Лабораторная работа №2

студента группы ИТ-212

Устинова Андрея Олеговича

ДИНАМИКА НА ВЕБ-СТРАНИЦАХ. ЯЗЫК ПРОГРАММИРОВАНИЯ JAVASCRIPT

Выполнение\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Защита\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Цель работы: ознакомиться с основными возможностями языка JavaScript, синтаксисом, встроенными объектами, событиями DHTML, получить практические навыки программирования на языке JavaScript. Получить практические навыки создания интерактивных веб-страниц.

Содержание работы

1. Изменить web-сайт, разработанный в лабораторной работе 1, добавив динамические эффекты на страницах при помощи средств JavaScript и DHTML.
2. Украсить свой сайт различными эффектами (параметры текста, замена изображения, динамическое изменение картинки, изменения прозрачности и других css-свойств, всплывающие подсказки и т.д.) по наступлению различных событий.
3. Сделать выпадающее меню. Допускается использование готовых решений.
4. *Произвести проверку данных, введенных в поля формы (на заполнение, на правильность email, на ввод только числового значения и т.д.) средствами JavaScript, не используя новые атрибуты html5 .Применить регулярные выражения.*
5. *В задании обязательно использовать таймер (осуществлять загрузку страницы через некоторый промежуток времени, часы в тексте и т.д.), встроенные объекты Date, Math, String и др.*
6. Для реализации динамических эффектов создать пользовательские функции во внешних файлах скриптов.
7. В задании использовать 5-*10* различных событий, сделав на каждое из них отдельный обработчик.
8. Использовать как внутренние (инлайновые) скрипты, так и внешние.
9. Страницы должны быть работоспособны в любом браузере.

Ход работы

1. Ознакомился с языком JavaScript. После приступил к написанию собственных скриптов.
2. Создал отдельно файлы скриптов, в которых будут храниться различные скрипты сайта.
3. Приступил к написанию скриптов:

* Скрипт фиксирования хэдера при скролле (Рисунок 1):

// Header при скролле фиксируется

var headerPlaceholder = document.getElementById("header-placeholder");

var header = document.getElementById("header-container");

var scrollThreshold = 48;

window.addEventListener(

  "scroll",

  function () {

    var currentScroll = window.scrollY || window.pageYOffset;

    if (currentScroll > scrollThreshold) {

      // Прокрутка вниз

      headerPlaceholder.style.display = "block";

      header.classList.add("fixed");

    } else if (currentScroll <= scrollThreshold) {

      // Прокрутка вверх

      headerPlaceholder.style.display = "none";

      header.classList.remove("fixed");

    }

  },

  false

);

