

12+

Денис Владимирович Голиков

ЗНАКОМЬТЕСЬ, ЭТО SNAP!

БЛОЧНАЯ СРЕДА
ПРОГРАММИРОВАНИЯ МОЩНЕЕ
SCRATCH

Денис Голиков

**Знакомьтесь, это Snap! Блочная
среда программирования
мощнее Scratch**

«Издательские решения»

Голиков Д. В.

Знакомьтесь, это Snap! Блочная среда программирования мощнее Scratch / Д. В. Голиков — «Издательские решения»,

ISBN 978-5-44-856644-8

Книга знакомит с основами новейшего блочного языка программирования Snap! Этот язык настолько мощен, что используется не только в школах, но и в университетах США. Модификации Snap! используются для программирования Arduino, для моделирования многоагентных систем, графов и рисования сложных фигур.

ISBN 978-5-44-856644-8

© Голиков Д. В.
© Издательские решения

Содержание

Описание Snap!	6
Разработчики Snap!	8
Основные отличия от Scratch	9
Совместимость	12
Основные понятия	13
Знакомство с интерфейсом Snap!	15
Элементы интерфейса	16
Строка меню. Первая кнопка	17
Project notes	18
New	19
Open	20
Save и Save as	21
Costumes...	24
Sounds...	25
Строка меню. Вторая кнопка	26
Login...	27
Signup...	28
Reset Password...	29
Строка меню. Третья кнопка	30
Language...	31
Zoom blocks...	32
Stage size...	33
Long form input dialog	34
Plain prototype labels	35
Input sliders	36
Turbo mode	37
Flat design	38
Thread safe scripts	39
Prefer smooth animations	40
Inheritance support	42
Строка меню. Кнопки над сценой	43
Возврат к стандартному размеру сцены	43
Конец ознакомительного фрагмента.	44

Знакомьтесь, это Snap!

Блочная среда программирования мощнее Scratch

Денис Владимирович Голиков

© Денис Владимирович Голиков, 2026

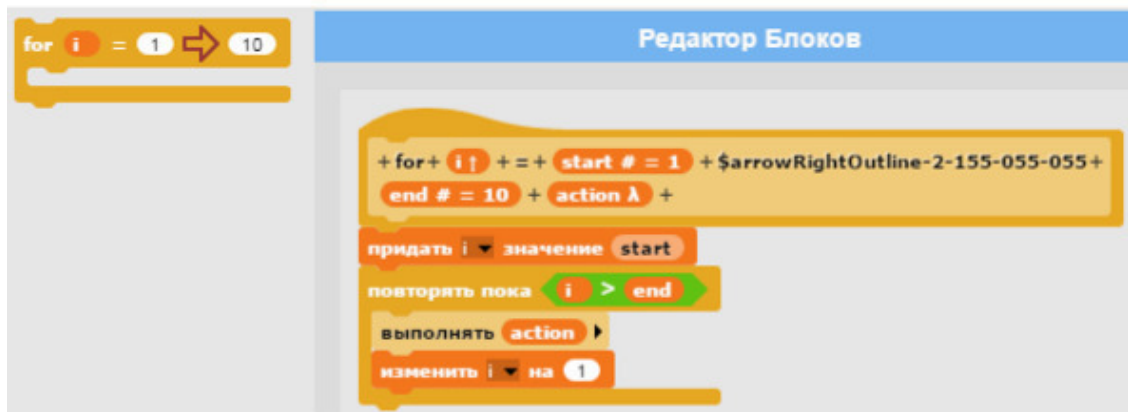
ISBN 978-5-4485-6644-8

Создано в интеллектуальной издательской системе Ridero

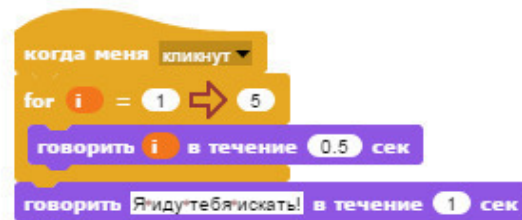
Описание Snap!

Snap! — блочная среда программирования, разработанная на основе Scratch в Калифорнийском университете в Беркли (University of California at Berkeley).

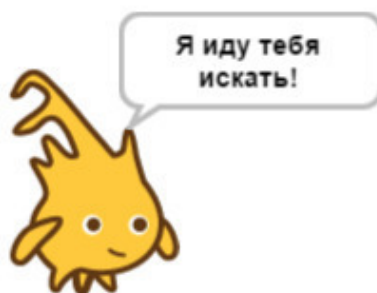
Основным отличием данной среды программирования от Scratch являются расширенные возможности по построению собственных блоков, а также списки и функции первого класса. Эти мощные возможности позволяют использовать Snap! не только в начальной школе, но и для обучения старшеклассников и студентов.



Пример создания цикла *for*.



Созданный блок *for* можно использовать наравне с любым встроенным блоком.



Алонзо досчитает до пяти, а затем скажет: «Я иду тебя искать!»

Snap! запускается и работает непосредственно в браузере и не требует установки на компьютер пользователя. Он написан на JavaScript и может оказывать ограниченное влияние на компьютер, так что вы можете безопасно запускать проекты других пользователей.

