

Содержание

- 8 **ВВЕДЕНИЕ** (Кэрол Вордемэн)
- 10 **КАК УСТРОЕНА ЭТА КНИГА**

1 ЧТО ТАКОЕ ПРОГРАММИРОВАНИЕ?

- 14 Что такое компьютерная программа?
- 16 Думай как компьютер
- 18 Как стать программистом

2 НАЧЕМ СО SCRATCH

- 22 Что такое Scratch?
- 24 Установка Scratch
- 26 Интерфейс Scratch
- 28 Спрайты
- 30 Цветные блоки и скрипты
- 32 **Проект 1: Убег от дракона!**
- 38 Перемещение объектов
- 40 Костюмы
- 42 Прятки
- 44 События
- 46 Простые циклы
- 48 Перья и черепашки
- 50 Переменные
- 52 Вычисления
- 54 Строки и списки
- 56 Координаты
- 58 Пошумим!
- 60 **Проект 2: Катись, кубик**
- 62 Истина или ложь?

- 64 Решения и ветвление
- 66 Считывание и распознавание
- 68 Сложные циклы
- 70 Обмен сообщениями
- 72 Создание блоков
- 74 **Проект 3: Обезьянье безумие**
- 82 Приступим к экспериментам!

3 ИГРЫ С PYTHON

- 86 Что такое Python?
- 88 Установка Python
- 92 Знакомство с IDLE
- 94 Ошибки
- 96 **Проект 4: Дом с привидениями**
- 98 Разбор «Дома с привидениями»
- 100 Выполнение программ
- 102 Простые команды
- 104 Команды посложнее
- 106 Разные окна
- 108 Переменные в Python
- 110 Типы данных
- 112 Вычисления в Python
- 114 Строки в Python
- 116 Ввод и вывод
- 118 Принятие решений
- 120 Ветвление

- 122 Циклы в Python
- 124 Цикл while
- 126 Выход из цикла
- 128 Списки
- 130 Функции
- 132 **Проект 5: Забавные фразы**
- 134 Кортежи и словари
- 136 Списки в переменных
- 138 Переменные и функции
- 140 **Проект 6: Чертежный автомат**
- 148 Ошибки и отладка
- 150 Алгоритмы
- 152 Библиотеки
- 154 Создание окон
- 156 Цвета и координаты
- 158 Рисование фигур
- 160 Изменение рисунков
- 162 Реакция на события
- 164 **Проект 7: Охотник за пузырями**
- 176 Что дальше?

4 УСТРОЙСТВО КОМПЬЮТЕРОВ

- 180 Внутри компьютера
- 182 Двоичная система
- 184 Символы и коды
- 186 Логические вентили

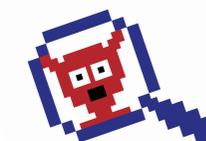
- 188 Процессоры и память
- 190 Необходимые программы
- 192 Хранение данных в файлах
- 194 Интернет

5 ПРОГРАММИРОВАНИЕ В РЕАЛЬНОМ МИРЕ

- 198 Компьютерные языки
- 200 Звезды программирования
- 202 Трудолюбивые программы
- 204 Компьютерные игры
- 206 Мобильные приложения
- 208 Программирование для Интернета
- 210 Использование JavaScript
- 212 Зловредные программы
- 214 Мини-компьютеры
- 216 Стань знатоком программирования

- 218 **ГЛОССАРИЙ**
- 220 **АЛФАВИТНЫЙ УКАЗАТЕЛЬ**
- 224 **БЛАГОДАРНОСТИ**

Find out more at:
www.dk.com/computercoding



Введение

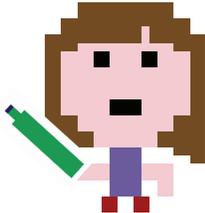
Совсем недавно компьютерное программирование казалось таинственным ремеслом, уделом специалистов. Мысль о том, что программирование может быть увлекательным занятием для каждого, большинству людей и в голову не приходила. Но мир изменился. Интернет, электронная почта, социальные сети, смартфоны и мобильные приложения, ураганом влетев в нашу жизнь, преобразили ее всего за несколько лет.

