

ОГЛАВЛЕНИЕ

Родители, спокойствие!	7
Веб-дизайн	9
Миссия 1. Создаем сайт — сенсационный блог для Стива	10
Миссия 2. Улучшаем сайт: для чего Стиву стили CSS	18
Миссия 3. Налаживаем общение: знакомство с JavaScript	29
Миссия 4. В путь: улучшаем работу сайта с помощью циклов	35
Миссия 5. Играем со Стивом: сочиняем игру	43
Анимационные приключения	47
Миссия 1. Работаем с персонажами: ресурсы и анимация	48
Миссия 2. Курица против крипера: программируем движение	53
Миссия 3. План спасения курицы: да здравствует вращение!	60
Миссия 4. Удивительные яйца: играем с плагинами	67
Сочиняем музыку	75
Миссия 1. Творчество Стива: создаем онлайн-пианино	76
Миссия 2. Импровизация: добавляем аудиоэффекты	84

Миссия 3. В очереди на прослушивание: музыка на сайте Scratch	88
Миссия 4. Номинация на золото: сочиняем музыку Minecraft	95
Миссия 5. Война миров: Новые звуки в мире Minecraft	99
Большая игра	107
Миссия 1. Лабиринт в духе Minecraft: простая игра с помощью Scratch.....	108
Миссия 2. На просторах Интернета: создаем онлайн-игру	117
Миссия 3. Очки против криперов: программируем игру на результат.....	124
Миссия 4. Новый уровень: совершенствуем игру	135
Миссия 5. Внутри Minecraft: замена объектов	143
Глоссарий	155
Исправляем ошибки	157

РОДИТЕЛИ, СПОКОЙСТВИЕ!

Время, когда компьютерная игра считалась пустой тратой времени, осталось позади. В наши дни такие игры как Minecraft способствуют развитию творческих способностей, формируют навыки программирования. Однако большая часть учебников и книг об основах программирования скучны — в них много теории. Да и зачем она, если, изучив ее, не получается воплотить в жизнь свои идеи? В них мало практики.

Книга «Изучаем программирование на примере Minecraft» в противоположность остальным пособиям прививает любовь к программированию, превращая его из рутины в хобби. Мы, конечно, рассмотрим основы — переменные, циклы, функции, но только через призму их возможностей. Мы создадим удивительные вещи!

В этой книге мы будем делать все задания в мире Minecraft с участием любимых персонажей. Каждый найдет что-то интересное для себя — будь то создание сайта Стива, анимация крипера или изменение непосредственно игрового мира.

Сегодня ваши дети просто играют в Minecraft и изучают основы написания кода, а в будущем станут крутыми профессионалами. Давайте поможем им сделать первые шаги к мечте! Их ждет удивительное путешествие в мир программирования.

Для выполнения миссии

НЕОБХОДИМО

- Игра Minecraft, установленная в операционной системе Windows или macOS
- Текстовый редактор
- Пользователям Windows для работы подойдет текстовый редактор Блокнот. Тем, у кого macOS, можно работать в приложении TextEdit. Однако же мы рекомендуем использовать программы с цветным оформлением кода.

Для Windows: приложение Notepad++

Можно скачать по ссылке: notepad-plus-plus.org

Для macOS: Atom

Можно скачать по ссылке: atom.io

- Браузер и подключение к Интернету. Некоторые элементы кода на языке HTML5 неправильно работают в браузерах Internet Explorer и Mozilla Firefox. Поэтому мы рекомендуем использовать Google Chrome, если есть такая возможность. Важно на время изучения этой книги разрешить всплывающие окна на страницах, которые мы будем создавать вместе. Скачать Google Chrome можно по ссылке: www.google.com/chrome/

The background features a complex, layered geometric design. It consists of numerous overlapping, semi-transparent planes and rectangular shapes that create a sense of depth and perspective. The color palette is primarily light grays and whites, with thin, sharp red lines that intersect and define the various planes and edges of the structure. The overall effect is that of a modern, architectural or digital space.

ВЕБ-ДИЗАЙН

МИССИЯ I

СОЗДАЕМ САЙТ — СЕНСАЦИОННЫЙ БЛОГ ДЛЯ СТИВА

Стив хочет создать собственный сайт, но не знает как. Обычный текстовый редактор и язык программирования HTML помогут нам создать основу для сайта. В результате сегодняшней миссии мы построим фундамент, на котором позже разместим много интересных элементов. Повеселимся!



Язык HTML составляет основу веб-страницы. Важно заложить фундамент будущего сайта правильно. В конце концов, прежде чем строить дом, удостоверься, что фундамент его выдержит!

НА СТАРТ!