Разработчики Snap!

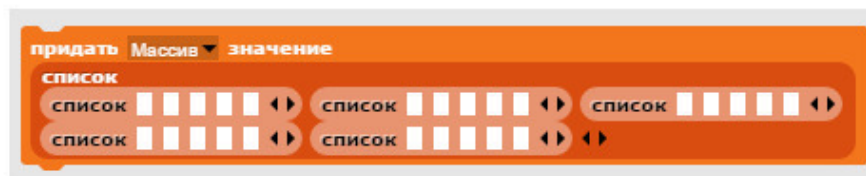
Snap! был разработан Дженсом Монигом (Jens Mönig at MioSoft Corporation, now at SAP), совместно с Брайаном Харви (Brian Harvey at Berkeley), при помощи студентов Калифорнийского университета в Беркли.

Основные отличия от Scratch

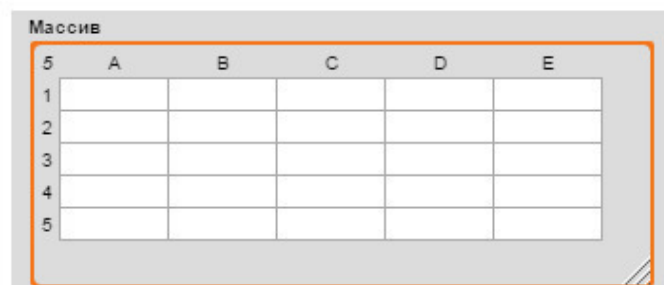
Типы данных.

Основным отличием Snap! от Scratch являются типы данных первого класса.

Объектами первого класса называются элементы, которые могут быть присвоены переменной или элементу массива, переданы как параметр и возвращены из функции. Благодаря этому в Snap! появилась возможность создавать двумерные массивы.

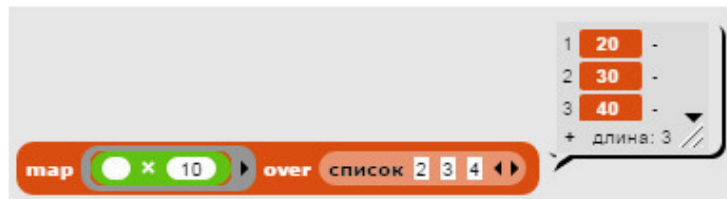


Создание двумерного массива.



Двумерный массив.

Также в Snap! в отличие от Scratch функции являются объектами первого класса. Это позволяет передавать функцию как аргумент в другую функцию. В следующем примере мы передаем функции *тар* как аргумент функцию *умножить на 10*, и функция *тар* производит это действие со всем списком, переданным ей в качестве второго аргумента.



Пример использования функции.

Еще одним удивительным отличием является то, что спрайты, тоже являются объектами первого класса, и могут быть переданы в качестве аргументов.



Спрайт — объект первого класса.

А вот такое возможно в Scratch?



Один спрайт «думает» о другом спрайте.

Эти особенности Snap! позволяют познакомить учащихся с функциональным программированием в отличие от Scratch, который используется для знакомства с императивным программированием.

Медиаредакторы.

Графический редактор Snap! не является его сильной стороной и уступает по функционалу Scratch 2.0 и даже Scratch 1.4. В нем отсутствует векторный редактор, а в растровом отсутствует масштабирование, выделение, штамп и поворот.

Редактор звука в Snap! отсутствует полностью.

Я рекомендую импортировать в Snap! изображения и звуки, подготовленные во внешних онлайн реакторах, а в редакторе изображений лишь немного корректировать загруженные изображения.

Совместимость

Совместимость с браузерами.

Snap! может работать в следующих браузерах:

- Chrome, версия не ниже 43 (это рекомендуемый браузер для работы в среде Snap!)
- Safari версия не ниже 8.4;
- Firefox версия не ниже 38;
- Edge версия не ниже 12;
- Opera версия не ниже 32.

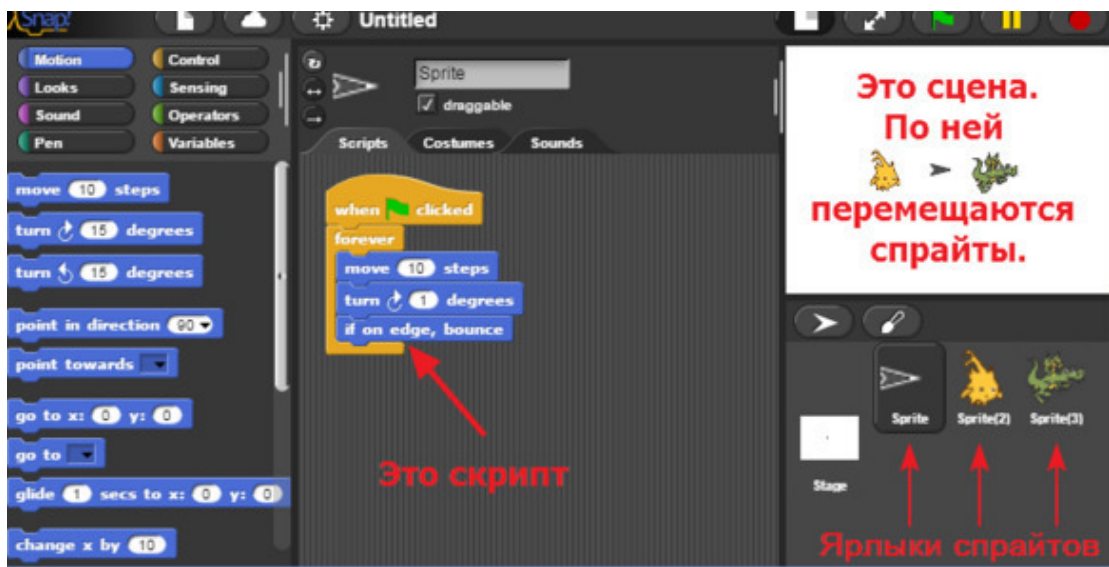
Совместимость с внешними устройствами.