Компьютеры занимают очень важное место в современном мире, и мы принимаем это как должное. Вместо звонков по телефону мы посылаем текстовые сообщения или используем социальные сети. Мы охотно поглощаем любые плоды компьютеризации — от интернет-покупок и развлечений до новостей и игр. Однако мы можем не только использовать эти технологии — научившись программировать, мы можем развивать их, создавать собственные произведения цифрового искусства.

Всем, что делает компьютер, управляют строки программного кода, введенные с клавиатуры. Компьютерный код похож на иностранный язык, но язык этот может освоить каждый и довольно быстро. Многие считают, что программирование — одно из наиболее важных в XX веке умений.

Учиться программировать очень интересно, ведь ты можешь получить результаты сразу же и неважно, сколько еще материала предстоит изучить. Более того, создание игр и программ — такое увлекательное занятие, что очень скоро покажется, будто это почти не требует усилий. Это отличная возможность для творчества, наверное, первая область науки, совмещающая искусство, логику, сочинительство и бизнес.

Кроме того, умение программировать очень пригодится в жизни. Оно развивает логику и интеллект, которые важны в самых разных областях — от науки и инженерного дела до медицины и юриспруденции. Количество вакансий, где нужно умение программировать, будет со временем только расти, причем хороших программистов не хватает уже сейчас. Научись программированию — и цифровой мир будет открыт для тебя!


КЭРОЛ ВОРДЕМЭН 

Как устроена эта книга

Эта книга знакомит с основными понятиями, которые нужно усвоить, чтобы научиться программировать. Еще ты найдешь здесь увлекательные проекты, воплощающие идеи программирования на практике. Чтобы информация было легче понять и усвоить, она разбита на небольшие блоки.

Каждая тема рассматривается подробно, с примерами и упражнениями.

В блоках «Смотри также» перечислены другие относящиеся к теме разделы.

Пиксельные человечки дают подсказки и советы.



Семь проектов для развития навыков программирования. Страницы проектов выделены синей полосой

Простые пошаговые инструкции проведут тебя по каждому из проектов.

Пометки с указанием на строку, чтобы ничего не перепутать.

Эта иконка указывает, что продолжение проекта на следующей странице.

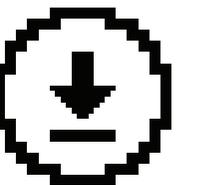
В рамках — дополнительная информация: подсказки, определения, важные для запоминания вещи.

Читай и начинай программировать!

СОВЕТЫ ЭКСПЕРТА

Сохранение

Иконка сохранения появляется на разворотах с проектами. Она напоминает тебе сохранять выполненную работу, чтобы ничего не потерялось в случае компьютерных проблем.



Сохраняй работу почаще.

Цветные рисунки поясняют различные понятия программирования.

Программный код и скрипты объясняются построчно.

Инструкции указывают, что надо нажать, перетащить или выделить.

Пометки поясняют каждый шаг

[Почитать описание, отзывы и купить на сайте МИФа](#)

42 STARTING FROM SCRATCH

Hide and seek

Welcome to the special effects studio! Using the purple "Looks" blocks, find out how to make sprites vanish and reappear, grow and shrink, and fade in and out.

Hiding sprites
To make a sprite disappear, use the "hide" block. The sprite is still on the stage, and it can still move around, but it can't be seen unless the "show" block is used to make it visible again.

Hide and show
To make a sprite vanish, use the "hide" block. When you're ready for it to be seen again, use the "show" block. These blocks are found in the "Looks" section of the blocks palette.

Disappearing cat
Try this script using the cat sprite. It disappears and reappears but it keeps moving, even when you can't see it.

EXPERT TIPS
Showing sprites
Select a sprite in the sprite list. Click the "i" button on it to open the information panel. There you can also use the "show" tick box to show or hide a sprite.

Sizes and effects

Scripts can be used to change the size of a sprite and add special effects to it.

change size by 10
Type in positive numbers to make sprites bigger and negative numbers to make them smaller.

change pixelate effect by 25
Choose the type of effect from the drop-down menu. The "pixelate" effect makes the sprite become blurred.

set size to 100 %
Higher numbers make sprites bigger and lower numbers make them smaller. 100 is normal size.

set color effect to 0
Each colour is represented by a number. Change the number to set the colour.

clear graphic effects
Resets all the effects.

Adding graphic effects
The graphic effects in Scratch can be used to change a sprite's appearance or distort its shape. They're fun to experiment with.

