

Автономная некоммерческая организация
Дополнительного профессионального образовательная организация
«Компьютерная Академия ТОП»

КУРСОВОЙ ПРОЕКТ

**ПО ДИСЦИПЛИНЕ:
WEB-РАЗРАБОТКА**

НА ТЕМУ:

Создание web-страницы с применением JavaScript

Выполнил:
Студент 3 курса
Направления
«Разработка Программного Обеспечения»
Группы П22:
Вольных Василий Семёнович

Научный руководитель:
Силантьева Елена Викторовна

Липецк, 2024

Содержание

- Цели проекта и процесс его выполнения
- Основная часть проекта
 - Использование HTML/CSS
 - Использование JS
- Вывод
- Используемый список сайтов

Цели проекта и процесс его выполнения

Целями данного проекта являются:

- Усвоение материала по HTML/CSS и разработка собственного сайта
- Добавление и активное использование JS в проект
- Загрузка проекта в портфолио на платформу GitHub

Основная часть проекта

Использование HTML/CSS

1. Header:

Содержит название и навигационное меню, которое позволяет пользователям переходить между различными секциями сайта.

2. Main: Основная часть страницы, разделенная на несколько секций:

- Главная: Вводная информация обо мне, как о фрилансере.
- Обо мне: Подробности о моём опыте и навыках.
- Портфолио: Пример проекта (игра "Змейка"), который можно реализовать с помощью HTML5 canvas.
- Мои услуги: Таблица с описанием услуг и их ценами.
- Контакт: Форма обратной связи для потенциальных клиентов.

3. Footer: Низ страницы с информацией о правах и ссылками на социальные сети.

Также был создан 1 файл формата .css, с названием style, для использования стилей на странице, а также импортирован файл reset.css, содержащий в себе стили “по умолчанию” для разных элементов разметки. В проекте сначала активизируется reset.css, задаёт изначальные значения стиля для всех элементов, после чего загружается style.css, где уже расписаны “кастомные” изменения от автора, содержащий в себе адаптив для телефонов.

Использование JS

При разработке игры "Змейка" на JavaScript, я следовал определенной логике и структуре, чтобы создать функциональное и интерактивное приложение. Давайте поэтапно разберем, как я подошел к написанию этого кода и какие соображения были в процессе.

| 1. Определение класса SnakePart Первым шагом было создание класса SnakePart, который представляет собой отдельный сегмент змеи. Этот класс имеет конструктор, принимающий координаты x и y, которые будут использоваться для определения положения сегмента на игровом поле.

```
class SnakePart {  
    constructor(x, y) { this.x = x; this.y = y; }  
}
```

| 2. Инициализация игровых переменных Далее я перешел к инициализации основных переменных игры. Я выбрал элемент <canvas> из HTML-документа, который будет служить игровым полем. Затем установил параметры, такие как скорость змеи, количество тайлов (квадратов), размер тайла и начальные координаты головы змеи.

```
canvas = document.getElementById("game");  
speed = 10;  
tileCount = 20;  
tileSize = canvas.width / tileCount - 2;  
headX = 10; headY = 10;
```

Эти переменные позволяют управлять размером игрового поля и поведением змеи.

| 3. Определение состояния игры Я создал массив snakeParts, который будет хранить все сегменты змеи, и переменные для отслеживания длины хвоста, координат яблока, а также текущего счета.

Также определил переменные для управления скоростью и направлениями движения змеи.

```
snakeParts = [];  
tailLength = 2;  
appleX = 5;  
appleY = 5;  
inputsXVelocity = 0;  
inputsYVelocity = 0;
```

| 4. Основной цикл игры Функция update() является основным циклом игры. Она отвечает за обновление состояния игры, проверку условий окончания игры, очистку экрана и отрисовку всех элементов (змеи, яблока, счета). В начале функции я устанавливаю текущую скорость и направление движения змеи.

```
function update() {  
    xVelocity = inputsXVelocity;  
    yVelocity = inputsYVelocity;  
}
```

| 5. Проверка окончания игры В функции isGameOver() я проверяю условия окончания игры: если голова змеи выходит за пределы игрового поля или сталкивается с собственным телом. Если игра окончена, я отображаю сообщение "Game Over!" на экране.

```
if (headX < 0 || headX === tileCount || headY < 0 || headY === tileCount) {  
    gameOver = true;  
}
```

| 6. Очистка экрана и отрисовка элементов Функция clearScreen() отвечает за очистку игрового поля перед каждой отрисовкой. Я использую черный цвет для фона:

```
function clearScreen() {  
    ctx.fillStyle = "black";  
    ctx.fillRect(0, 0, canvas.width, canvas.height);  
}
```

Затем в функции drawSnake() я отрисовываю змею, проходя по массиву snakeParts и рисуя каждый сегмент. Я также добавляю новый сегмент в массив на каждой итерации цикла и удаляю старый сегмент, если длина хвоста превышает заданную.

```
function drawSnake() {  
    ctx.fillStyle = "green";  
    for (let i = 0; i < snakeParts.length; i++) {  
        let part = snakeParts[i]; ctx.fillRect(part.x * tileCount, part.y * tileCount, tileSize,  
        tileSize);  
    }  
}
```

| 7. Обработка столкновения с яблоком Функция checkAppleCollision() проверяет, съела ли змея яблоко. Если да, то новые координаты яблока генерируются случайным образом, увеличивается длина хвоста и счет игрока.

```
if (appleX === headX && appleY == headY) {  
    appleX = Math.floor(Math.random() * tileCount);  
    appleY = Math.floor(Math.random() * tileCount);  
    tailLength++;  
    score++;  
}
```

| 8. Управление движением змеи Для управления движением змеи я использую обработчик событий keydown. В зависимости от нажатой клавиши (стрелки или W/A/S/D) устанавливаются соответствующие значения для переменных направления.

```
document.body.addEventListener("keydown", keyDown);  
function keyDown(event) {  
    if (event.keyCode == 38 || event.keyCode == 87) {  
        // up if (inputsYVelocity == 1) return; inputsYVelocity = -1; inputsXVelocity = 0;  
    }  
}
```

Вывод

Я делал проект на протяжении 2 месяцев.

За проект было усвоено множество знаний, которые были продемонстрированы в нём, в копилку знаний добавились: язык разметки HTML, язык стилей CSS, а также язык программирования JavaScript, опыт работы с IDE – Visual Studio Code.

Используемый список сайтов

<https://journal.top-academy.ru/ru/auth/login/index> - учебные материалы

<https://stackoverflow.com>

<https://releases.jquery.com>

<https://webdesign.tutsplus.com/ru/html5-form-validation-with-the-pattern-attribute--cms-25145t>

<https://habr.com/ru/articles/>

<https://learn.javascript.ru/first-steps>