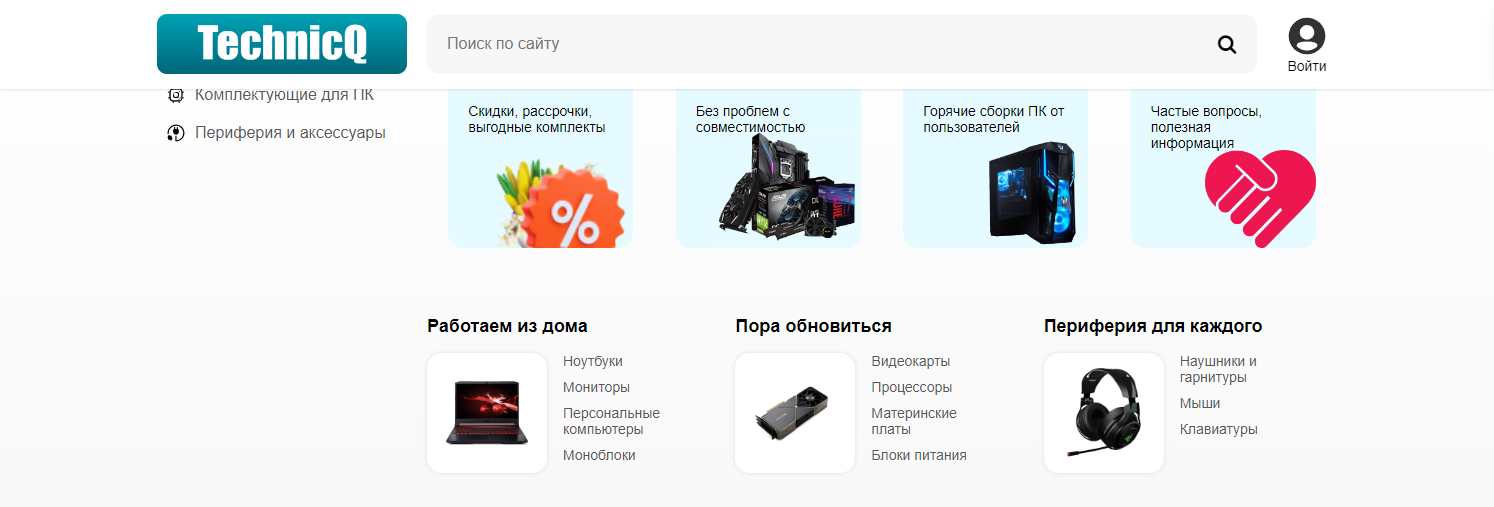


Рисунок 1 Фиксирование хэдера

* Скрипт карусели карточек брендов (Рисунок 2):

// Карусель

const track = document.querySelector(".carousel-track");

const items = document.querySelectorAll(".carousel-item");

const prevButton = document.querySelector(".prev");

const nextButton = document.querySelector(".next");

let index = 0;

const itemWidth = items[0].offsetWidth;

const visibleItems = Math.floor(track.offsetWidth / itemWidth); // Количество видимых элементов

prevButton.addEventListener("click", () => {

  if (index > 0) {

    index = Math.max(index - 1, 0);

    updateTrack();

    updateButtonsState();

  }

});

nextButton.addEventListener("click", () => {

  if (index < items.length - 5) {

    index = Math.min(index + 1, items.length - visibleItems);

    updateTrack();

    updateButtonsState();

  }

});

function updateTrack() {

  const offsetX = -index \* itemWidth;

  track.style.transform = `translateX(${offsetX}px)`;

}

function updateButtonsState() {

  if (index === 0) {

    prevButton.classList.add("disabled");

  } else {

    prevButton.classList.remove("disabled");

  }

  if (index >= items.length - visibleItems) {

    nextButton.classList.add("disabled");

  } else {

    nextButton.classList.remove("disabled");

  }

}

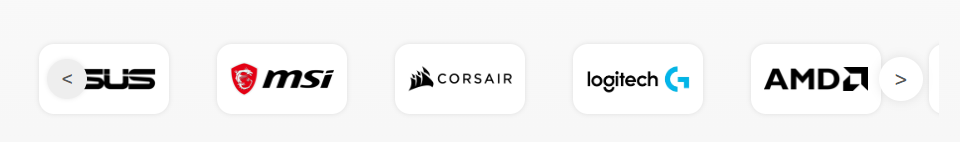


Рисунок 2 Карусель карточек брендов

* Скрипт, позволяющий при нажатии на кнопку аккаунта перейти в окно входа. В этом окне входа можно переключить режим отображения пароля (скрыт или виден):

const modal = document.getElementById("modal");

const user = document.getElementById("user");

const emailInput = document.getElementById("e-mail");

const passwordInput = document.getElementById("password");

const emailErrorMessage = document.getElementById("email-error-message");

const passwordErrorMessage = document.getElementById("password-error-message");

const loginButton = document.querySelector(".login-button");

const togglePasswordButton = document.getElementById("togglePassword");

const closeModalBtn = document.querySelector(".close");

// Открытие модального окна при клике на элемент

user.addEventListener("click", function (event) {

  modal.style.display = "block";

  event.stopPropagation(); // Предотвращение дальнейшего распространения события

});

// Закрытие модального окна при клике на кнопку закрытия

closeModalBtn.addEventListener("click", function (event) {

  modal.style.display = "none";

  resetForm(); // Обнуляем содержимое формы при закрытии

  event.stopPropagation(); // Предотвращение дальнейшего распространения события

});

// Закрытие модального окна при клике вне modal-content

modal.addEventListener("click", function (event) {

  if (

    event.target.closest(".modal") &&

    !event.target.closest(".modal-content")

  ) {

    modal.style.display = "none";

    resetForm(); // Обнуляем содержимое формы при закрытии

    event.stopPropagation(); // Предотвращение дальнейшего распространения события

  }

});

function toLowercase(input) {

  input.value = input.value.toLowerCase();

}

// Скрытие/раскрытие пароля

togglePasswordButton.addEventListener("click", function () {

  if (passwordInput.type === "password") {

    passwordInput.type = "text";

    togglePasswordButton.innerHTML =

      '<i class="fa fa-eye-slash" style="font-size: 20px"></i>';

  } else {

    passwordInput.type = "password";

    togglePasswordButton.innerHTML =

      '<i class="fa fa-eye" style="font-size: 20px"></i>';

  }

});

