

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа №3
CSS-переменные, темизация сайта

Выполнил:
Чернышев Михаил Павлович
Группа К3441

Проверил:
Добряков Д. И.

Санкт-Петербург
2026 г.

Содержание

1	Задача	2
2	Теоретическая часть	2
2.1	CSS Custom Properties (переменные)	2
2.2	prefers-color-scheme	2
3	Ход работы	4
3.1	Определение CSS-переменных	4
3.2	Светлая тема (по умолчанию)	5
3.3	Тёмная тема	6
3.4	Применение переменных в компонентах	7
3.5	Автоматическое определение системной темы	9
3.6	Переключатель темы (HTML)	9
3.7	JavaScript для переключения темы	11
3.8	Предотвращение мерцания при загрузке	13
3.9	Дополнительные улучшения	13
3.10	Адаптация изображений для тёмной темы	14
4	Вывод	14

1 Задача

Выполнить темизацию сайта сервиса аренды недвижимости с использованием CSS-переменных. Реализовать светлую и тёмную темы с учётом пользовательских системных настроек и переключателем через JavaScript.

2 Теоретическая часть

2.1 CSS Custom Properties (переменные)

CSS-переменные (Custom Properties) позволяют определять значения, которые можно переиспользовать по всему документу:

```
:root {
  --primary-color: #0d6efd;
  --font-size-base: 16px;
}

.button {
  background: var(--primary-color);
  font-size: var(--font-size-base);
}
```

Преимущества:

- Каскадность и наследование
- Изменение в runtime через JavaScript
- Возможность переопределения в медиа-запросах
- Поддержка fallback-значений: `var(-color, #000)`

2.2 prefers-color-scheme

Медиа-запрос для определения системной темы пользователя:

```
@media (prefers-color-scheme: dark) {
  :root {
    --bg-color: #1a1a1a;
    --text-color: #ffffff;
  }
}

@media (prefers-color-scheme: light) {
  :root {
    --bg-color: #ffffff;
  }
}
```

```
    --text-color: #1a1a1a;  
  }  
}
```

3 Ход работы

3.1 Определение CSS-переменных

```
styles/variables.css
:root {
  /* Цвета бренда (не меняются) */
  --brand-primary: #0d6efd;
  --brand-primary-hover: #0b5ed7;
  --brand-success: #198754;
  --brand-danger: #dc3545;
  --brand-warning: #ffc107;

  /* Размеры и отступы */
  --spacing-xs: 4px;
  --spacing-sm: 8px;
  --spacing-md: 16px;
  --spacing-lg: 24px;
  --spacing-xl: 32px;

  /* Типографика */
  --font-family: 'Segoe UI', system-ui, sans-serif;
  --font-size-sm: 14px;
  --font-size-base: 16px;
  --font-size-lg: 18px;
  --font-size-xl: 24px;
  --font-size-2xl: 32px;

  /* Скругления */
  --radius-sm: 4px;
  --radius-md: 8px;
  --radius-lg: 16px;
  --radius-full: 9999px;

  /* Тени */
  --shadow-sm: 0 1px 2px rgba(0, 0, 0, 0.05);
  --shadow-md: 0 4px 6px rgba(0, 0, 0, 0.1);
  --shadow-lg: 0 10px 15px rgba(0, 0, 0, 0.1);

  /* Анимации */
  --transition-fast: 150ms ease;
  --transition-base: 300ms ease;
  --transition-slow: 500ms ease;
}
```

