

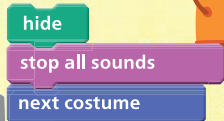
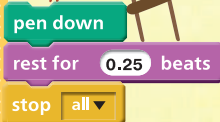
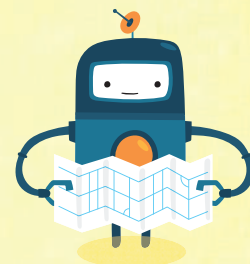
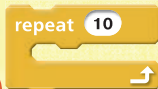
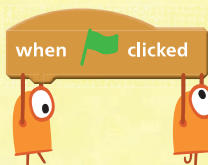
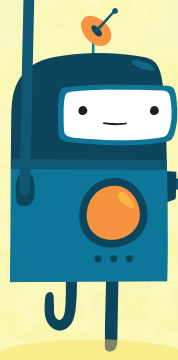
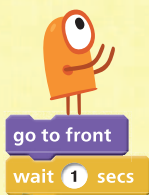


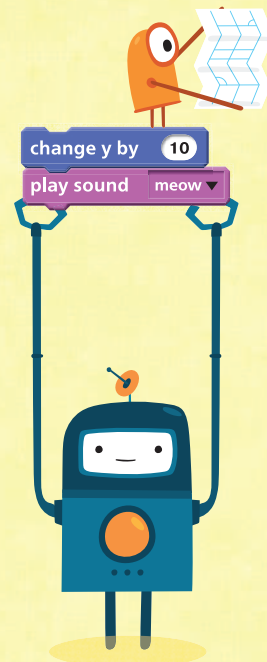
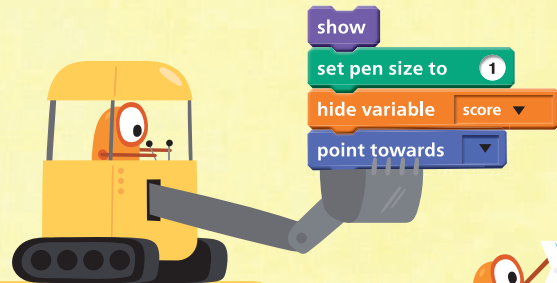
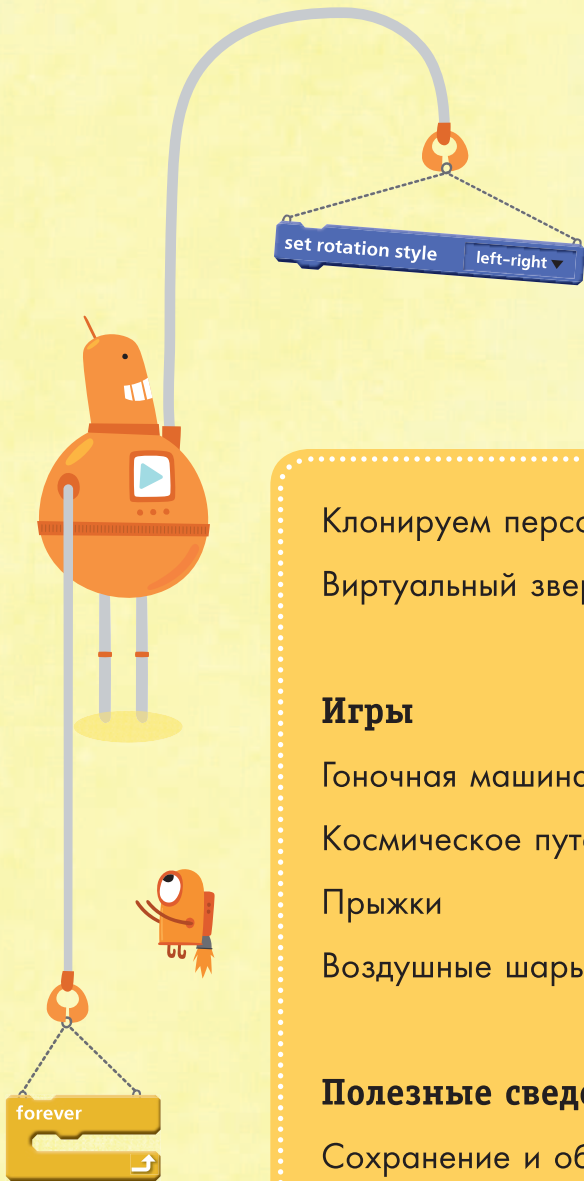
Программирование
для детей
на языке