Using effects to teleport

Add a ghost sprite from the "Fantasy" category of the sprite library, and create the script shown below. It makes the ghost appear to teleport when clicked.

when this sprite clicked

- clear graphic effects
- repeat (20)
 - change ghost effect by 5
- glide (0.1) secs to x: pick random (-150) to (150) y: pick random (-150) to (150)
- repeat (20)
 - change ghost effect by -5

The "ghost" effect makes the sprite fade slightly; by repeating this block 20 times the sprite fades away completely.

This "Operators" block selects a random horizontal position.

This block selects a random vertical position.

This block makes the ghost move slowly, hidden from view.

Using this block makes the sprite fade back in.

170 PLAYING WITH PYTHON

BUBBLE BLASTER

Working out the distance between points

In this game, and lots of others, it is useful to know the distance between two objects. Here's how to use a well-known mathematical formula to have the computer work it out.

11 This function calculates the distance between two objects. Add this bit of code directly after the code you wrote in step 9.

```

from math import sqrt
def distance(id1, id2):
    x1, y1 = get_coords(id1)
    x2, y2 = get_coords(id2)
    return sqrt((x2 - x1)**2 + (y2 - y1)**2)
  
```

Loads the "sqrt" function from the Math library

Gets the position of the first object

Gets the position of the second object

Gives back the distance between them

13 Now update the main game loop to use the functions you have just created. Remember that the order is important, so make sure you put everything in the right place. Then run the code. Bubbles should burst when they hit the sub. Check the shell window to see the score.

```

score = 0
#MAIN GAME LOOP
while True:
    if randint(1, BUB_CHANCE) == 1:
        create_bubble()
    move_bubbles()
    clean_up_bubs()
    score += collision()
    print(score)
    window.update()
    sleep(0.01)
  
```

Sets the score to zero when the game starts

Creates new bubbles

Adds the bubble score to the total

Shows the score in the shell window - it will be displayed properly later

This pauses the action for a very short time - try removing this and see what happens

EXPERT TIPS
Python shortcut
The code "score += collision()" is a shortcut for writing "score = score + collision()". It adds the collision score to the total score, then updates the total score. Code like this is common, so a shortcut is useful. You can also do the same thing using the "-" symbol. For example, "score -= 10" is the same as "score = score - 10".

Don't forget to save your work

43 HIDE AND SEEK

Hide and seek

Use the "hide" block to make sprites disappear in games.

and the centre of their radiuses have collided.

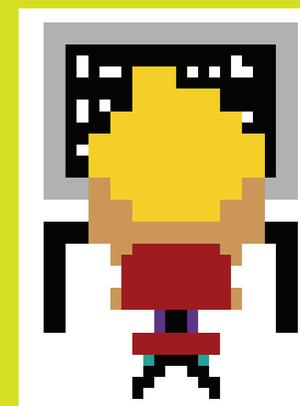
Checks for collisions between the sub and any bubbles

number of bubble is worth "points"

Don't forget to save your work

1

Что такое программирование?



[Почитать описание, отзывы и купить на сайте МИФа](#)

Что такое компьютерная программа?

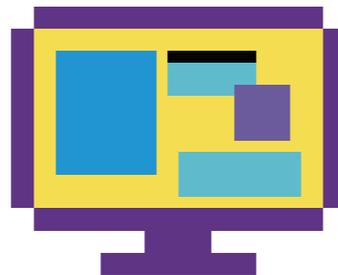
Компьютерная программа — это набор инструкций, следуя которым компьютер выполняет поставленную задачу. Программировать — значит писать для компьютера пошаговые инструкции, объясняющие, что и как ему нужно делать.

Компьютерные программы — повсюду

Нас окружают компьютерные программы. Множество приборов и мобильных устройств, которыми мы пользуемся изо дня в день, управляются программами. Эти устройства следуют пошаговым инструкциям, которые написаны программистами.

СМОТРИ ТАКЖЕ

Думай как компьютер	16-17 >
Как стать программистом	18-19 >



△ Компьютерные приложения

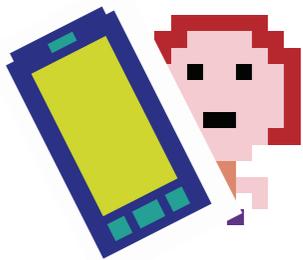
Все, что мы делаем с помощью компьютера, от просмотра сети Интернет до составления документов и проигрывания музыки, возможно благодаря коду, написанному программистами.