Snap! может быть использован со следующими внешними устройствами, подключаемыми к компьютеру:

- Orbotix Sphero;
- Lego NXT;
- Nintendo Wiimote;
- Finch and Hummingbird robots;
- Parallax S2 robot;
- LEAP Motion;
- Speech synthesis by Connor Hudson;
- Arduino;
- Fischertechnik ROBOTICS TXT Controller;
- Raspberry Pi.

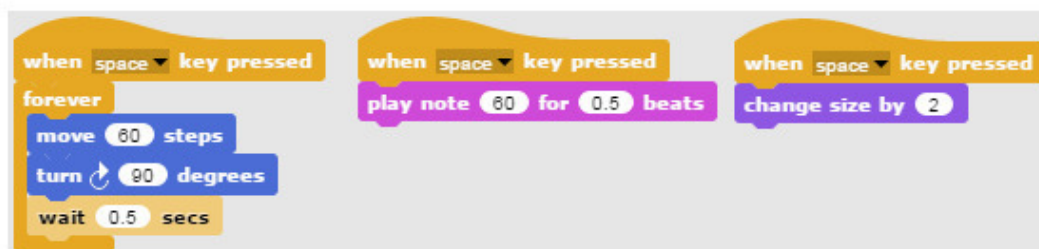
Основные понятия

Создаваемый проект состоит из отдельных спрайтов, которые перемещаются по сцене. Поведение каждого спрайта может быть запрограммировано с помощью скриптов, которых может быть несколько.



Скрипты собираются из отдельных блоков. Наверху каждого скрипта должен быть расположен блок с овальной шапкой. Этот блок содержит условие, при выполнении которого он начнет выполняться. Выполнение скрипта начинается сверху вниз, все блоки выполняются поочередно. Линейное выполнение программы может быть изменено с помощью блоков условия или цикла. Все скрипты проекта постоянно проверяют условия в своих шапках и при их выполнении запускаются одновременно.

Например, один скрипт перемещает спрайт, второй проигрывает звук, а третий изменяет его размер.



Все три скрипта выполняются одновременно.

Помимо прямоугольных блоков существуют овалы и шестиугольные блоки. Овалы являются встроенными переменными, функциями или операторами. Овалы вставляются в окошки других блоков.



Пример использования овального блока.

Шестиугольные блоки — это логические (булевы) переменные, логические операторы или логические выражения (предикаты). Они вставляются в окошки шестиугольной формы.



Пример использования предиката.

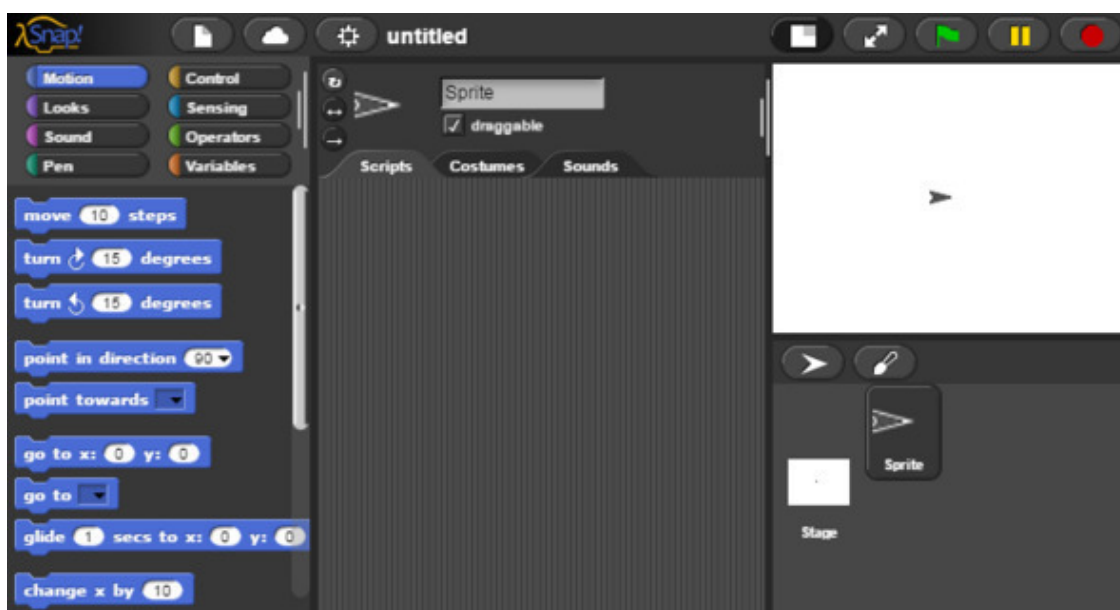
Знакомство с интерфейсом Snap!