SCRATCH

Содержание

Что такое программирование?	4
Начинаем с языка Scratch	6
Первые проекты	
Кошки-мышки	8
Танцующие спрайты	12
Организуем ансамбль	14
Программируем страшилки	18
Черчение	22
Жили-были спрайты	26
Рисование спрайтов	32
Угадай число	36
Бита и мяч	40





Клонируем персонажа 44

Виртуальный зверек 46

Игры

Гоночная машина 54

Космическое путешествие 60

Прыжки 66

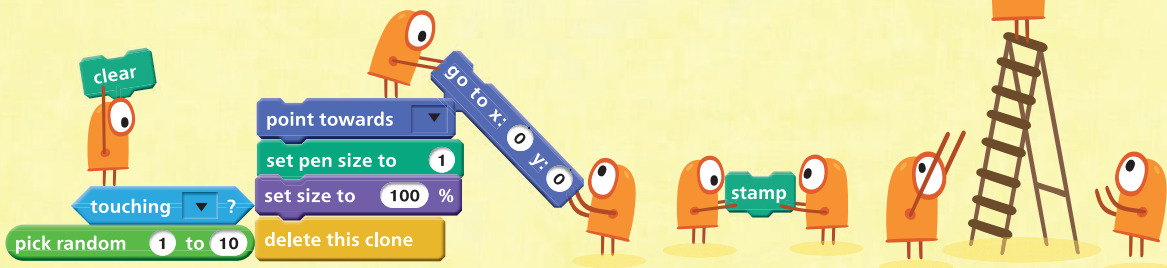
Воздушные шары 72

Полезные сведения

Сохранение и обмен проектами 80

Путеводитель по меню 82

Глоссарий 92



ЧТО ТАКОЕ ПРОГРАММИРОВАНИЕ?

Программирование — это написание инструкций для компьютера, а окончательная версия таких инструкций называется программой. Когда вы научитесь программировать, то сможете создавать свои собственные программы.

Быть понятным

Для того чтобы программа работала, она должна быть написана понятным компьютеру образом. Это означает, что все инструкции должны быть разбиты на простые и ясные шаги, не говоря уже о том, что они должны быть написаны на определенном языке программирования.

(COMMAND > POUR MILK)



Опс, я забыл сказать «Стоп»!



ВНИМАНИЕ!

Компьютер четко следует инструкциям. Он не может сам мыслить, так что все действия должны быть ясно описаны и ничего не должно быть оставлено без внимания.

Язык программирования

Язык программирования во многом схож с обычным языком человеческого общения. Только он имеет ограниченный набор слов и точные правила, как писать этими словами программы.

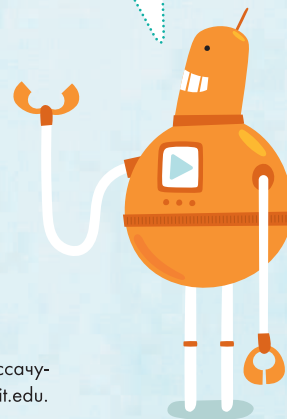
Существует множество языков программирования, адаптированных под разные типы задач. Для многих юных читателей первым языком программирования может стать язык Scratch (читается «Скретч»), специально созданный для учебных целей.

Язык Scratch прекрасно подходит для создания игр и анимации, а также позволяет получить основные навыки написания программ.