◁ Игры

Игровая приставка — это всего лишь специальный тип компьютера. Все игры, сделанные для приставок, — это программы. Графика, звук и управление в играх расписаны в программном коде.

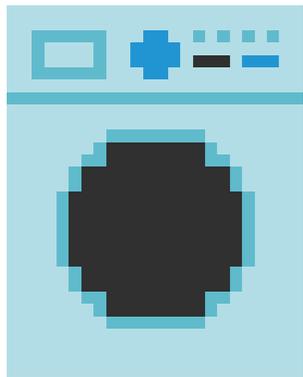
▷ Автомобили

В некоторых автомобилях компьютерные программы следят за скоростью, температурой воздуха и количеством топлива в баке. Программы могут даже помочь вовремя притормозить, чтобы поездка была безопаснее.



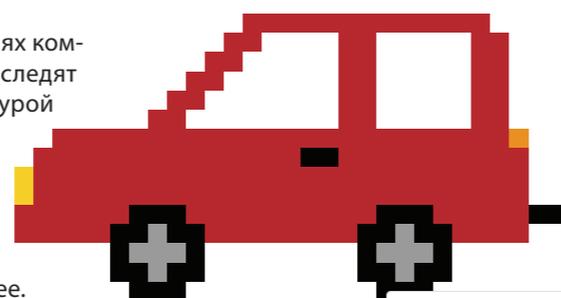
◁ Мобильные телефоны

Программы позволяют сделать звонок или отправить текстовое сообщение. При поиске контакта по имени программа сама находит нужный телефонный номер.



△ Стиральные машины

Стиральные машины программируются на разные режимы стирки. Компьютерный код следит за температурой воды и временем стирки.



[Почитать описание, отзывы и купить на сайте МИФа](#)

Как работают компьютерные программы

Компьютеры могут казаться очень умными, но это всего лишь напичканные электроникой ящики, которые умеют очень быстро и точно выполнять инструкции. Мы, разумные существа, можем давать компьютерам задачи, описывая их в виде программ, то есть пошаговых инструкций.

1 Компьютеры не умеют думать

Сам по себе компьютер ни на что не способен. Задача программиста — давать ему инструкции.



Без инструкций компьютер совсем бестолковый

Это компьютерная программа, выполняющая обратный отсчет перед запуском

2 Напиши программу

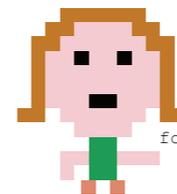
Ты можешь объяснить компьютеру, что делать, написав очень подробные инструкции, которые называются программой. Каждая инструкция должна быть достаточно простой, чтобы компьютер ее понял. Если инструкция написана неверно, компьютер поведет себя не так, как ты хочешь.

```
for count in range(10, 0, -1):
    print("Counting down", count)
```



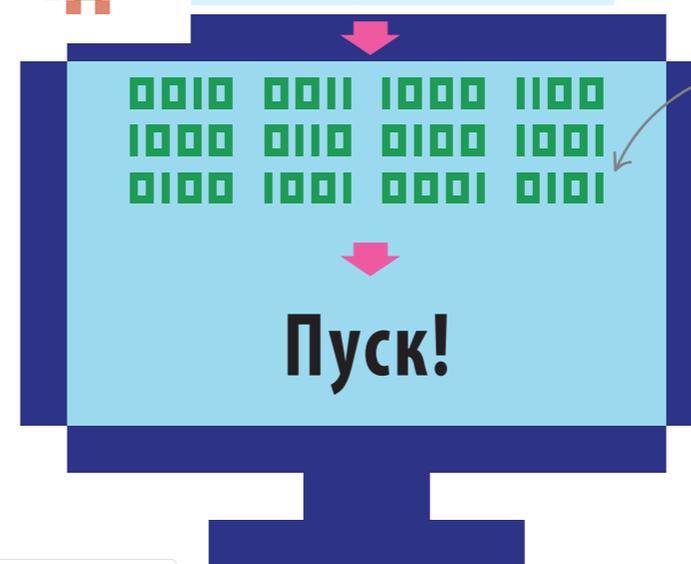
3 Языки программирования:

Компьютер может выполнять инструкции лишь на тех языках, которые ему понятны. Программистам приходится выбирать, какой язык лучше подходит для решения конкретной задачи.



```
for count in range(10, 0, -1):
    print("Counting down", count)
```

В итоге все программы становятся двоичным кодом — простейшим компьютерным языком, состоящим из нулей и единиц



СЛЕНГ:

Софт и железо

Железо (или аппаратные средства) — это части компьютера, которые ты можешь увидеть или потрогать (провода, схемы, клавиатура, экран и так далее). Софтом, или программными средствами, называют программы, которые выполняются на компьютере и управляют его работой. Аппаратные и программные средства работают сообща, чтобы компьютер мог выполнять полезные задачи.