1. Открой текстовый редактор (при необходимости см. врезку «Для выполнения миссии **необходимо**»). Запомни, что программирование на языке **HTML ВСЕГДА НАЧИНАЕТСЯ СО СЛЕДУЮЩИХ СТРОК:**

```
<!DOCTYPE html>  
<html>
```

Элемент DOCTYPE сообщает браузеру, какую версию языка HTML ты используешь при создании сайта. До выхода пятой версии HTML — HTML5 — эта строка могла быть очень длинной. Теперь намного проще запомнить, что именно нужно указывать в элементе DOCTYPE. И это отлично, ведь такая строка должна быть в начале каждой страницы сайта! Давай пропишем элемент DOCTYPE прямо сейчас.

2. Видишь текст, заключенный между угловыми скобками? Это **HTML-ЭЛЕМЕНТ**. Чтобы **ЗАКРЫТЬ ПОЧТИ ЛЮБОЙ HTML-ЭЛЕМЕНТ, ЕГО ПЕЧАТАЮТ СНОВА**, но с косой чертой перед первой буквой. **HTML-ЭЛЕМЕНТЫ СОСТОЯТ ИЗ ОТКРЫВАЮЩЕГО И ЗАКРЫВАЮЩЕГО ТЕГОВ**. Итак, мы открыли раздел HTML-документа с помощью открывающего тега `<html>` элемента `html`. Теперь мы должны закрыть раздел с помощью закрывающего тега `</html>` элемента `html` внизу страницы так, как показано на рисунке:

```
<head>
```

3. Теперь, как и у Стива, у **НАШЕЙ СТРАНИЦЫ ДОЛЖНЫ ПОЯВИТЬСЯ ГОЛОВА И ТЕЛО** (но без ножек!). Напечатай под тегом `<html>`:

```
<html>
<head>
</head>
<body>
</body>
</html>
```

«Голову» (все, что будет содержаться между тегами `<head>` и `</head>`) мы наполним важной информацией о веб-странице. А в тело (все, что будет содержаться между тегами `<body>` и `</body>`) поместим то, чем мы хотим делиться на нашем сайте. Пора заполнить нашу страницу!

4. Добавим несколько элементов `div` в тело нашего документа. **ЭЛЕМЕНТ `div` СЛУЖИТ КОНТЕЙНЕРОМ**,

в который можно поместить другие объекты. Нам понадобится несколько таких контейнеров, так что продолжим и **ДОБАВИМ ДВА ЭЛЕМЕНТА `div`** после открывающего тега `<body>`. Обязательно добавь закрывающие теги `</div>`!

ВОЛШЕБНЫЙ КОНТЕЙНЕР

X

Контейнер `div` можно сравнить с сундуком в Minecraft. Он нужен, чтобы хранить содержимое в идеальном порядке, куда лучше, чем в твоей комнате! Так же, как сундук помогает улучшить внешний вид комнаты, элемент `<div>` поможет нам украсить сайт Стива. Этот маленький элемент кода может изменить все.



Теперь щелкни мышкой перед тегом `<div>` и **НАЖМИ КЛАВИШУ TAB** (эта клавиша находится рядом с клавишей буквы **Й** на клавиатуре). Заметил, как текст сдвинулся вправо? Программисты делают это для того, чтобы можно было легко увидеть структуру и содержимое других элементов. Элементы `div`, помещенные в «тело» нашего сайта (элемент `body`), должны быть немного сдвинуты вправо.

Также мы **ДОБАВИЛИ АТТРИБУТ ID** (он называется идентификатором) **В КАЖДЫЙ ТЕГ <DIV>**. Не волнуйся, пока это не имеет значения! Позже ты поймешь, для чего мы это сделали. Ты сможешь изменить значение атрибута `id` в любой момент, однако обрати внимание: важно случайно не добавить пробел!

```
<!DOCTYPE html>
<html>

<head>
</head>

<body>

    <div id = 'topbar'>
    </div>

    <div id = 'mainsection'>
    </div>

</body>

</html>
```

ВНИМАНИЕ! МАРШ!!

5. Наши контейнеры готовы. Пора заполнить их!

```
<body>
  <div id = 'topbar'>
    <p> Привет, я - Стив! Приветствую тебя на моем сайте! </p>
  </div>

  <div id = 'mainsection'>
    <p> Кто я?: </p>
    <p> Что я делаю?: </p>
    <p> Больше всего я люблю: </p>
  </div>
</body>
```

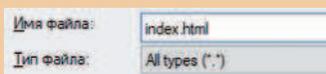
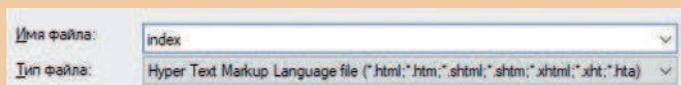
Ты видишь, что в контейнер `div` мы поместили элементы `p`? Они **ЗАПОЛНЯЮТ САЙТ ТЕКСТОМ** и разделяют его на абзацы. Текст в каждом `p` автоматически начинается с новой строки. Текст легче читать, если он разделен на абзацы.