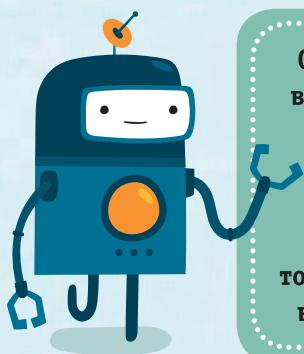
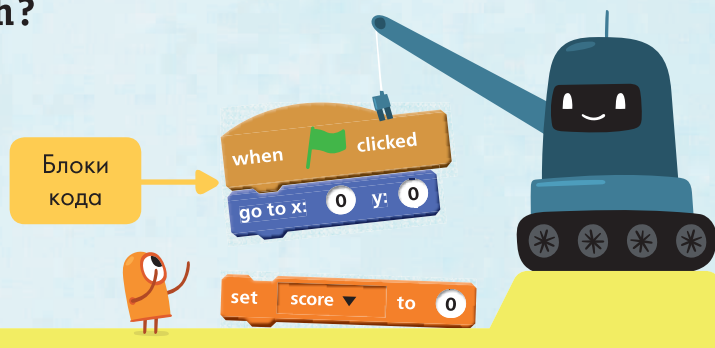
SCRATCH

Язык Scratch разработан Lifelong Kindergarten Group при MIT (Массачусетский технологический институт) Media Lab. См. <http://scratch.mit.edu>.



Почему мы выбрали Scratch?

Язык Scratch разработан таким образом, чтобы писать программы на нем было не сложнее, чем играть в конструктор. Вставляя друг в друга замочки готовых к использованию блоков кода, вы по сути, уже пишете программу.



Обратите внимание на такие вот текстовые вставки. В этих вставках на зеленом фоне объясняются различные **КЛЮЧЕВЫЕ ПОНЯТИЯ**. Если текст на синем фоне, то это **СОВЕТ** по использованию возможностей языка Scratch.

Немного об этой книге

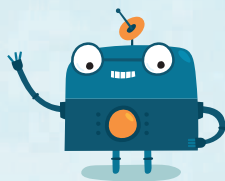
Эта книга покажет вам, как использовать большую часть инструментов языка Scratch для создания анимированных историй и игр. Также вы получите множество советов, как создавать свои собственные программы. Все примеры разбиты на короткие шаги, которым будет легко следовать при обучении.

Приступаем

Чтобы начать использовать язык Scratch, проще всего посетить веб-сайт этого обучающего проекта. Все, что вам надо, это компьютер — тот, который с клавиатурой (планшет вам не подойдет) — и выход в Интернет.

Посетите ресурс <https://scratch.mit.edu>.

Для использования программы необходимо создать учетную запись.



Scratch есть и на русском языке, но лучше привыкать работать с английским интерфейсом, ведь более серьезные программы для программирования имеют только английский интерфейс.



Если вы хотите использовать языковую среду Scratch в режиме оффлайн — т.е. без связи с Интернетом, то можете скопировать установочную программу на свой компьютер. Также с программой можно работать без установки — прямо через Интернет.