Думай как компьютер

Программист должен научиться думать как компьютер. Все задачи нужно разбивать на небольшие подзадачи, которым легко следовать и которые невозможно понять неправильно.

Думай как робот

Представь себе кафе с официантом-роботом. Поскольку у робота бесхитрый компьютерный мозг, ему нужно объяснить, как доставлять блюда с кухни посетителям, ожидающим в обеденном зале. Сперва эту проблему нужно разбить на простые, понятные компьютеру подзадачи.

1 Программа 1 для робота-официанта

Следуя этой программе, робот берет еду с тарелки, вламывается из кухни в обеденный зал прямо сквозь стену и кладет еду на пол. Этот алгоритм составлен недостаточно подробно.

1. Взять еду.
2. Идти из кухни к столу посетителя.
3. Положить еду.

2 Программа 2 для робота-официанта

На этот раз мы сказали роботу, чтобы он шел через кухонную дверь. Так он и делает, но затем натывается на местного кота, падает и разбивает тарелку.

1. Взять тарелку с едой.
2. Идти из кухни к столу посетителя таким образом:
 - идти к кухонной двери;
 - идти от двери к столу посетителя.
3. Поставить тарелку на стол перед посетителем.

СМОТРИ ТАКЖЕ

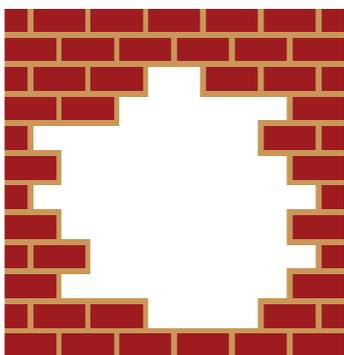
◀ 14–15 Что такое компьютерная программа?

Как стать программистом 18–19 ▶

СЛЕНГ:

Алгоритм

Алгоритм — это набор простых инструкций, объясняющих, как выполнить задачу. Программа — это алгоритм, переведенный на понятный компьютеру язык.



◀ Беда!

Инструкции слишком неоднозначны: мы забыли объяснить роботу, что идти нужно через дверь. Людям это показалось бы очевидным, но компьютеры не умеют думать сами.



△ Все еще не идеально

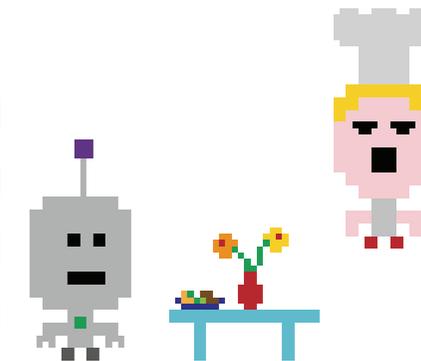
Робот не знает, как быть с препятствиями (вроде кошек). Чтобы он мог безопасно перемещаться, программа должна содержать еще более подробные инструкции.

[Почитать описание, отзывы и купить на сайте МИФа](#)

3 Программа 3 для робота-официанта

Следуя этому варианту программы, робот благополучно доставляет блюдо посетителю, огибая препятствия. Однако, поставив тарелку на стол, робот застывает на месте, а на кухне тем временем скапливается приготовленная еда.

1. Взять тарелку с едой и держать ее ровно.
2. Идти из кухни к столу посетителя таким образом:
 - идти к кухонной двери;
 - если замечено препятствие, обойти его;
 - идти от двери к столу посетителя;
 - если замечено препятствие, обойти его.
3. Поставить тарелку на стол перед посетителем.



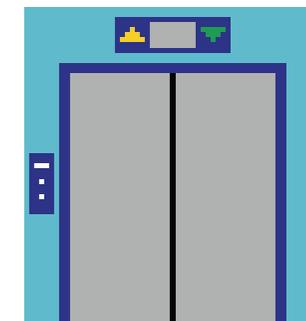
△ Неужели получилось?

