



Занятие №7

BOM, LocalStorage, Cookie



# BOM

**Основным объектом спецификации является объект окна.**

Многие вещи в браузере можно контролировать с помощью **javascript**, например, открытие окон, открытие вкладок, закрытие страниц, избранное и т. Д.

Эти функции не имеют ничего общего с веб-контентом. Поскольку стандарта нет, разные браузеры могут реализовывать одну и ту же функцию по-разному.

# DOM

**Самый фундаментальный объект DOM - это документ.**

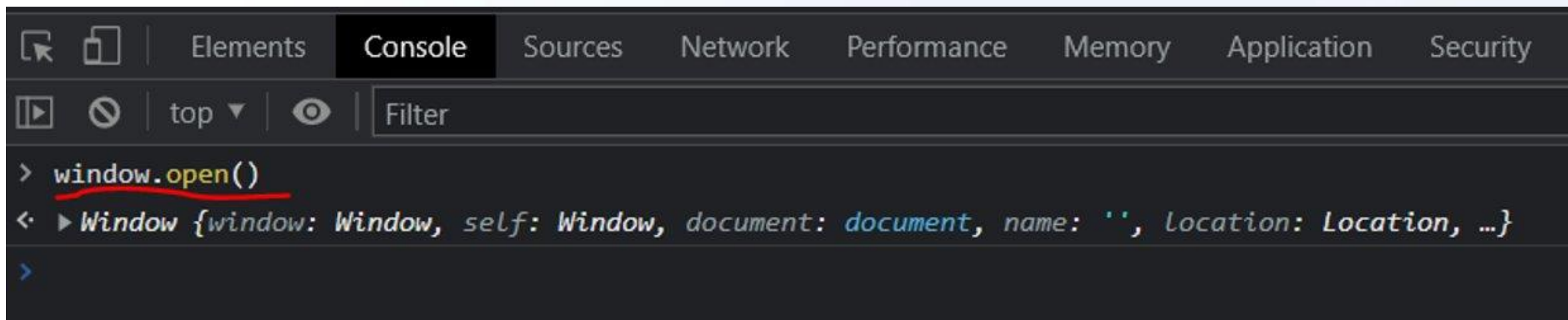
Под документом здесь понимается веб-страница, которая представляет собой html-документ.

DOM не имеет ничего общего с браузером.

Его интересует содержимое самой веб-страницы.

# Объект WINDOW

Объект «**window**» является основой браузерной объектной модели (BOM). Этот объект «представляет» окно браузера как главный контейнер для веб-страницы.



```
> window.open()
< ▶ Window {window: Window, self: Window, document: document, name: '', location: Location, ...}
>
```

# Основные свойства объекта «WINDOW»

**innerHeight** - Высота контента окна (без панелей браузера)

**innerWidth** - Ширина контента окна

**OuterHeight** - Высота окна браузера (включая панели)

**outerWidth** - Ширина окна браузера

**screenLeft, screenX** - Отступ от левого края окна браузера до начала содержимого страницы (контента)

**screenTop, screenY** - Отступ от верхнего края окна браузера до начала содержимого страницы (контента)

**scrollX, scrollY** - Величина (в пикселях) сдвига контента за счет полос прокрутки

Все эти свойства **доступны только для чтения**, то есть можно узнать их величину из JavaScript, но изменения их значений не будут иметь эффекта на отображении окна.

# Основные методы объекта «WINDOW»

**open()** Открывает новое пустое окно браузера

**open(addr,id,attr)** Открывает новое окно и загружает страницу с адресом **addr**. Окну присваивается имя «**id**» и применяются атрибуты «**attr**»

**stop()** Прекращает загрузку окна

**close()** Закрывает окно

**moveTo(X,Y)** Перемещает окно в точку экрана с заданными координатами

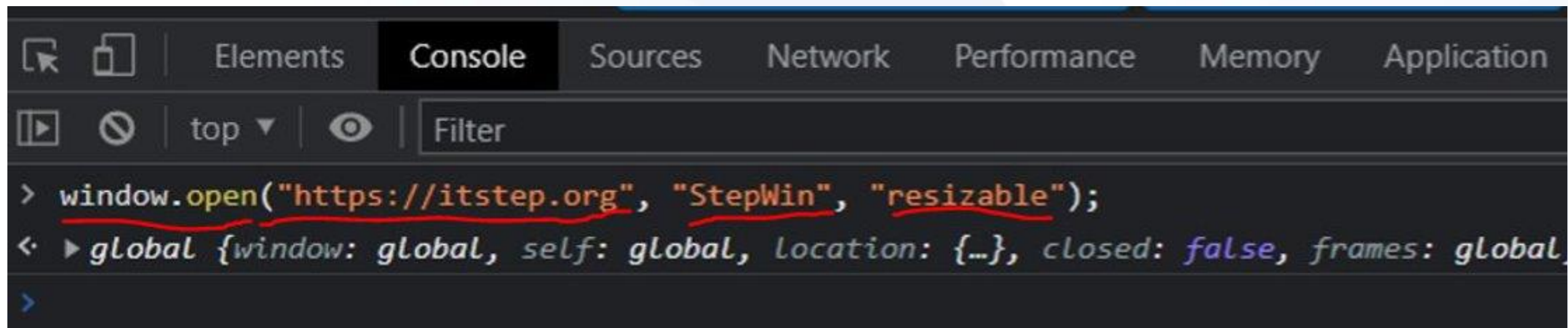
**moveBy(dX,dY)** Сдвигает позицию окна по каждой координате

**resizeTo(W,H)** Устанавливает размеры окна на заданные

**resizeBy(dW,dH)** Прибавляет к размерам окна переданные значения

**Позволяют управлять самими окнами.**

Откройте страницу в браузере и откройте консоль



# Локальное хранилище

**Локальное хранилище** - это свойство, которое позволяет сайтам и приложениям JavaScript сохранять пары ключ-значение в веб-браузере без даты истечения срока действия.

Это означает, что данные, хранящиеся в браузере, будут сохраняться даже после закрытия окна браузера.

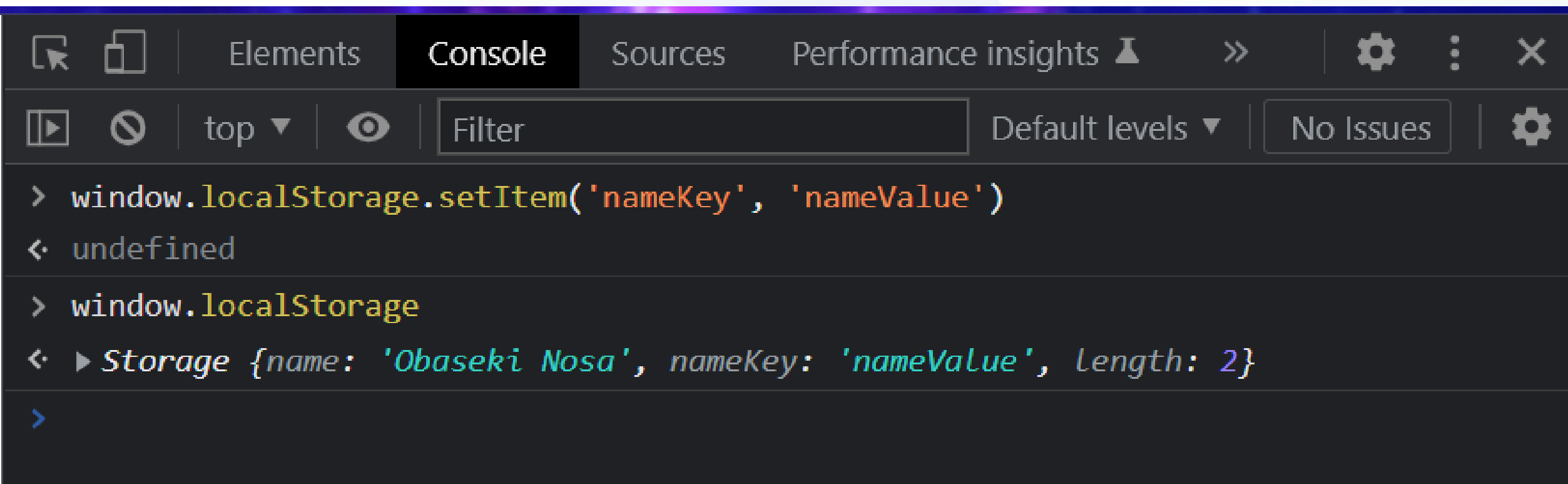
Механизм локального хранилища доступен через **window.localStorage.property**.

Окно.**localStorage** — это часть оконного интерфейса в JavaScript, который представляет окно, содержащее документ DOM.

Интерфейс Window содержит широкий спектр функций, конструкторов, объектов и пространств имен. **window.localStorage** - это свойство, доступное только для чтения, которое возвращает ссылку на объект локального хранилища, используемый для хранения данных, доступных только для источника, который его создал.



# Открываем консоль



# setItem()

Как следует из названия, этот метод позволяет сохранять значения в объекте `localStorage`.

Он принимает два параметра: ключ и значение. На ключ можно ссылаться позже, чтобы получить прикрепленное к нему значение.

# getItem()

Чтобы получить элементы из `localStorage`, используйте метод `getItem()`. `getItem()` позволяет вам получить доступ к данным, хранящимся в объекте `localStorage` браузера.

`getItem()` принимает только один параметр, который является ключом, и возвращает значение в виде строки.

# removeItem()

Чтобы удалить элементы локального хранилища, используйте метод `removeItem()`.

При передаче имени ключа метод `removeItem()` удаляет этот ключ из хранилища, если он существует. Если нет элемента, связанного с данным ключом, этот метод ничего не сделает.

# Cookie

Куки – это небольшие строки данных, которые хранятся в браузере.

Куки обычно устанавливаются веб-сервером при помощи заголовка Set-Cookie. Затем браузер будет автоматически добавлять их в (почти) каждый запрос на тот же домен при помощи заголовка Cookie.

# Для примера откройте сайт Додо пицца

## Переходим в консоль

```
> document.cookie  
⏪ '_ym_uid=1597728330919151894; dodo_utm=; rerf=AAAAAG0tZ5QEU61tAx7rAg==; ipp_  
uid=1672308628080/H49Zk0u1F6JmKD8x//FR7sYzN8Iq/HJnvFNkEGw==; ipp_key=v167230  
8628080/v33947245ba5adc7a72e273//JInJGXR3CFAwHgro5pjlA==; dodo_visit=3962512  
f-c160-4c13-890f-d9c306b45344; dodo_visitor=eaa326aa-acbf-45d4-b8b4-14e06440  
2e05; sessionID=1672308629981.owxa88ie; _ga=GA1.2.1332871374.1672308630; _gi  
d=GA1.2.858492759.1672308630; _fbp=fb.1.1672308630496.529835890; _ym_d=16723  
08631; _ym_isad=2; _ym_visorc=w; mindboxDeviceUUID=6f40471e-661b-4be2-90d5-e  
8cb4258e80b; directCrm-session=%7B%22deviceGuid%22%3A%226f40471e-661b-4be2-9  
0d5-e8cb4258e80b%22%7D; _gcl_au=1.1.1638514443.1672308637; locality=almaty'
```

```
> document.cookie = "newKey=newValue"
```

```
< 'newKey=newValue'
```

```
> document.cookie
```

```
< '_ym_uid=1597728330919151894; dodo_utm=; rerf=AAAAAG0tZ5QEU61tAx7rAg==; ipp_  
uid=1672308628080/H49ZkOulF6JmKD8x//FR7sYzN8Iq/HJnvFNkEGw==; ipp_key=v167230  
8628080/v33947245ba5adc7a72e273//JInJGXR3CFAwHgro5pjlA==; dodo_visit=3962512  
f-c160-4c13-890f-d9c306b45344; dodo_visitor=eaa326aa-acbf-45d4-b8b4-14e06440  
2e05; sessionID=1672308629981.owxa88ie; _ga=GA1.2.1332871374.1672308630; _gi  
d=GA1.2.858492759.1672308630; _fbp=fb.1.1672308630496.529835890; _ym_d=16723  
08631; _ym_isad=2; _ym_visorc=w; mindboxDeviceUUID=6f40471e-661b-4be2-90d5-e  
8cb4258e80b; directCrm-session=%7B%22deviceGuid%22%3A%226f40471e-661b-4be2-9  
0d5-e8cb4258e80b%22%7D; _gcl_au=1.1.1638514443.1672308637; locality=almaty;  
newKey=newValue'
```

```
> |
```

ВОПРОСЫ



ПРАКТИКА



Давайте подведем итоги.