НАЧИНАЕМ С ЯЗЫКА SCRATCH

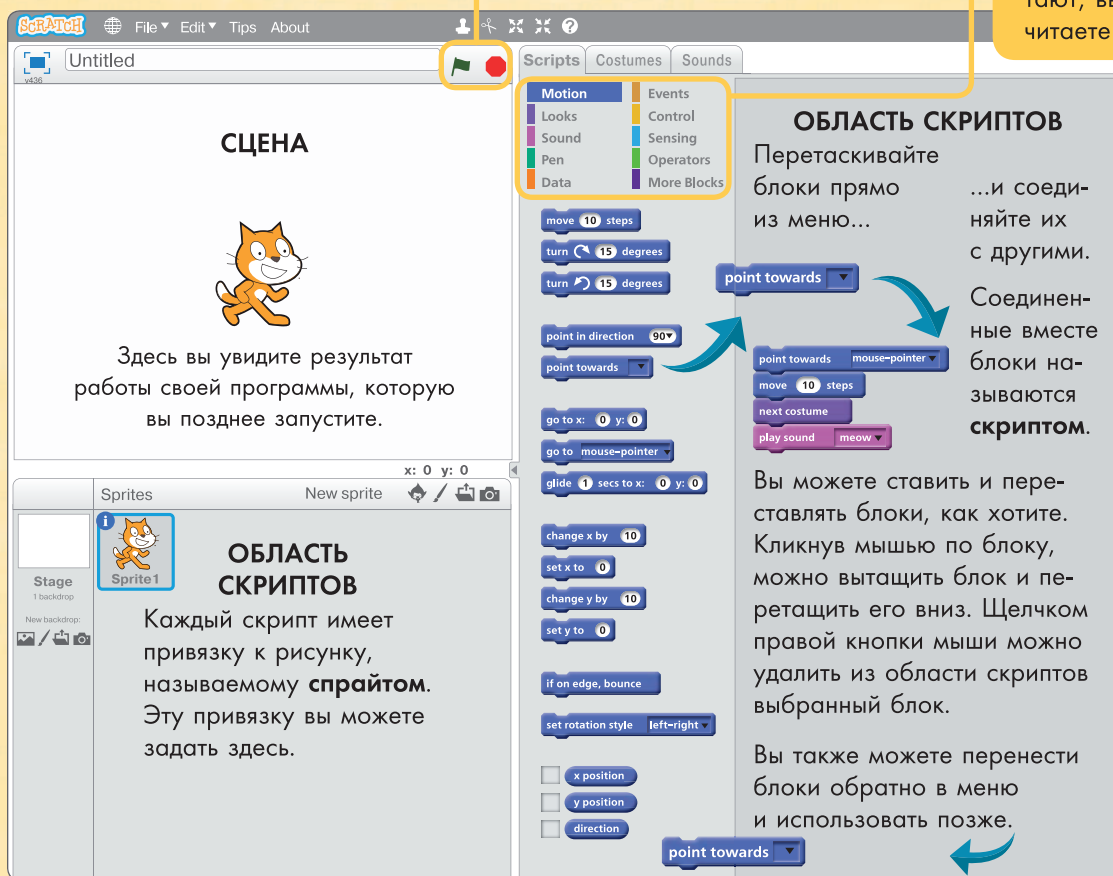
Когда вы запустите на компьютере Scratch,
то увидите такую картинку.

Для начала работы кликните на синей
плашке по надписи Create («Создать»).



Эти кнопки — **зеленый флаг** и **красный шарик** — используются для запуска и остановки программы.

Это **блоки меню**, а о том, как они работают, вы прочитаете ниже.



Блоки меню

В каждом из **пунктов меню** вы увидите большое разнообразие различных по форме и цвету **блоков**. Например...

- **Motion** («Движение») — блоки меню (синие), заставляющие спрайты двигаться.
- **Looks** («Внешний вид») — блоки меню (фиолетовые), изменяющие внешний вид спрайтов.
- **Control** («Управление») — блоки меню (золотистые), контролирующие исполнение скриптов.

Кликните мышью по **пункту меню** и вы увидите доступные вам **блоки** (полный список этих блоков вы также можете увидеть, начиная со страницы 82).

Motion	Events
Looks	Control
Sound	Sensing
Pen	Operators
Data	More Blocks

Вот все десять **пунктов меню**.

Первые шаги

1 Для начала перетащите первые два блока (например, из меню **Motion** («Движение»)) в область скриптов и научите кота ходить...

Затем кликните мышью по пункту меню **Sound** («Звук») и добавьте блок **Play** («Проигрывать звук»).

2 Чтобы **запустить** скрипт, просто кликните по нему мышью. Для эксперимента кликните по скрипту несколько раз и посмотрите, что будет.

Код запущенного скрипта подсветится, а кот начнет перемещаться по экрану и мяукать. (Если кот уйдет слишком далеко вбок, вы можете перетащить его обратно мышью.)

3 Но кот не будет выглядеть так, словно он ходит. Для этого его лапы должны двигаться...