Наконец-то робот смог благополучно доставить еду. Но мы забыли сказать ему, что потом нужно вернуться на кухню за новой тарелкой.

Пример из жизни

Может, наш робот-официант и выдуманный, однако алгоритмы такого типа встречаются повсюду. Например, управляемый компьютером лифт имеет дело со схожими проблемами. Вверх или вниз ему следует двигаться? На какой следующий этаж ехать?

1. Ждать до закрытия дверей.
2. Ждать нажатия кнопки.
 - Если нажата кнопка этажа выше текущего:
 - начать движение вверх.
 - Если нажата кнопка этажа ниже текущего:
 - начать движение вниз.
3. Ждать, пока этаж не будет соответствовать нажатой кнопке.
4. Открыть двери.



◀ Программа для лифта

Чтобы лифт работал правильно и безопасно, каждый шаг программы должен быть точным, понятным и учитывать все возможные случаи. Программисты должны убедиться, что они составили подходящий алгоритм.

Как стать программистом

Программисты (или кодеры) — это люди, которые создают программы, управляющие всем, что мы видим и делаем на компьютере. Ты сможешь писать собственные программы, если освоишь какой-нибудь язык программирования.

Языки программирования

Языков программирования очень много. Каждый из этих языков можно использовать для выполнения разных задач. Вот некоторые из самых популярных языков и цели, для которых их чаще всего используют:

СМОТРИ ТАКЖЕ

Что такое Scratch? **22–23** >
 Что такое Python? **86–87** >

C Мощный язык для написания компьютерных операционных систем;

Ada Используется для управления космическими кораблями, спутниками и самолетами;

Java Работает на компьютерах, мобильных телефонах и планшетах;

MATLAB Идеален, когда нужно выполнять много математических вычислений;

Ruby Для автоматического отображения информации на интернет-страницах;

Javascript Язык для написания интерактивных веб-сайтов;

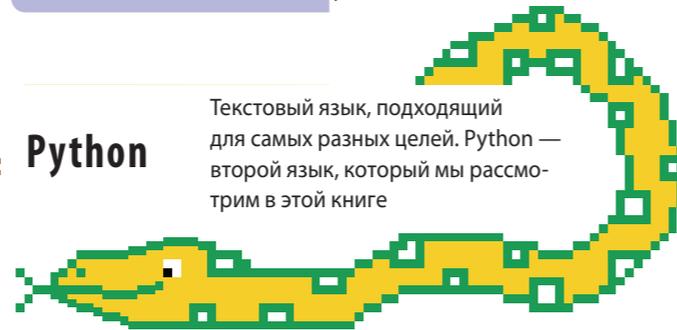
Scratch

Визуальный язык, который идеально подходит для изучения программирования. Scratch — первый язык, с которым мы познакомимся в этой книге;



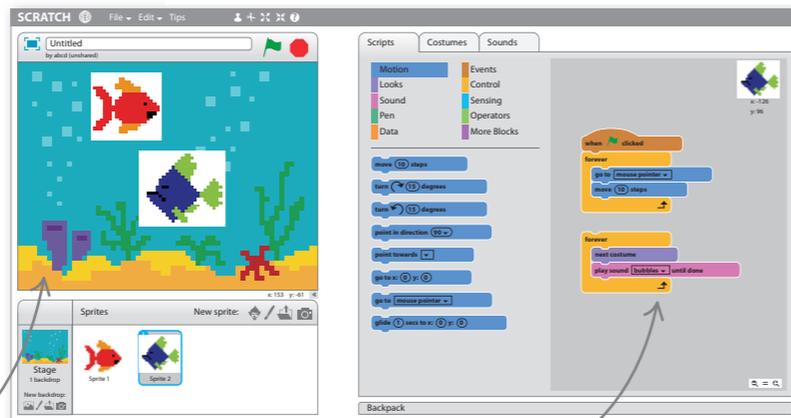
Python

Текстовый язык, подходящий для самых разных целей. Python — второй язык, который мы рассмотрим в этой книге



Что такое Scratch?