3.2 Светлая тема (по умолчанию)

styles/theme-light.css

```
:root,
[data-theme="light"] {
  /* Фоны */
  --bg-primary: #ffffff;
  --bg-secondary: #f8f9fa;
  --bg-tertiary: #e9ecef;
  --bg-inverse: #212529;

  /* Текст */
  --text-primary: #212529;
  --text-secondary: #6c757d;
  --text-muted: #adb5bd;
  --text-inverse: #ffffff;

  /* Границы */
  --border-color: #dee2e6;
  --border-color-light: #e9ecef;

  /* Карточки */
  --card-bg: #ffffff;
  --card-shadow: var(--shadow-md);
  --card-border: 1px solid var(--border-color-light);

  /* Инпуты */
  --input-bg: #ffffff;
  --input-border: #ced4da;
  --input-focus-border: var(--brand-primary);
  --input-placeholder: #6c757d;

  /* Навигация */
  --navbar-bg: var(--brand-primary);
  --navbar-text: #ffffff;

  /* Футер */
  --footer-bg: #212529;
  --footer-text: #adb5bd;

  /* Модальные окна */
  --modal-overlay: rgba(0, 0, 0, 0.5);
  --modal-bg: #ffffff;

  /* Скроллбар */
  --scrollbar-track: #f1f1f1;
  --scrollbar-thumb: #c1c1c1;
  --scrollbar-thumb-hover: #a1a1a1;
}
```

3.3 Тёмная тема

styles/theme-dark.css

```
[data-theme="dark"] {  
  /* Фоны */  
  --bg-primary: #121212;  
  --bg-secondary: #1e1e1e;  
  --bg-tertiary: #2d2d2d;  
  --bg-inverse: #ffffff;  
  
  /* Текст */  
  --text-primary: #e4e4e4;  
  --text-secondary: #a0a0a0;  
  --text-muted: #6c6c6c;  
  --text-inverse: #121212;  
  
  /* Границы */  
  --border-color: #3d3d3d;  
  --border-color-light: #2d2d2d;  
  
  /* Карточки */  
  --card-bg: #1e1e1e;  
  --card-shadow: 0 4px 6px rgba(0, 0, 0, 0.3);  
  --card-border: 1px solid var(--border-color);  
  
  /* Инпуты */  
  --input-bg: #2d2d2d;  
  --input-border: #3d3d3d;  
  --input-focus-border: var(--brand-primary);  
  --input-placeholder: #6c6c6c;  
  
  /* Навигация */  
  --navbar-bg: #1e1e1e;  
  --navbar-text: #e4e4e4;  
  
  /* Футер */  
  --footer-bg: #0d0d0d;  
  --footer-text: #6c6c6c;  
  
  /* Модальные окна */  
  --modal-overlay: rgba(0, 0, 0, 0.8);  
  --modal-bg: #1e1e1e;  
  
  /* Скроллбар */  
  --scrollbar-track: #1e1e1e;  
  --scrollbar-thumb: #3d3d3d;  
  --scrollbar-thumb-hover: #4d4d4d;  
  
  /* Корректировка теней для тёмной темы */  
  --shadow-sm: 0 1px 2px rgba(0, 0, 0, 0.3);  
  --shadow-md: 0 4px 6px rgba(0, 0, 0, 0.4);  
  --shadow-lg: 0 10px 15px rgba(0, 0, 0, 0.5);  
}
```

3.4 Применение переменных в компонентах

```
styles/components.css

/* Базовые стили */
body {
  font-family: var(--font-family);
  font-size: var(--font-size-base);
  background-color: var(--bg-secondary);
  color: var(--text-primary);
  transition: background-color var(--transition-base),
              color var(--transition-base);
}

/* Карточка */
.card {
  background: var(--card-bg);
  border: var(--card-border);
  border-radius: var(--radius-md);
  box-shadow: var(--card-shadow);
  transition: background var(--transition-base),
              box-shadow var(--transition-base);
}

.card:hover {
  box-shadow: var(--shadow-lg);
}

/* Кнопка */
.btn-primary {
  background: var(--brand-primary);
  color: var(--text-inverse);
  border: none;
  padding: var(--spacing-sm) var(--spacing-md);
  border-radius: var(--radius-md);
  transition: background var(--transition-fast);
}

.btn-primary:hover {
  background: var(--brand-primary-hover);
}

/* Инпут */
.form-control {
  background: var(--input-bg);
  border: 1px solid var(--input-border);
  color: var(--text-primary);
  padding: var(--spacing-sm) var(--spacing-md);
  border-radius: var(--radius-md);
  transition: border-color var(--transition-fast),
              background var(--transition-base);
}

.form-control:focus {
```



```
border-color: var(--input-focus-border);
outline: none;
box-shadow: 0 0 0 3px rgba(13, 110, 253, 0.25);
}

.form-control::placeholder {
color: var(--input-placeholder);
}

/* Навигация */
.navbar {
background: var(--navbar-bg);
color: var(--navbar-text);
transition: background var(--transition-base);
}

/* Текстовые стили */
.text-secondary {
color: var(--text-secondary);
}

.text-muted {
color: var(--text-muted);
}
```