Запуск Snap!

Найдите в интернете сайт Snap! и запустите его с помощью кнопки.

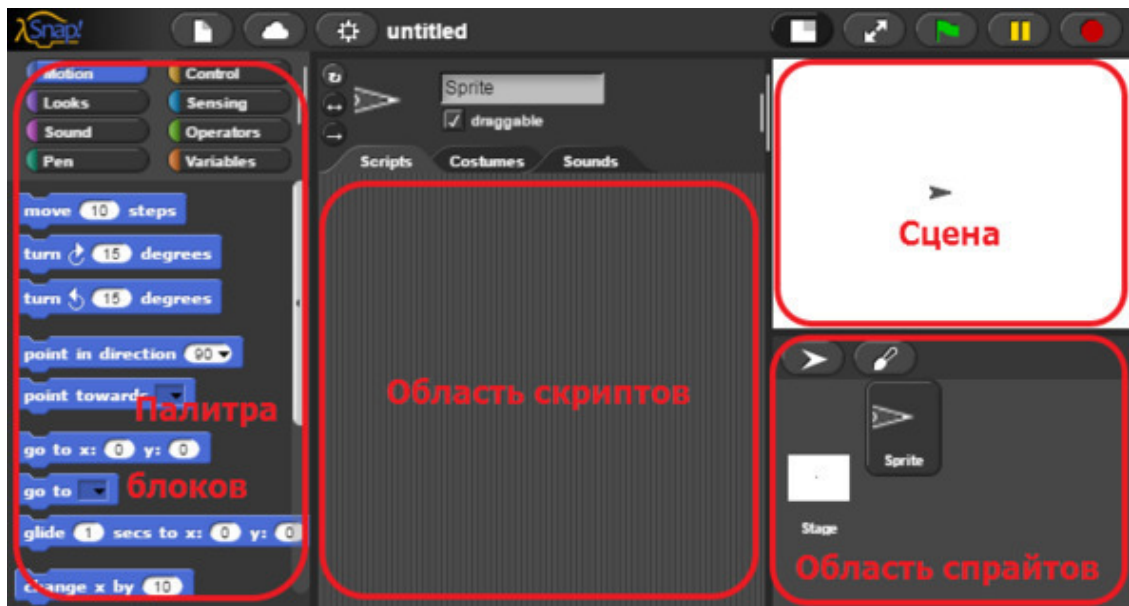
Run Snap! now

Кнопка запуска Snap!



Интерфейс Snap!

Элементы интерфейса



Интерфейс очень похож на интерфейс Scratch 1.4.

По умолчанию в Snap! установлен английский язык интерфейса. Однако его можно переключить на один из 38 языков.



Раздел меню для переключения языка.

Я не советую переключать его на русский, так как вся литература, по Snap! на английском языке.

Строка меню. Первая кнопка



Первая кнопка меню.

Первый раздел строки меню содержит следующие команды:

Project notes



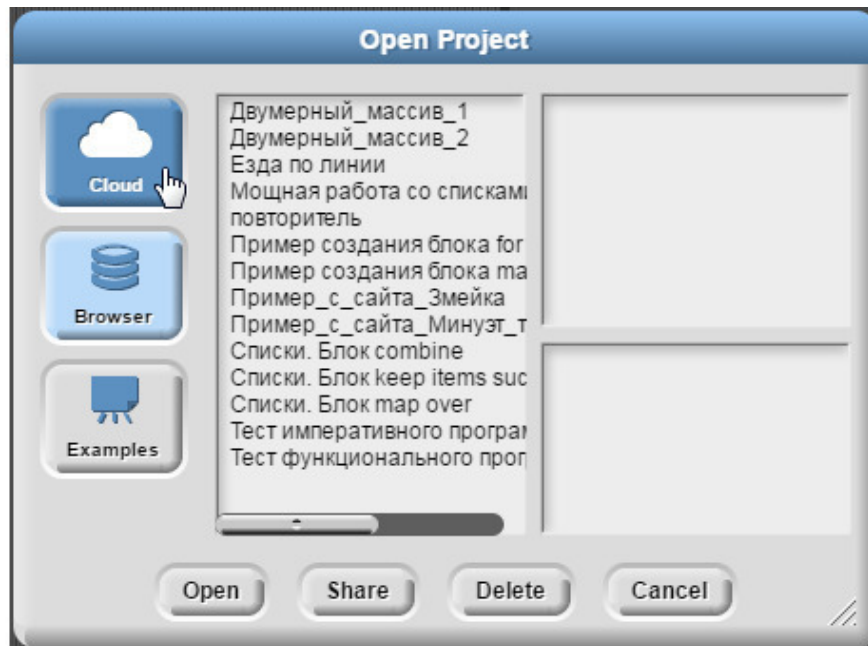
Текстовое описание проекта.