Scratch (читается «скрэтч») отлично подходит, чтобы начать программировать. Программы на этом языке состоят не из текстовых инструкций, а из блоков кода, соединенных между собой. Scratch — простой и эффективный язык, знакомящий с основными понятиями, необходимыми, чтобы программировать на других языках.



Программа запускается на этой части экрана

Код создают, соединяя цветные блоки между собой

[Почитать описание, отзывы и купить на сайте МИФа](#)

Что такое Python?

Люди по всему миру используют Python (читается «пайтон») для создания игр, приложений и веб-сайтов. Освоив этот замечательный язык, можно писать самые разные программы. Код на Python состоит из слов английского языка и символов.

Программа, написанная на Python

```

IDLE  File  Edit  Shell  Debug  Window  Help
ghostgame
# Ghost Game
from random import randint
print('Ghost Game')
feeling_brave = True
score = 0
while feeling_brave:
    ghost_door = randint(1, 3)
    print('Three doors ahead...')
  
```

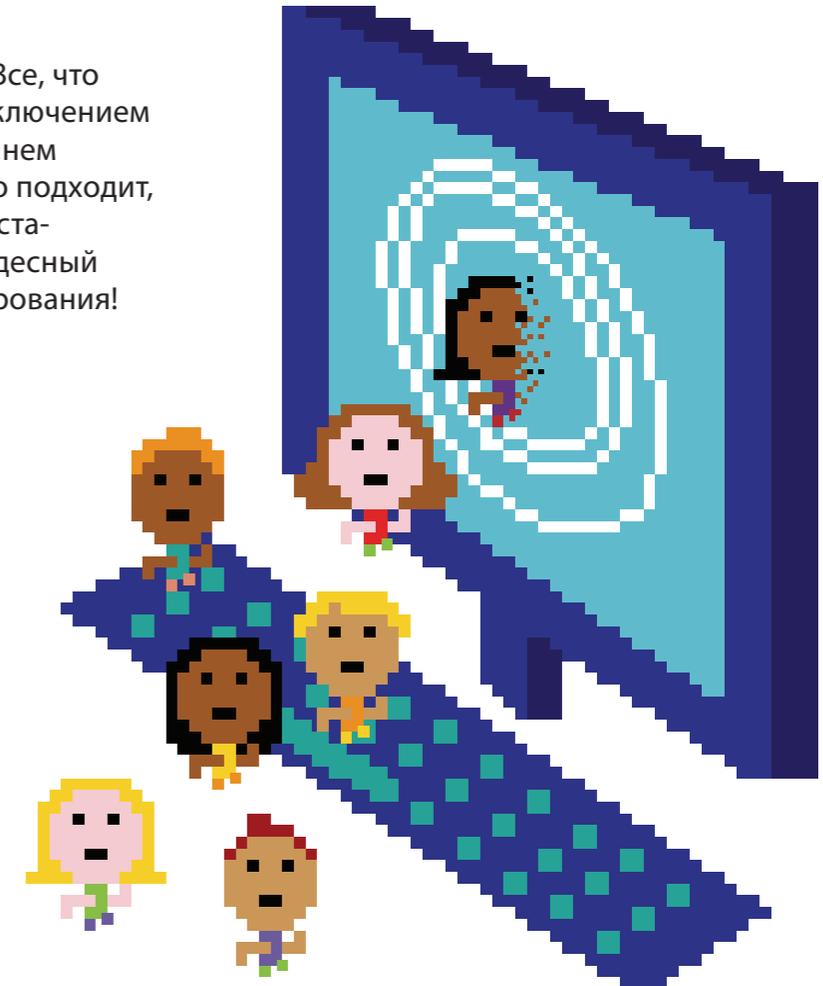
Приступим

Пора начать программировать! Все, что тебе нужно, — компьютер с подключением к Интернету. В этой книге мы начнем с языка Scratch, который отлично подходит, чтобы встать на путь программиста-знатока. Приготовься войти в чудесный мир компьютерного программирования!

■ ■ СОВЕТЫ ЭКСПЕРТА

Экспериментируй!

Тебе как программисту-новичку полезно экспериментировать с кодом своих программ. Один из лучших способов чему-то научиться — это менять разные части кода и смотреть, что произойдет. Играя с программой, ты найдешь новые способы решения задач, узнаешь больше о программировании и тебе станет еще интереснее!





[Почитать описание, рецензии
и купить на сайте](#)

Лучшие цитаты из книг, бесплатные главы и новинки:

