Cookies и Web Storage

№ урока:	17	**Курс:**	JavaScript Базовый

Средства обучения:	Visual Studio Code Web Browser

Обзор, цель и назначение урока

Научиться сохранять данные в браузере используя cookie и We Storage API.

Изучив материал данного занятия, учащийся сможет:

- Понимать, что такое cookie.
- Использовать методы encodeURIComponent и decodeURIComponent.
- Работать с cookie страницы через JavaScript код.
- Использовать localStorage и sessionStorage для хранения данных.
- Сериализовывать и десериализовывать данные в JSON формат

Содержание урока

1. Что такое cookie
2. Работа с cookie через JavaScript код
3. Обзор Web Storage API
4. sessionStorage и localStorage

Резюме

- Cookie – небольшой фрагмент данных, отправляемый веб сервером, который сохраняется на стороне клиента. Веб браузер отправляет cookie каждый раз, когда выполняет запрос на сайт, с которого был получен cookie.

- Доступ к cookie, через JavaScript, можно получить с помощью свойства document.cookie.

- Часто cookie используются для авторизации пользователя. Когда пользователь заходит на закрытую часть сайта, сервер проверяет наличие специального зашифрованного значения, которое хранится в cookie. Если этого значения нет, то пользователя перенаправляют на страницу входа на сайт, если значение есть – показывают содержимое закрытой части сайта.
  Если пользователь, попав на страницу входа, вводит правильный логин и пароль (пользователь с таким логином и паролем находится в базе), тогда сервер возвращает пользователю cookie авторизации, которые сохраняется в памяти браузера. При каждом последующем запросе на сервер, этот cookie будет частью запроса и сервер будет понимать, что пользователь выполнил вход на сайт, и что ему можно отображать закрытые страницы.

- Cookie предается на севре как HTTP заголовок.

- Значение одного cookie не должно превышать 4Кб и на один домен, большинство браузеров, разрешают создавать около 20 cookie.

CyberBionic systematics

CyberBionic Systematics ®
19 Evgenia Sverstiuka Str., 5 floor
Kyiv, Ukraine

t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- Cookie не предназначены для хранения больших данных, и больше подходят для отслеживания идентификатора пользователя, при переходах по разным страницам сайта или сохранению незначительных блоков данных.

- Cookie представляется строкой состоящее из ключей и значений, разделенных точкой с запятой.

- Атрибуты cookie:
  path - указывает страницы на которых будет работать данный cookie
  /catalog на страницах /catalog/folder и т.д., но на странице /home работать не будет
  / - для всех страниц
  max-age - время жизни значения в секундах
  expires - дата, когда значение должно быть удалено, пример значения 20 May 2022 05:15:05 GMT. Для удаления cookie необходимо установить прошедшую дату или max-age=0
  samesite - значения strict, lax или none. Настройка необходима для защиты от уязвимости CSRF. Атрибут не поддерживается старыми браузерами (до 2017 года).
  secure - cookie можно отправлять только по HTTPS
  httpOnly - cookie используются только для HTTP запросов и не могут быть получены с помощью JS кода

  Пример создания cookie с параметрами:
  document.cookie = "color=green; path=/; max-age=30;" – название color, значение – green, будет отправляться на сервер при запросах к любой странице и удалится из памяти через 30 секунд после создания.

- Cookie могут содержать только текстовые значения и для правильного последующего чтения значения должны быть закодированы с помощью метода encodeURIComponent. При чтении ранее закодированного значения используется метод decodeURIComponent
  encodeURIComponent - метод для кодирования строки в компонент, который будет использоваться в URI. Этот метод заменяет все символы кроме символов латинского алфавита, цифр и символов _ . ! ~ * ' ( )
  Метод используется для предотвращения некорректных запросов, например:

  Строка без использования URI кодировки –
  https://itvdn.com/search?term=position=absolute

  Строка без использования URI кодировки –
  https://itvdn.com/search?term=position%3Dabsolute

  Во второй строке символ = был заменен на %3D, так как данный символ используется для определения значения для ключа, в определенных ситуациях это может повлиять на правильно распознавание ключе и их значений.

- Web Storage API – механизм, который позволяет создавать пары ключ значение и сохранять их в памяти браузера, используя более интуитивно понятный синтаксис чем при работе с cookies

- Существует два механизма для хранения данных на клиенте.

  sessionStorage

CyberBionic Systematics ®
19 Evgenia Sverstiuka Str., 5 floor
Kyiv, Ukraine
t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Создает отдельное хранилище для каждого источника, который существует пока существует страница (при перегрузке страницы информация не удаляется). Данные sessionStorage удаляются при закрытии браузера. Данные sessionStorage не передаются на сервер. Ограничение хранилища до 5Мб, но может отличаться, в зависимости от используемого браузера.

localStorage
То же самое, что и sessionStorage, но информация сохраняется в хранилище бессрочно и остается доступной после перезапуска браузера. Данные удаляются из хранилища с помощью JavaScript или во время очистки кэша браузера.

- Origin (источник) – источник, из которого осуществляется загрузка контента. Включает протокол, имя хоста, порт.
  Например, http://localhost:88, https://itvdn.com, http://example.com, https://example.com.
  Все перечисленные значения относятся к разным источникам. Браузер создает хранилище отдельно для каждого источника, при условии, что из источника был получен код, который выполнил обращение к веб хранилищу.

- Методы для работы с localStorage и sessionStorage
  setItem(key, value) – сохраняет значение по ключу
  getItem(key) – возвращает значение по ключу, null если значения нет
  removeItem(key) – удаляет запись по ключу
  clear() – удаляет все записи для текущего источника
  key(index) – возвращает имя ключа по указанному индексу
  length – возвращает количество ключей в хранилище

- JSON (JavaScript Object Notation) – формат хранения данных, который базируется на JavaScript. Применяется для хранения и передачи данных между сервером и клиентом и является не зависимым от языка программирования.
  JSON.stringify(object) может использоваться для преобразования объекта или массива в JSON формат. Для преобразования значения из JSON формата в объект используется метод JSON.parse(json_string).
  Так как веб хранилище может работать только со строковыми значениями, при сохранении объектов может пригодиться использование данных методов.

- Получить доступ к cookies и web storage можно с помощью инструментов разработчика во вкладке Application.

Закрепление материала

- Что такое cookie, как они работают?
- Приведите пример задачи для которой могут используются cookies.
- Как можно удалить cookie с помощью JavaScript кода?
- Для чего нужен метод encodeURIComponent?
- Сколько времени хранятся данные в sessionStorage и сколько в localStorage?
- Какие лимиты на объем сохраненной информации существуют для веб хранилищ?
- Что такое источник?
- Что такое JSON, как сохранить и восстановить объект используя этот формат в JavaScript?

Самостоятельная деятельность учащегося

CyberBionic systematics
CyberBionic Systematics ®
19 Evgenia Sverstiuka Str., 5 floor
Kyiv, Ukraine
t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Выполните задания в директории Exercises\Tasks\017 Cookies and Web Storage. Текст задач, находится в комментариях в тегах script.

Рекомендуемые ресурсы

CSRF
https://ru.wikipedia.org/wiki/Межсайтовая_подделка_запроса

Свойство samesite
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie/SameSite

Cookie
https://developer.mozilla.org/ru/docs/Web/API/Document/cookie

Web Storage API
https://developer.mozilla.org/ru/docs/Web/API/Web_Storage_API

Origin (источник)
https://developer.mozilla.org/ru/docs/Web/HTTP/Headers/Origin

CyberBionic systematics

CyberBionic Systematics ®
19 Evgenia Sverstiuka Str., 5 floor
Kyiv, Ukraine

t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com