New

Создать новый проект.

Open

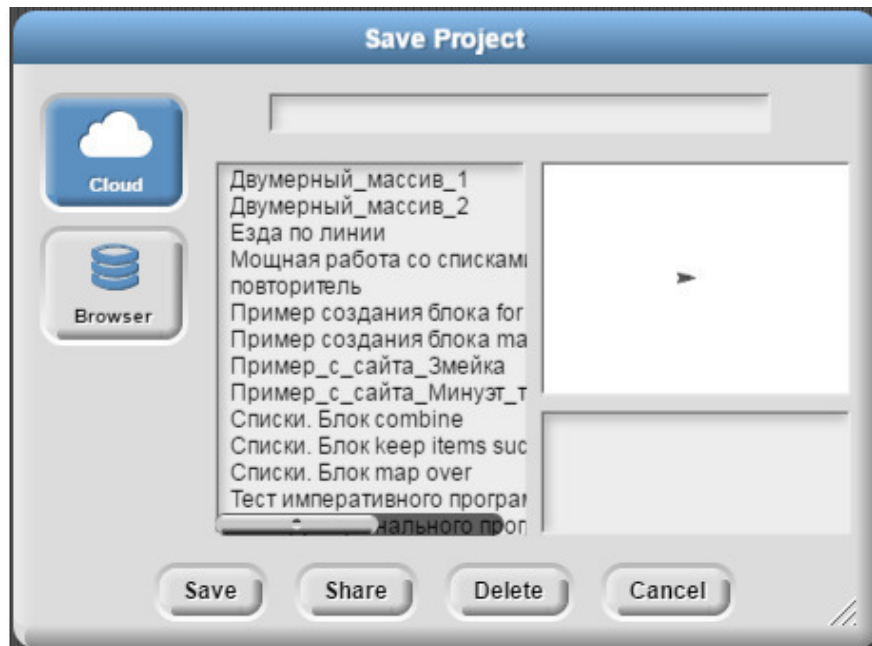
Открытие проекта. Проекты могут быть сохранены как на локальном компьютере (кнопка **Browser**), так и в облаке (кнопка **Cloud**). При сохранении файлов на локальном компьютере доступ к ним возможен только из Snap! Для сохранения в облаке нужно зарегистрироваться.



Окно открытия проекта.

Также есть возможность загрузить готовые примеры (кнопка **Examples**). Кнопка **Share** позволяет поделиться проектом. «Расшаренный» проект будет выделен жирным шрифтом. Для того чтобы им поделиться, надо скопировать ссылку на проект вида http://snap.berkeley.edu/snapsource/snap.html#present:Username=***&ProjectName=*** из строки браузера и отправить эту ссылку.

Save и Save as



Окно сохранения проекта.

Import

Этот раздел меню позволяет импортировать в проект следующие объекты:

- проект, экспортированный в файл формата XML;
- спрайт, экспортированный в файл формата XML;
- блок, экспортированный в файл формата XML;
- проект из Scratch 1.4 в формате SB;
- проект из Scratch 2.0 в формате SB2;
- проект из BYOB 3.1
- изображения (костюмы спрайтов, фоны сцены) в любом формате, поддерживаемом Вашим браузером.
- звуки в любом формате, поддерживаемом Вашим браузером.

Также для импорта любого файла в Snap! достаточно перетащить его мышкой в окно Snap! в браузере.

Export project...

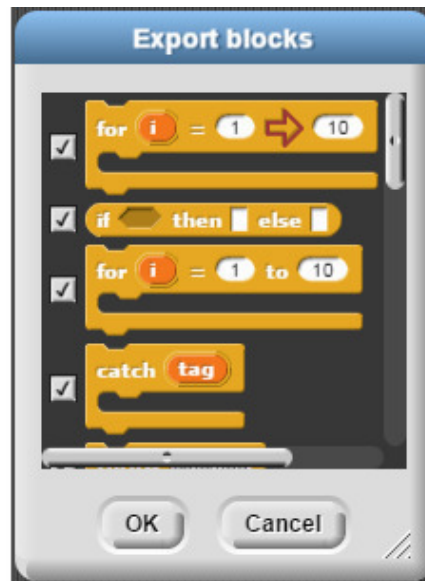
Экспорт проекта в файл XML.

Export summary...

Экспорт сводной информации о проекте в виде html страницы.

Export blocks...

Этот раздел меню появляется только в случае создания в проекте собственных блоков или импорта блоков из библиотек. Он позволяет сохранить созданные блоки в файл формата XML.



Окно экспорта блоков.

Unused blocks...

Этот раздел меню появляется только в случае создания в проекте собственных блоков или импорта блоков из библиотек. Он позволяет увидеть неиспользованные блоки чтобы удалить их и уменьшить размер готового проекта.

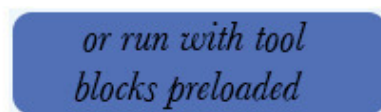


Окно удаления неиспользуемых блоков.

Import tools

Этот раздел позволяет импортировать большое количество полезных блоков созданных на Snap! Подробнее об этих блоках будет рассказано в следующих главах.

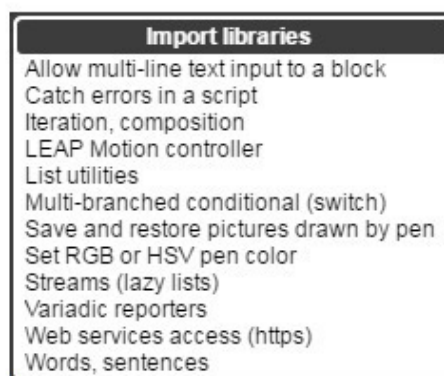
Snap! можно сразу запустить с автоматической загрузкой указанных блоков. Для этого на странице <http://snap.berkeley.edu/> есть кнопка <http://snap.berkeley.edu/init/>.



Кнопка запуска Snap! с автоматической загрузкой дополнительных блоков.

Libraries...

Импорт библиотек, расширяющих возможности Snap! и содержащих дополнительные полезные блоки, созданные на Snap!



Импорт библиотек.

Costumes...

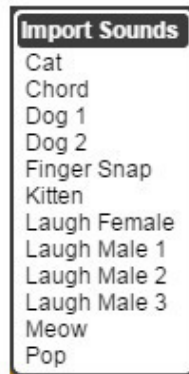
Импорт стандартных костюмов спрайтов.



Импорт стандартных костюмов.

Sounds...

Импорт стандартных звуков.



Импорт стандартных звуков.

Строка меню. Вторая кнопка



Вторая кнопка меню.

Login...

Вход в облачный сервис для зарегистрированных пользователей.



Вход в облачный сервис.

Signup...

Регистрация нового пользователя облачного сервиса.



Sign up

User name:

Birth date: year:

E-mail address:

Terms of Service... Privacy...

☐ I have read and agree to the Terms of Service

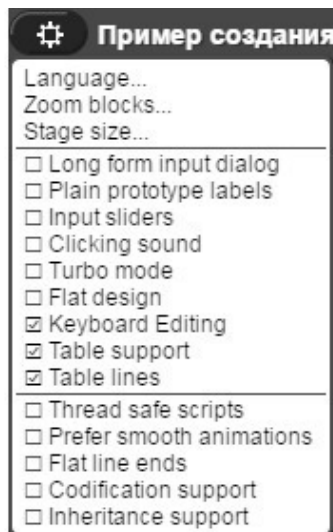
OK Cancel

Регистрация в облачном сервисе.

Reset Password...

Сброс пароля зарегистрированного пользователя.

Строка меню. Третья кнопка



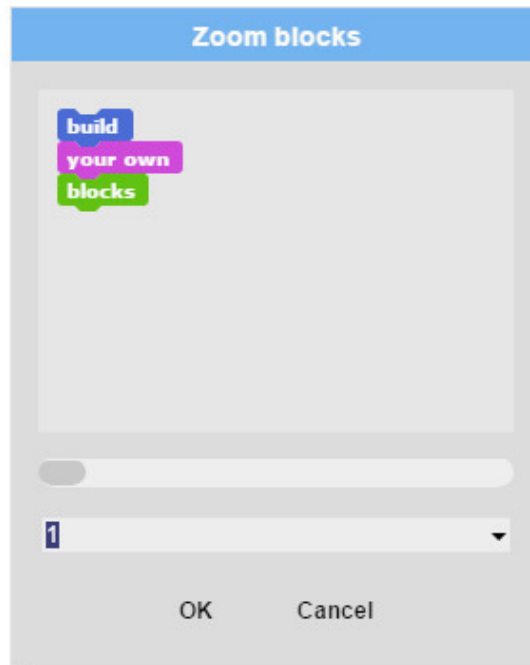
Третья кнопка меню.

Language...

Выбор одного из 38 языков интерфейса.

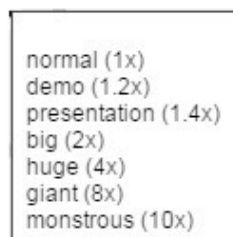
Zoom blocks...

Изменение размера блоков. Стандартный размер равен 1.



Окно изменения размеров блоков.

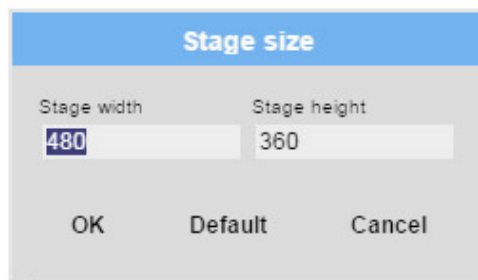
В выпадающем списке можно выбрать подходящий размер блоков.



Меню выбора размера блоков.

Stage size...

Изменить размеры сцены. Ширина изменяется в пределах от 480 до 1 000 000, а высота от 180 до 1 000 000. Не имеет смысла делать сцену размером больше экрана.



