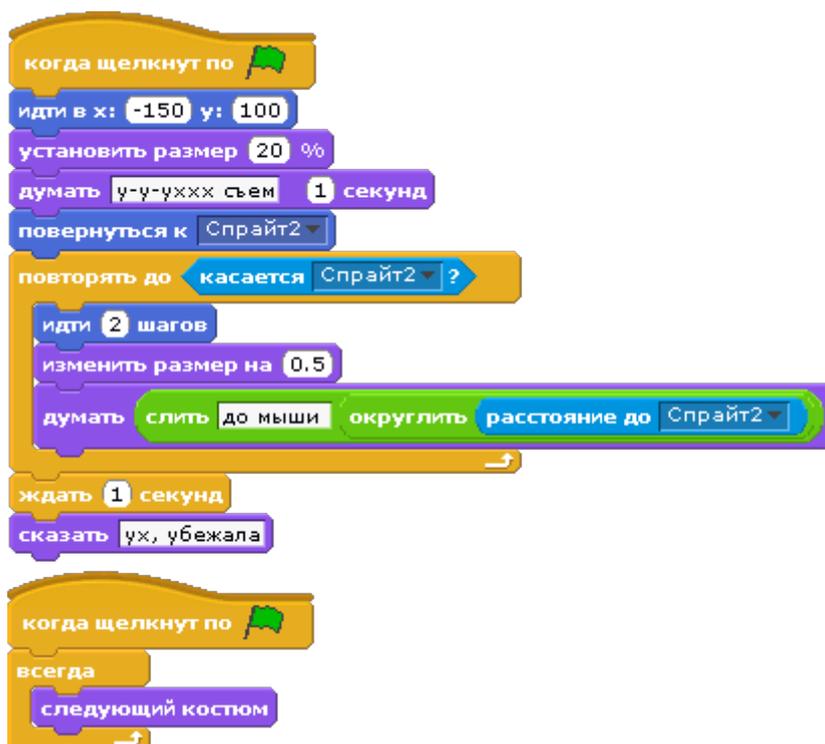
Кирпичики контроля позволяют собирать блоки команд и управлять последовательностью исполнения скриптов. Они определяют то, при каких условиях совершается то или иное действие. Они позволяют спрайтам передавать, принимать и реагировать на управляющие сообщения.

Команда	Назначение
Когда щелкнут по флажку	Запускает скрипт в ответ на щелчок по зеленому флажку. Это обычный способ запуска всей модели.
Когда нажата () клавиша	Запускает выполнение блока команд в ответ на нажатие выбранной клавиши. Позволяет передать управление на клавиатуру. Например: когда нажата стрелка вверх, повернуть вверх
Ждать ()	Команда ожидания. Параметр () указывает сколько секунд следует ждать.
Всегда()	Блок команд, заключенных внутри конструкции будет выполняться постоянно.
Если <>	Условие, при выполнении которого должны выполняться команды, заключенные внутри конструкции. Если не выполняется, то никаких действий
Если <> или	Условие, при выполнении которого должны выполняться команды, заключенные внутри конструкции если . Если не выполняется, то нужно перейти к действиям внутри блока или. 
Повторить()	Повторение. Параметр указывает, сколько раз нужно повторить блоки команд, заключенные внутри блока повторить() Например, повторить 20 (идти 50 повернуть 45)
Передать ()	Передать сообщение. Переданное сообщение может запускать активность другого исполнителя. Работает в сочетании с командой «Когда я получил ()» (Про блоки Передать и Когда получил у нас отдельный урок)
Когда получил ()	Запускает выполнение блока команд в ответ на полученное сообщение. Работает в сочетании с «передать ()»
Всегда если ()	Выполняется до тех пор, пока условие () верно
Ждать до ()	Ждать, пока не выполнится условие
Остановить скрипт	Остановить выполнение программы для данного исполнителя
Остановить всё	Остановить выполнение всех программ

Задание:

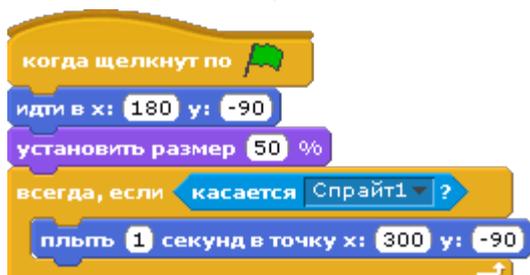
- 1) Слоненок издали подходит к мячику. Как-будто ударяет по нему и мяч улетает. Слоненок радостный кричит: "Урааа!". Подсчитать количество шагов, сделанных слоненком до мяча. Применить эффекты приближения и удаления.
- 2) Придумайте свою подобную историю. *Подсказка: пусть это задание будет подготовкой к выполнению итоговой работы.*
- 3) Разместите проекты на сайте Скретч, в комментариях напишите ссылки на проекты.

Для примера



Это скрипт летучей мыши. Особенности:

1. Первые 4 команды (*Идти ... Повернуться*) устанавливают начальное положение и состояние мыши.
2. Далее, пока летучая мышь не коснется обычной мыши, она будет двигаться к ней, постепенно увеличиваясь в размере. Тем самым создается эффект приближения.
3. Второй скрипт - Всегда следующий костюм - служит для того, чтобы летучая мышь махала крыльями.



Это скрипт обычной мыши. Особенности:

1. Первые две команды описывают начальное положение и состояние объекта.
2. Используется форма "*Всегда, если...*" для того, чтобы скрипт постоянно следил за тем, касаются ли спрайты. Как только они коснутся, мышь убежит.

Использовались материалы с [Паутины соучастия](#).

Комментарии