3.5 Автоматическое определение системной темы

Медиа-запрос prefers-color-scheme

```
/* Автоматическое применение тёмной темы */
@media (prefers-color-scheme: dark) {
  :root:not([data-theme="light"]) {
    /* Все переменные тёмной темы */
    --bg-primary: #121212;
    --bg-secondary: #1e1e1e;
    --text-primary: #e4e4e4;
    /* ... остальные переменные ... */
  }
}
```

3.6 Переключатель темы (HTML)

HTML компонент переключателя

```
<button class="theme-toggle"
  id="themeToggle"
  aria-label="Переключить тему"
  aria-pressed="false">
  <span class="theme-toggle__icon theme-toggle__icon--light">
    <i class="bi bi-sun-fill" aria-hidden="true"></i>
  </span>
  <span class="theme-toggle__icon theme-toggle__icon--dark">
    <i class="bi bi-moon-fill" aria-hidden="true"></i>
  </span>
</button>
```

CSS для переключателя

```
.theme-toggle {
  background: var(--bg-tertiary);
  border: 1px solid var(--border-color);
  border-radius: var(--radius-full);
  padding: var(--spacing-sm);
  cursor: pointer;
  position: relative;
  width: 44px;
  height: 44px;
  display: flex;
  align-items: center;
  justify-content: center;
  transition: background var(--transition-fast);
}

.theme-toggle:hover {
  background: var(--bg-inverse);
  color: var(--text-inverse);
}

/* Показываем нужную иконку */
[data-theme="light"] .theme-toggle__icon--dark,
:root:not([data-theme]) .theme-toggle__icon--dark {
```

```
    display: none;
}

[data-theme="dark"] .theme-toggle__icon--light {
    display: none;
}

[data-theme="dark"] .theme-toggle__icon--dark {
    display: block;
}
```

3.7 JavaScript для переключения темы

```
js/theme.js

class ThemeManager {
  constructor() {
    this.STORAGE_KEY = 'theme';
    this.THEMES = { LIGHT: 'light', DARK: 'dark' };

    this.init();
  }

  init() {
    // Применяем сохранённую тему или системную
    const savedTheme = this.getSavedTheme();
    const systemTheme = this.getSystemTheme();

    this.setTheme(savedTheme || systemTheme);

    // Слушаем изменения системной темы
    this.watchSystemTheme();

    // Инициализируем переключатель
    this.initToggle();
  }

  getSystemTheme() {
    if (window.matchMedia('(prefers-color-scheme: dark)').matches) {
      return this.THEMES.DARK;
    }
    return this.THEMES.LIGHT;
  }

  getSavedTheme() {
    return localStorage.getItem(this.STORAGE_KEY);
  }

  saveTheme(theme) {
    localStorage.setItem(this.STORAGE_KEY, theme);
  }

  setTheme(theme) {
    document.documentElement.setAttribute('data-theme', theme);
    this.currentTheme = theme;

    // Обновляем aria-pressed
    const toggle = document.getElementById('themeToggle');
    if (toggle) {
      toggle.setAttribute('aria-pressed',
        theme === this.THEMES.DARK ? 'true' : 'false');
    }

    // Обновляем meta theme-color
    const metaThemeColor = document.querySelector('meta[name="theme-color"]');
  }
}
```

```

        if (metaThemeColor) {
            metaThemeColor.setAttribute('content',
                theme === this.THEMES.DARK ? '#1e1e1e' : '#0d6efd');
        }
    }

    toggleTheme() {
        const newTheme = this.currentTheme === this.THEMES.LIGHT
            ? this.THEMES.DARK
            : this.THEMES.LIGHT;

        this.setTheme(newTheme);
        this.saveTheme(newTheme);
    }

    watchSystemTheme() {
        const mediaQuery = window.matchMedia('(prefers-color-scheme: dark)');

        mediaQuery.addEventListener('change', (e) => {
            // Применяем только если пользователь не выбрал тему вручную
            if (!this.getSavedTheme()) {
                this.setTheme(e.matches ? this.THEMES.DARK : this.THEMES.LIGHT);
            }
        });
    }

    initToggle() {
        const toggle = document.getElementById('themeToggle');
        if (toggle) {
            toggle.addEventListener('click', () => this.toggleTheme());
        }
    }
}

// Инициализация
document.addEventListener('DOMContentLoaded', () => {
    window.themeManager = new ThemeManager();
});