Кликните мышью по пункту меню **Looks** («Внешний вид») и добавьте блок **Next costume** («Следующий костюм»). Такое переключение на следующую картинку, или «костюм» того же спрайта создаст иллюзию того, что кот двигает лапами. Кликните по этому скрипту несколько раз.

4 Кот действительно будет двигать лапами, но только тогда, когда вы кликаете по скрипту мышью. Чтобы он делал это без вашего участия, надо обратиться к меню **Control** («Управление») и выбрать там блок **Repeat** («Повторить»). Этот блок сделает так, что все инструкции внутри него будут повторяться.



Выберите из выпадающего меню пункт «указатель мыши».



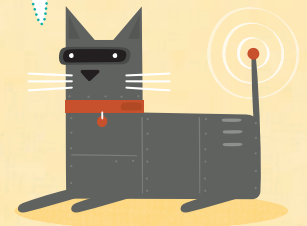
Выберите из выпадающего меню пункт «мяу».



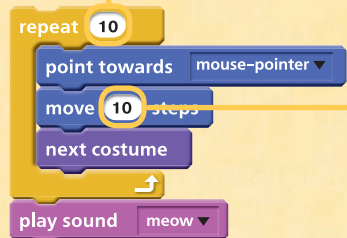
КЛЮЧЕВЫЕ СЛОВА

Слова инструкций, такие как **ДВИЖЕНИЕ** или **ПРОИГРЫВАНИЕ**, иногда называют **КЛЮЧЕВЫМИ СЛОВАМИ**, поскольку они имеют в языке программирования ясное и точное значение.

Поздравляем, только что вы написали свою первую программу!



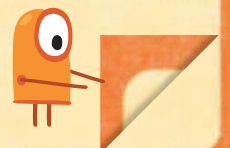
В этом окошечке вы можете указать цифрой, сколько раз будет повторяться действие.



ЦИКЛЫ

Циклы очень широко используются в программировании, поскольку они позволяют сделать программу намного короче. С этими командами программы пишутся намного быстрее.

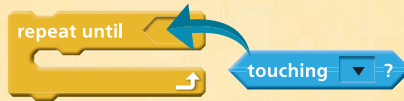
Проверните страницу и вы увидите, как превратится этот скрипт в простую игру «Кошки-мышки».



КОШКИ-МЫШКИ

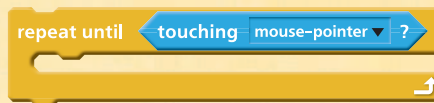
В этой простой игре надо не дать коту догнать мышку, удерживая ее в одном шаге перед ним. Если кот коснется указателя мыши, он скажет «Поймал!» и игра закончится.

1 Эта игра требует новой команды организации цикла: **repeat until** («Повторять, пока не...») из меню **Control** («Управление»).



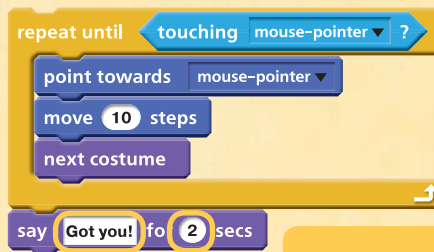
И не забудьте ромбовидный блок из меню **Sensing** («Сенсоры»).

2 Вставьте условие в соответствующее окошко команды повторения (блок повторения при этом увеличится и вместит окошко условия). Затем кликните по черному треугольнику и выберите из выпадающего меню слова «указатель мыши» (mouse-pointer).

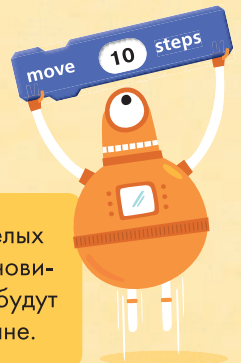


Этот цикл будет выполняться снова и снова до тех пор, пока кот не коснется указателя мыши.

3 Теперь возвратитесь к скрипту с предыдущей страницы и кликните мышью по первому блоку в цикле. Затем перетащите этот блок вместе со всеми присоединенными снизу блоками в свой новый цикл.