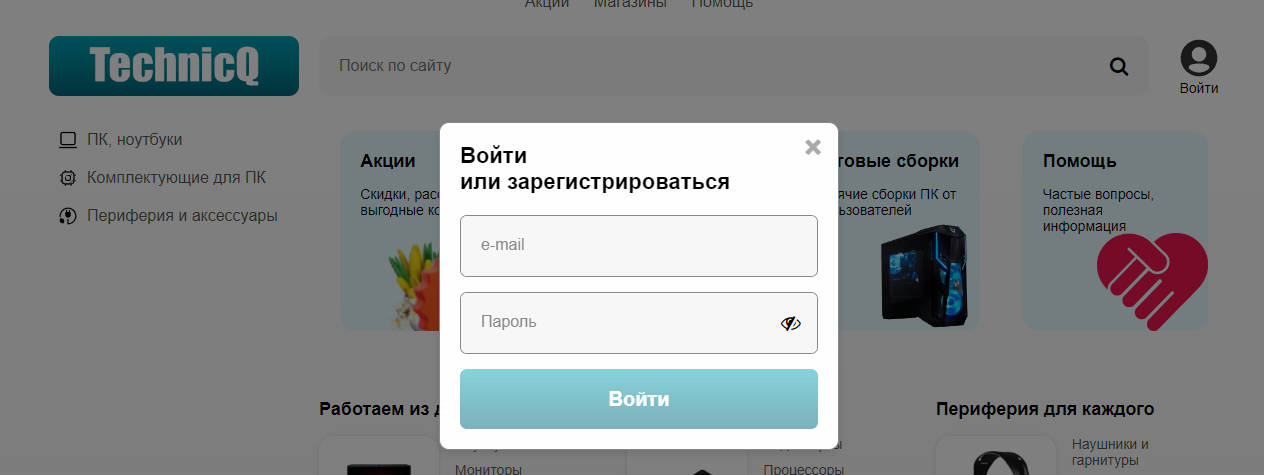


Рисунок 3 Окно входа

В этом же скрипте есть проверка правильности ввода почты и пароля. При правильных данных кнопка входа станет активной. В данном скрипте используется таймер для корректной работы (Рисунок 4, Рисунок 5):

const emailRegex = /^[a-z0-9](?:[a-z0-9.\_-]\*[a-z0-9])?@[a-z]+\.[a-z]{2,}$/;

const passwordRegex = /^[a-zA-Z0-9!@#$%^&\*()\_+{}:;<>,\.?\-].{7,}$/;

function checkInputs() {

  if (!emailRegex.test(emailInput.value)) {

    emailErrorMessage.style.display = "block";

  } else {

    emailErrorMessage.style.display = "none";

  }

  if (!passwordRegex.test(passwordInput.value)) {

    passwordErrorMessage.style.display = "block";

  } else {

    passwordErrorMessage.style.display = "none";

  }

  if (

    emailRegex.test(emailInput.value) &&

    passwordRegex.test(passwordInput.value)

  ) {

    loginButton.classList.add("active");

    loginButton.disabled = false;

  } else {

    loginButton.classList.remove("active");

    loginButton.disabled = true;

  }

}

// Проверка почты и пароля при изменении значения поля e-mail

emailInput.addEventListener("blur", function () {

  checkInputs();

});

// Проверка почты и пароля при изменении значения поля пароля

passwordInput.addEventListener("blur", function () {

  checkInputs();

});

// Функция для обнуления содержимого формы

function resetForm() {

  emailInput.value = "";

  passwordInput.value = "";

  emailErrorMessage.style.display = "none";

  passwordErrorMessage.style.display = "none";

  setTimeout(function () {

    passwordInput.type = "password";

    togglePasswordButton.innerHTML =

      '<i class="fa fa-eye" style="font-size: 20px"></i>';

    loginButton.classList.remove("active");

    loginButton.disabled = true;

  }, 100);

}

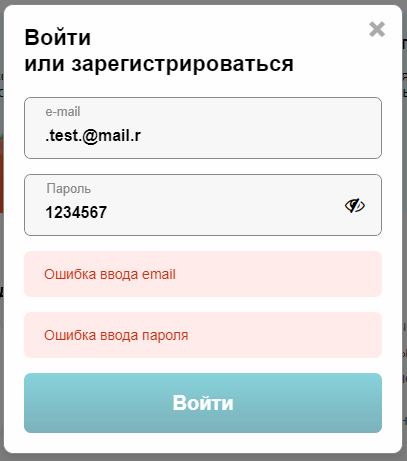


Рисунок 4 Неверный ввод

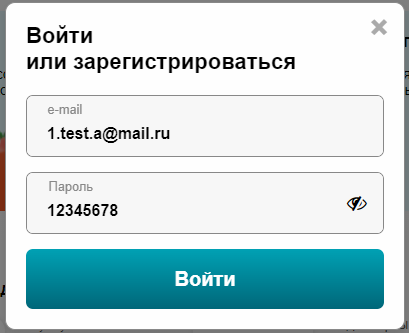


Рисунок 5 Верный ввод

* Скрипт для выпадающего меню и открытия его при переходе по ссылке на конкретно этот пункт (Рисунок 6, Рисунок 7):