```

3.8 Предотвращение мерцания при загрузке

Скрипт в head (inline)

```
<head>
  <!-- Применяем тему до рендеринга страницы -->
  <script>
    (function() {
      const theme = localStorage.getItem('theme');
      if (theme) {
        document.documentElement.setAttribute('data-theme', theme);
      } else if (window.matchMedia('(prefers-color-scheme: dark)').matches) {
        document.documentElement.setAttribute('data-theme', 'dark');
      }
    })();
  </script>

  <!-- CSS с переменными -->
  <link rel="stylesheet" href="styles/variables.css">
  <link rel="stylesheet" href="styles/theme-light.css">
  <link rel="stylesheet" href="styles/theme-dark.css">
</head>
```

3.9 Дополнительные улучшения

Плавная анимация перехода между темами

```
/* Глобальная анимация при смене темы */
html.theme-transition,
html.theme-transition *,
html.theme-transition *::before,
html.theme-transition *::after {
  transition: background-color 0.3s ease,
              color 0.3s ease,
              border-color 0.3s ease,
              box-shadow 0.3s ease !important;
}
```

JS для анимации перехода

```
toggleTheme() {
  // Добавляем класс для анимации
  document.documentElement.classList.add('theme-transition');

  const newTheme = this.currentTheme === this.THEMES.LIGHT
    ? this.THEMES.DARK
    : this.THEMES.LIGHT;

  this.setTheme(newTheme);
  this.saveTheme(newTheme);

  // Удаляем класс после анимации
  setTimeout(() => {
    document.documentElement.classList.remove('theme-transition');
  }, 300);
}
```

3.10 Адаптация изображений для тёмной темы

Адаптивные изображения

```
/* Уменьшаем яркость изображений в тёмной теме */
[data-theme="dark"] img:not(.no-dim) {
  filter: brightness(0.9);
}

/* Инвертируем логотип */
[data-theme="dark"] .logo-light {
  display: none;
}

[data-theme="dark"] .logo-dark {
  display: block;
}

[data-theme="light"] .logo-dark,
:root:not([data-theme]) .logo-dark {
  display: none;
}
```

HTML с разными версиями логотипа

```
<picture>
  <source srcset="logo-dark.svg" media="(prefers-color-scheme: dark)">
  
</picture>
```

4 Вывод

В ходе выполнения домашней работы была реализована темизация сайта с использованием CSS-переменных.

Реализованный функционал:

- Определены CSS-переменные для всех цветов, размеров и теней
- Созданы светлая и тёмная темы
- Автоматическое определение системной темы через `prefers-color-scheme`
- Переключатель темы с сохранением в `localStorage`
- Плавная анимация при переключении
- Предотвращение мерцания при загрузке страницы
- Адаптация изображений для тёмной темы

Использованные технологии:

- CSS Custom Properties (переменные)

- CSS медиа-запрос `prefers-color-scheme`
- data-атрибуты для переключения темы
- `localStorage` для сохранения выбора пользователя
- JavaScript класс `ThemeManager`

Результат: сайт поддерживает светлую и тёмную темы, автоматически определяет системные настройки пользователя и позволяет переключать тему вручную с сохранением выбора.