Кликните мышью на белых полях слов кота и установите, сколько секунд они будут высвечиваться на экране.



4 Закончите программу блоком **say** («Сказать») из меню **Looks** («Внешний вид»).

Тестирование скрипта

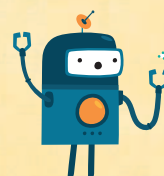
5 Кликните мышью по скрипту и поведите указателем мыши по экрану. Кот должен следовать за «мышкой» (указателем мыши), пока не поймает ее. Запустите скрипт несколько раз.

Если кот достигнет края сцены, он начнет мерцать (его изображение станет неустойчивым). Однако вы можете исправить эту ошибку, введя в начале цикла команду **bounce** («Отскакивать») из меню **Motion** («Движение»).



ВЕТВЛЕНИЯ

Такие условные конструкции ветвления, как **IF** (если) и **REPEAT UNTIL** (повторять, пока не...), позволяют сделать так, чтобы компьютер по-разному реагировал на выполнение условия (в данном случае, где находится кот). Именно поэтому эти команды называют и командами ветвления, и условными конструкциями.



Если ваша программа работает, возьмите с полочки пирожок.

6 Также вы можете упростить процедуру использования скрипта, добавив в него в самом начале блок **green flag** («зеленый флаг») из меню **Events** («События»).

Теперь вы можете запускать скрипт, кликнув мышью на **зеленом флаге**, который изображен на экране над **сценой**. (Там же вы найдете **красную кнопку**, которая останавливает все скрипты.)

Обратите внимание, что блоки имеют определенную форму, так что их можно совместить друг с другом далеко не как попало...

```

when green flag clicked
repeat until touching mouse-pointer ?
  if on edge, bounce
  point towards mouse-pointer
  move 10 steps
  next costume
say Got you! for 2 secs
  
```

Форма шапки, которую имеет стартовый блок, означает, что этот блок может быть поставлен только сверху.

А сюда встанут только блоки ромбовидной формы.

СИНТАКСИС

Набор правил, определяющий, как писать программный код, называется **СИНТАКСИСОМ**. Если вы совершаете синтаксическую ошибку, компьютер может запутаться. Однако в языке Scratch все устроено так, что вы не допустите ошибки. Блоки защелкнутся только в том случае, если они стоят правильно.



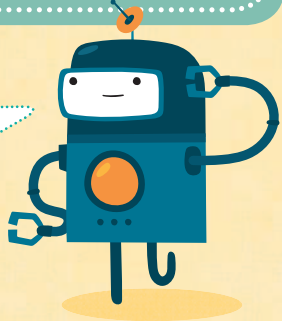
7 Чтобы сделать игру более справедливой, вы можете задать условия такими, что кот каждый раз будет начинать с середины **сцены**. Для этого надо добавить блок **go to x y** («перейти к x y») из меню **Motion** («Движение»).

Теперь вы можете задать положение кота координатами...

```

when green flag clicked
go to x: 0 y: 0
repeat until touching mouse-pointer ?
  if on edge, bounce
  point towards mouse-pointer
  move 10 steps
  next costume
say Got you! for 2 secs
  
```

Так что я всегда прав!



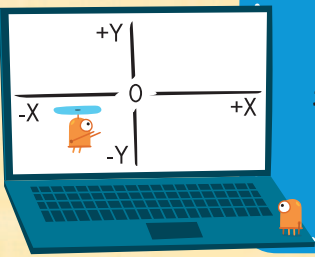
Если вы хотите, чтобы кот всегда появлялся в центре экрана, поставьте нули в обоих полях.



КООРДИНАТЫ

Мы можем задать положение любой точки на сцене при помощи двух чисел: **x** (право-лево) и **y** (верх-низ). Эти значения **x** и **y** называются **КООРДИНАТАМИ**.

Когда **x** и **y** равны нулю, вы в центре сцены.