// Выпадающее меню со сдвигом (аккордеон)

document.addEventListener("DOMContentLoaded", function () {

  // Код для аккордеона

  var helpCards = document.querySelectorAll(".help-card");

  helpCards.forEach(function (helpCard) {

    var h4 = helpCard.querySelector("h4");

    h4.addEventListener("click", function () {

      var accordion = helpCard.querySelector(".accordion");

      var arrow = h4.querySelector(".arrow");

      if (accordion.classList.contains("active")) {

        accordion.classList.remove("active");

        helpCard.style.height = "60px";

        arrow.classList.remove("opened");

        h4.classList.remove("opened");

        accordion.classList.add("closing"); // для моментального закрытия без анимации

      } else {

        accordion.classList.remove("closing");

        accordion.classList.add("active");

        helpCard.style.height = "auto";

        arrow.classList.add("opened");

        h4.classList.add("opened");

      }

    });

  });

  // Проверка хэша URL и открытие соответствующего аккордеона

  var hash = window.location.hash;

  if (hash) {

    var helpCard = document.getElementById(hash.substring(1));

    if (helpCard && helpCard.classList.contains("help-card")) {

      var accordion = helpCard.querySelector(".accordion");

      var arrow = helpCard.querySelector("h4 .arrow");

      if (accordion && arrow) {

        accordion.classList.add("active");

        helpCard.style.height = "auto";

        arrow.classList.add("opened");

        arrow.parentElement.classList.add("opened");

      }

    }

  }

});

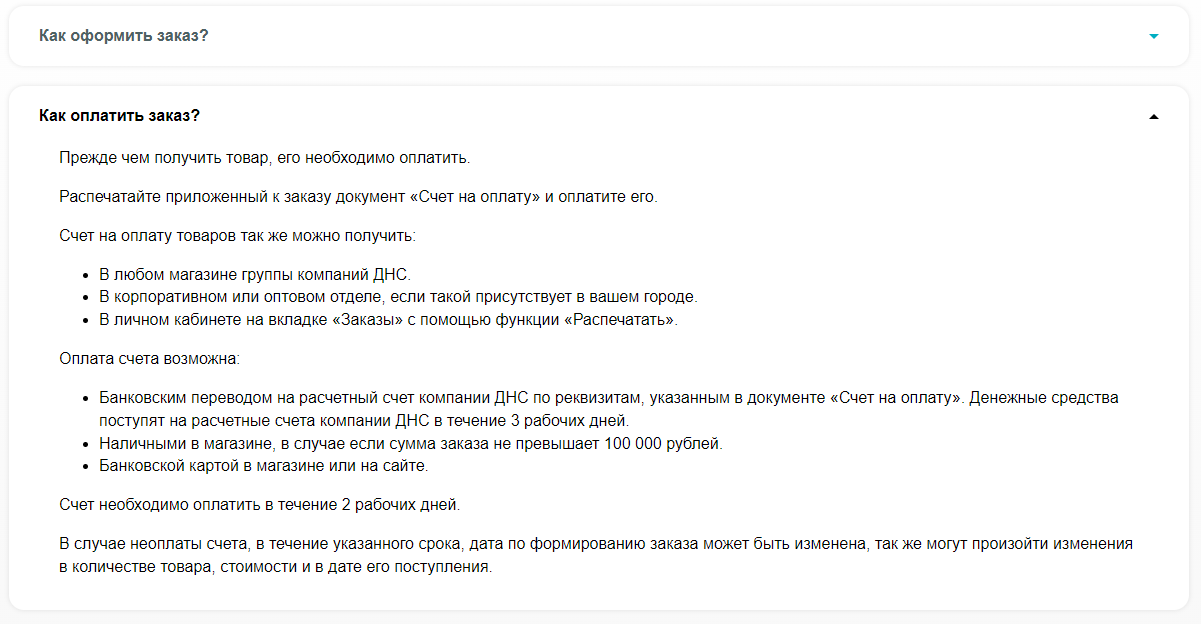


Рисунок 6 Текущая дата

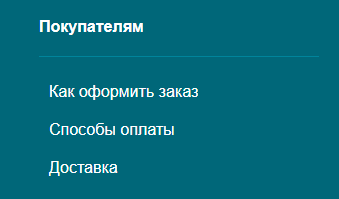


Рисунок 7 Ссылки, при переходе по которым нужное выпадающее меню будет сразу открыто