ПРОЧИТАЙ ВОПРОСЫ, ЗАКЛЮЧЕННЫЕ МЕЖДУ ЭЛЕМЕНТАМИ `p`, и введи свои ответы перед закрывающим тегом `</p>`.

6. **ОТКРОЙ МЕНЮ ФАЙЛ** в левом верхнем углу окна, выбери в меню команду **СОХРАНИТЬ КАК**. Самое время присвоить имя твоей странице. Поскольку мы работаем над главной страницей будущего сайта, назовем ее **INDEX**. Создай папку, в которую сохранишь последующие файлы. Тогда ты легко найдешь их в нужный момент.

Если в открытом окне **СОХРАНИТЬ КАК** тебе предлагают выбрать тип файла, выбирай вариант HTML или Hyper Text Markup Language.

Если же такую команду ты найти не можешь, выбери вариант **ВСЕ ФАЙЛЫ**, а в конце имени файла укажи расширение **.HTML**. Теперь все готово — нажми кнопку **СОХРАНИТЬ**.



7. Готов посмотреть, что мы создали? Найди только что сохраненный тобой **HTML-ФАЙЛ И ЩЕЛКНИ ПО НЕМУ**. Перед тобой откроется созданная твоими руками страница сайта Стива!

Привет, я - Стив! Приветствую тебя на моем сайте!

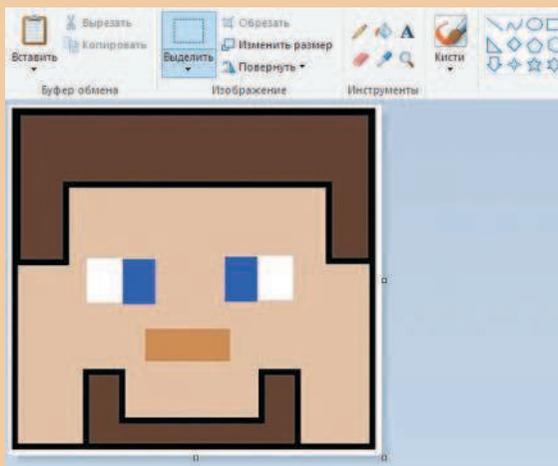
Кто я?: Я - игрок в Minecraft

Что я делаю?: Строю крутые здания в игре

Больше всего я люблю: Добывать алмазы

Выглядит неплохо, но пусто, не так ли? Давай добавим картинку!

8. Открой свою любимую программу для рисования, чтобы **НАРИСОВАТЬ И РАСКРАСИТЬ ПОРТРЕТ СТИВА**. Избавься от белых полей вокруг рисунка. В противном случае картинка будет окружена белым на веб-странице. Сделать это тебе поможет команда **ОБРЕЗАТЬ**. Выдели область рисунка, постарайся включить в область выделения весь рисунок без лишних белых участков.



Как только ты закончишь, выбери команду **СОХРАНИТЬ**. Присвой файлу с рисунком имя **STEVE.JPG**. Убедись, что в раскрывающемся списке Тип файла выбран вариант JPEG. Файл следует сохранить в ту же папку, куда ранее ты сохранил документ HTML.

9. Самое время вернуться в текстовый редактор, чтобы после первого тега `<div>` ввести следующее:

```
<img id = 'steve' src = 'steve.jpg' />
```

ВИДИШЬ, МЫ ПРИСВОИЛИ РИСУНКУ ИДЕНТИФИКАТОР ID? Он пригодится нам позже, когда мы решим изменить расположение или размер рисунка. Атрибут `src` позволит веб-странице безошибочно определить, где находится нужный рисунок. Особенность элементов `img` в том, что закрыть их можно более простым способом — с помощью символов `/>`. Но если тебе так удобнее, воспользуйся привычным закрывающим тегом ``.



Привет, я - Стив! Приветствую тебя на моем сайте!

Кто я?: Я - игрок в Minecraft

Что я делаю?: Строю крутые здания в игре

Больше всего я люблю: Добывать алмазы

Перезагрузи сайт, чтобы появился рисунок!

Мы проделали большую работу и создали твой первый сайт! Но до финиша еще далеко. В следующей главе мы расскажем, как улучшить внешний вид сайта Стива, используя разные цвета.

МИССИЯ 2

УЛУЧШАЕМ САЙТ: ДЛЯ ЧЕГО СТИВУ СТИЛИ CSS

До сих пор мы знакомили тебя с основами HTML. Теперь узнаем, что такое каскадные таблицы стилей, или CSS. Разобраться со стилями CSS довольно легко. Они позволят оформить сайт по твоему желанию. Готов узнать задание второй миссии?



Стив способен на многое! Так же как и его сайт — с твоей помощью и с помощью стилей CSS!