Достигаем большего

На этой странице вы сможете узнать, как усовершенствовать свою игру «кошки-мышки», добавив в нее подсчет очков. Также вы можете добавить в игру еще одного мультяшного персонажа — мышь, создав для этого еще один спрайт.

Эти незаменимые переменные

Когда вы играете в игры, то счет не может не меняться. Так что компьютер должен отслеживать эту величину, и вы должны дать имя этому фрагменту информации. В программировании такое имя называется **переменной**.

Создание переменной

1 Зайдите в меню **Data** («Данные») и выберите опцию «Создать переменную» (Make a variable). Напечатайте имя переменной «Счет» во всплывающем окошке. Кнопку «Для всех спрайтов» (For all sprites) можно оставить нажатой. Для завершения операции нажмите «ОК» и...

...вы увидите целый набор новых блоков, связанных с переменной «Счет».

Оставьте эту рамку с блоками поверх, чтобы значение переменной изображалось на сцене — части экрана, где вы увидите результат исполнения скрипта.

2 Вставьте блок **Set score** («Установить счет») в начало программы. Для этого просто перетащите блок на нужное место.

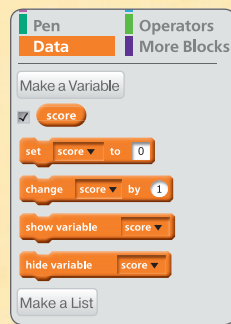
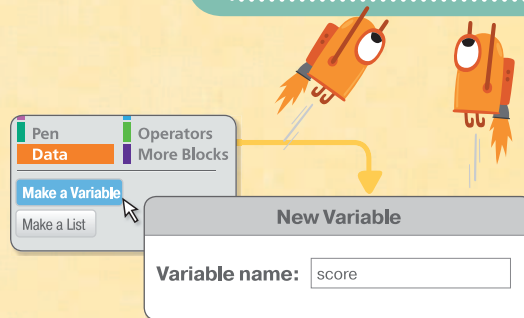
Вставьте в цикл блок **change score** («Изменить счет») и прибавляйте одно очко каждый раз, когда вам удаётся увернуться от кота.

Теперь вы увидите в углу **сцены** счетчик своих удач, и число, которое он показывает, будет расти на единицу при каждой вашей победе и обнулится, когда кот вас схватит.

ПЕРЕМЕННЫЕ

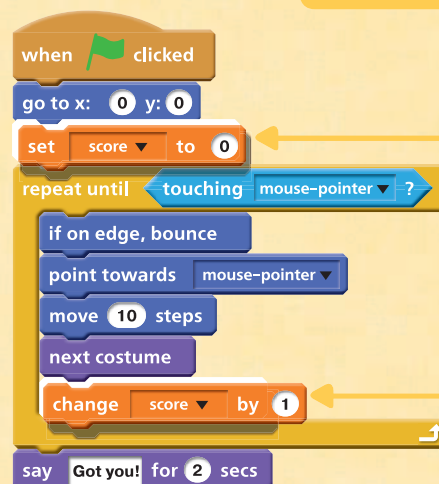
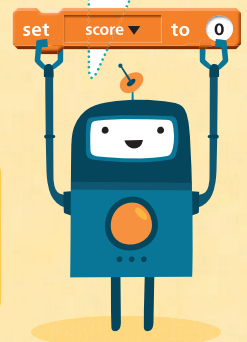
Переменные подобны боксу хранения данных с именем. Вы можете изменять его содержимое как хотите, но имя менять не можете. Причем имя можно выбрать практически любое (некоторые из имен вам не даст выбрать сам компьютер, поскольку они уже зарезервированы операционной системой). Но такие имена, как «Счет», «Наилучший счет» можно выбирать без проблем.

Любая информация, которую вы даете компьютеру, должна быть помечена именем, в противном случае она потеряется.



Впишите здесь имя своей новой переменной.

Какое бы имя вы не ввели, оно отобразится в новых блоках.



Этот блок гарантирует, что ваш счет очков каждый раз будет начинаться с нуля.

Если вы хотите быстрее набирать очки, впишите здесь не 1, а большее число.