Окно изменения размеров сцены.

Long form input dialog

При создании переменной в пользовательском блоке вы увидите не простую форму ввода имени переменной.



Простая форма создания переменных.

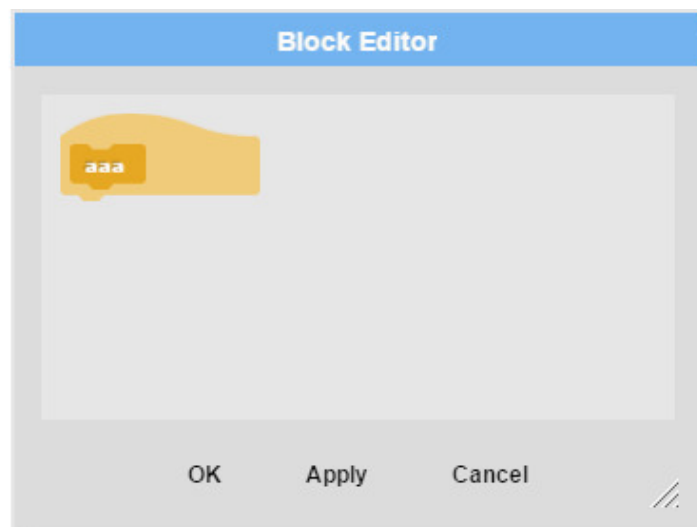
А сразу полную форму, которая появляется при нажатии на раскрывающийся список в предыдущей форме.



Расширенная форма создания переменных.

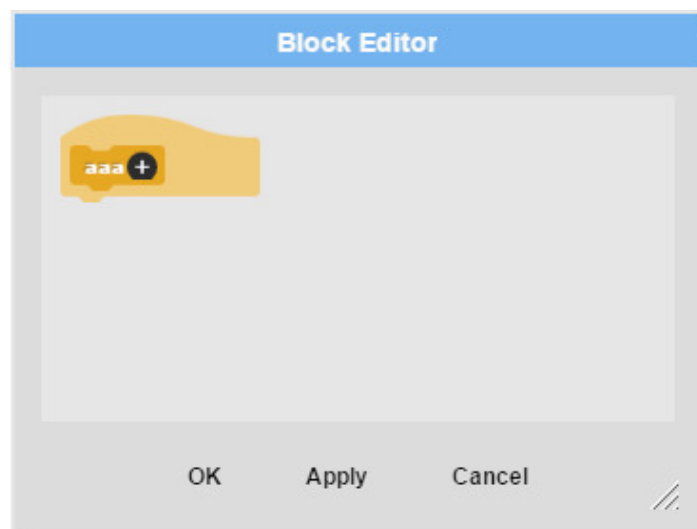
Рекомендую держать этот параметр всегда включенным.

Plain prototype labels



Убирает плюсики из редактора блоков.

Если немного поводить мышкой над именем блока, то появится плюсики, и можно вводить имена параметров.



Появился плюсики.

Эта функция используется для создания красивых скриншотов при написании руководств и учебников по Snap!

Input sliders



Отображение слайдеров для ввода числовых значений.

Clicking sound

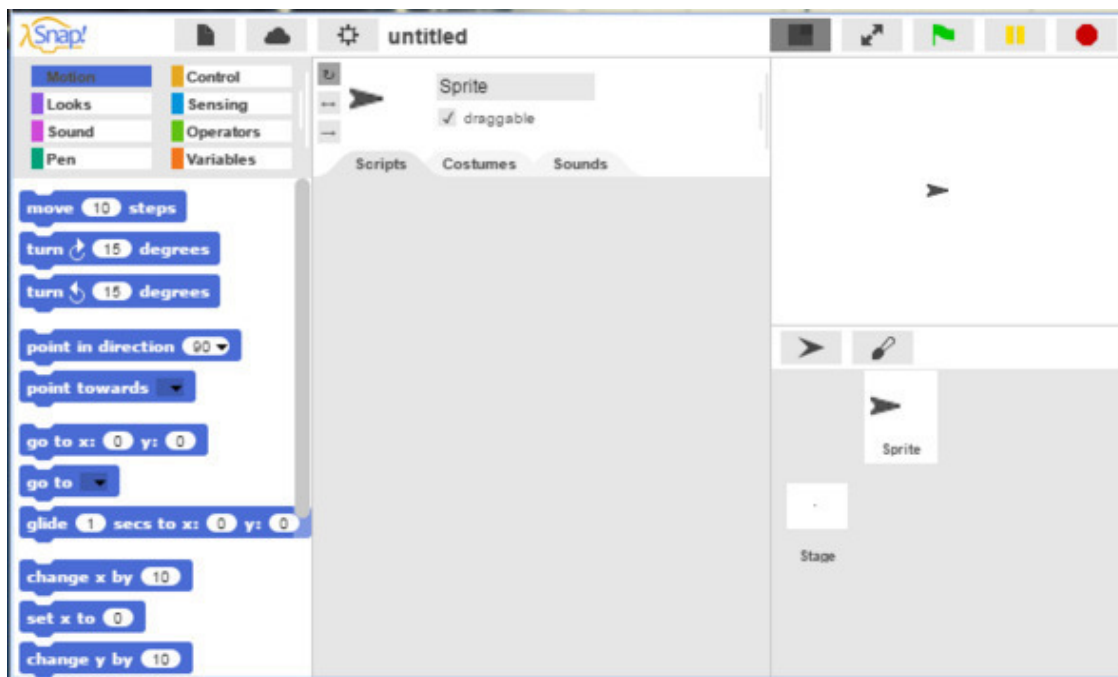
Звук при соединении блоков.