* Скрипт карты (Рисунок 8):

<script

          type="text/javascript"

          charset="utf-8"

          async

          src="https://api-maps.yandex.ru/services/constructor/1.0/js/?um=constructor%3A01409a394611156f49be55c041aa4ec306cd27ee00799edeb8f184c80ea1639a&amp;width=512&amp;height=512&amp;lang=ru\_RU&amp;scroll=true"

        ></script>

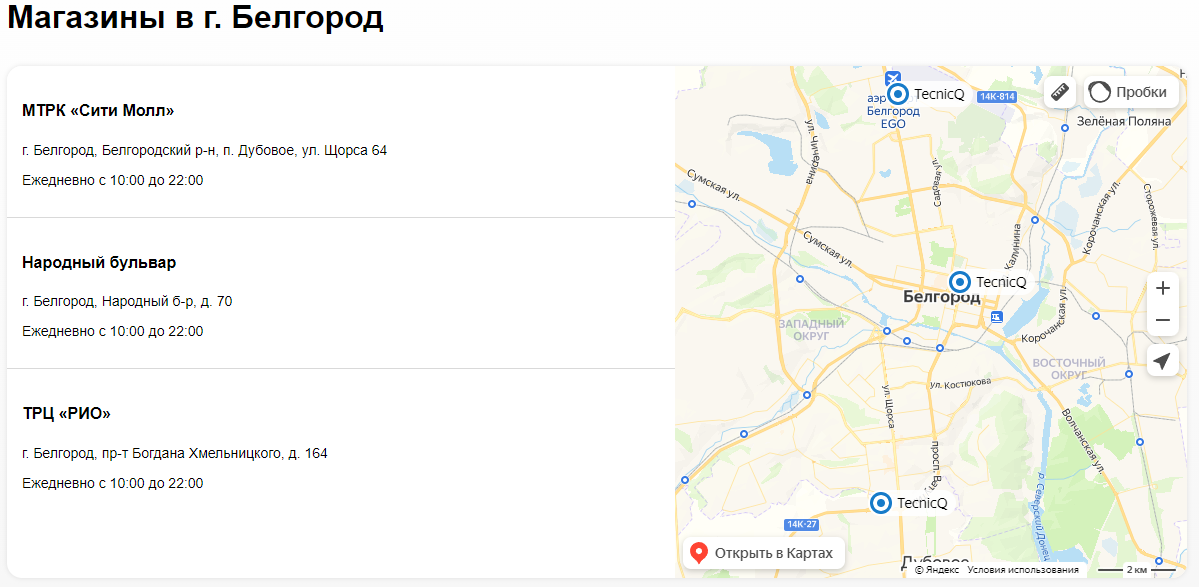


Рисунок 8 Карта

* Скрипт переливания логотипа при наведении (Рисунок 9):

let hoverInterval;

const logo = document.querySelector(".logo");

const initialGradient = window

  .getComputedStyle(logo)

  .getPropertyValue("background-image");

function startGradient() {

  let degree = 0;

  hoverInterval = setInterval(() => {

    degree = (degree + 1) % 360;

    const color1 = `hsl(${degree + 197}, 100%, 24%)`;

    const color2 = `hsl(${(degree + 177) % 360}, 89%, 46%)`;

    logo.style.backgroundImage = `linear-gradient(to top, ${color1}, ${color2})`;

  }, 50);

}

function stopGradient() {

  clearInterval(hoverInterval);

  logo.style.backgroundImage = initialGradient;

}





Рисунок 9 Переливание логотипа

* Скрипт выделения поиска при вводе (Рисунок 10):

const input = document.querySelector(".search-container input");

const searchContainer = document.querySelector(".search-container");

input.addEventListener("focus", function () {

  searchContainer.classList.add("focused");

});

input.addEventListener("blur", function () {

  searchContainer.classList.remove("focused");

});

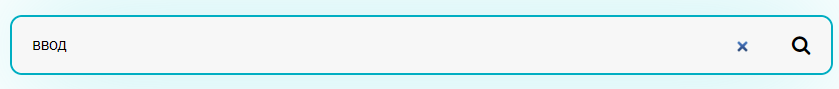


Рисунок 10 Выделение поиска