Turbo mode

Включение турбо режима, в котором сложные проекты работают быстрее.

Flat design

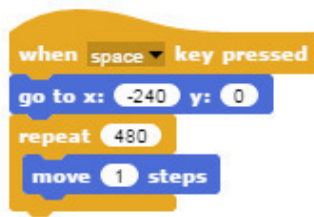
Переключение к плоскому дизайну в стиле Scratch 2.0. Далее везде будут использованы скриншоты этого дизайна.



Плоский дизайн в стиле Scratch 2.0

Thread safe scripts

Защита выполнения скрипта. Если скрипт, который начинается с какого либо события, выполняется, и это событие происходит снова, то скрипт начинает выполнение сначала, с самого верхнего блока. Например, если эта функция выключена, то при нажатии на пробел спрайт каждый раз будет начинать движение слева направо из точки (-240; 0). Если же галочка будет установлена, то скрипт будет выполнен до конца, не смотря на нажатия на пробел.



Пример скрипта.

Prefer smooth animations

Изменение частоты обновления сцены до 1/30 секунды, как в Scratch. В обычном режиме работы Snap! Обновляет экран с максимально возможной частотой. Это используется для более плавного отображения анимации движения спрайтов, при использовании большого количества различных подвижных объектов, а также для одинакового отображения анимации на компьютерах различной мощности.

Flat line ends

Закругление концов линий. Линии с закругленными краями выглядят аккуратнее, однако, в случае если надо нарисовать объекты прямоугольной формы, то эту опцию лучше не включать.



Линия с прямоугольным концом.



Линия с закругленным концом.

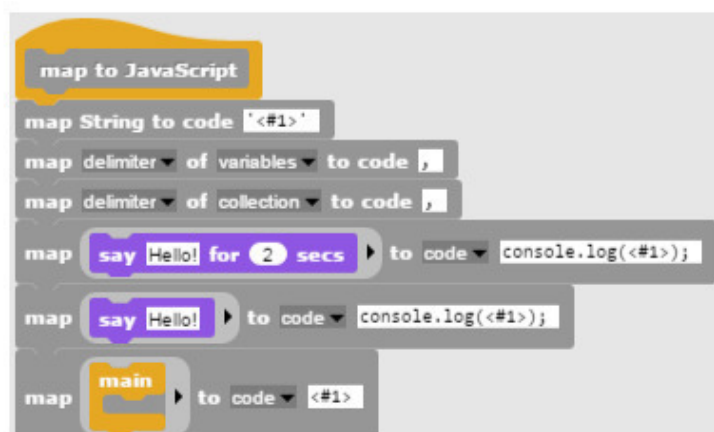
Codification support

Добавляет экспериментальную функцию позволяющую транслировать скрипты Snap! в программы на других языках программирования. Добавляет 4 блока.

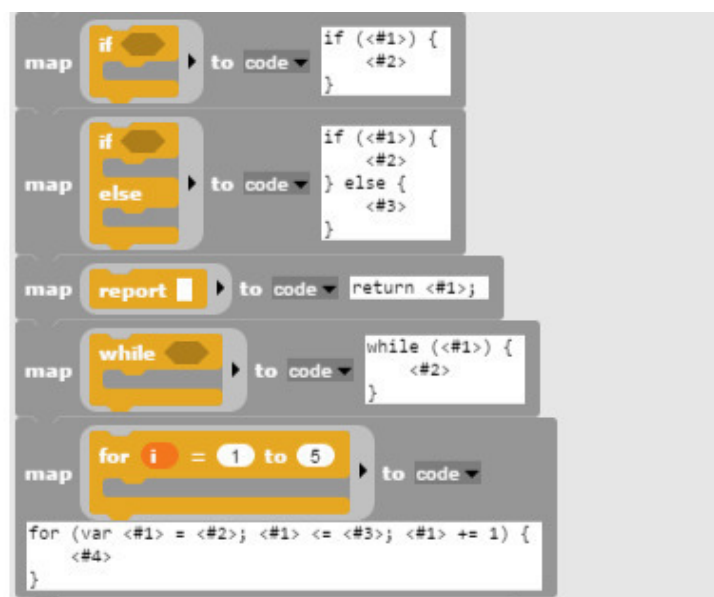


Поддержка трансляции.

Для знакомства с этими экспериментальными возможностями Snap! загрузите пример «Codification».



Часть блока транслирующего блоки в код на JavaScript.



Пример трансляции.

Inheritance support

Поддержка наследования.

Строка меню. Кнопки над сценой

Возврат к стандартному размеру сцены



Кнопка возвращает сцену к стандартному размеру.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.