



# Jquery



# План

- що таке jquery
- вибір html-елементів
- прохід DOM методами jquery
- керування класами і атрибутами
- керування css-властивостями
- події (event)
- анімація
- аjax
- jquery бібліотеки
- робота із слайдером

# Jquery

- це бібліотека JavaScript, призначена для спрощення різних завдань шляхом написання меншого коду
- для використання jquery потрібно її підключити до сторінки

```
<script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
```

<https://jquery.com/download/>

<https://api.jquery.com/>

<https://www.kobzarev.com/wp-content/uploads/books/jquery/jQuery-tutorial-for-beginners-1.0.0beta.pdf>

# Можливості

- **Маніпулювання DOM** - полегшує вибір елементів DOM
- **Робота з подіями** - простий спосіб захоплення різноманітних подій
- **Підтримка AJAX** - легко використовувати технологію AJAX
- **Анімації** - має велику кількість вбудованих ефектів анімації

# Вибір елемента

`$()` — це синонім для `jQuery()`

`$(selector)` - вибір елемента за вказаним селектором

`$("#content")` // вибираємо елемент з `id = content`

`$(".wrapper")` // вибираємо елементи з `class = wrapper`

`$("h2")` // вибираємо все теги `h2`

<https://jquery-docs.ru/category/selectors/>

# Фільтрація елементів DOM

- `.eq(n)` - повертає елемент з колекції що йде по заданому номеру
- `.find(selector)` - пошук дочірніх за селектором
- `.not(selector)` - фільтрує набір виключає з нього елементи з селектором
- `.is()` - перевіряє чи заданий елемент відповідає умові

<https://jquery-docs.ru/category/traversing/filtering/>

# прохід дерева DOM

`.children()` - знаходить дочірні елементи

`.next()` - елемент який лежить безпосередньо після заданого

`.nextAll()` - всі елементи які лежить безпосередньо після заданого

`.nextUntil(selector)` - всі елементи які лежить безпосередньо після заданого але до вказаного селектора

`.prev()` - елемент який лежить безпосередньо перед заданим

`.prevAll()` - всі елементи які лежить безпосередньо перед заданим

`.prevUntil(selector)` - всі елементи які лежить безпосередньо перед заданим але до вказаного селектора

`.siblings()` - всі сусідні елементи до і після вказаного

# прохід дерева DOM

`.parent()` - безпосередньо батьківський елемент

`.parents()` - пошук батьківського елемента вгору по ієрархії

`.parentsUntil(selector)` - пошук батьківського елемента до вказаного селектора

`.closest()` - перший елемент який співпадає селектору рухаючись вгору по ієрархії

<https://jquery-docs.ru/category/traversing/>



# Цикли по дереву DOM

`.each()` - ітерація над об'єктом JQuery, виконує функцію для кожного елемента

```
$( "li" ).each(function( index, element ) {  
    console.log( index + ": " + $( this ).text() );  
});
```

# Маніпулювання елементами

- класами
- атрибутами
- доступ до тексту
- додавання і видалення елементів

<https://jquery-docs.ru/category/manipulation/>

# Керування класами і атрибутами

`.hasClass()` - перевіряє чи елемент має клас

`.addClass()` - додає клас

`.removeClass()` - видаляє клас

`.toggleClass()` - якщо клас є то видаляє, якщо нема то додає

`.attr()` - отримує значення атрибуту, або встановлює атрибут

`.removeAttr()` - видаляє атрибут

`.prop()` - отримує значення властивості, або встановлює властивість

`.removeProp()` - видаляє властивість

# Керування CSS

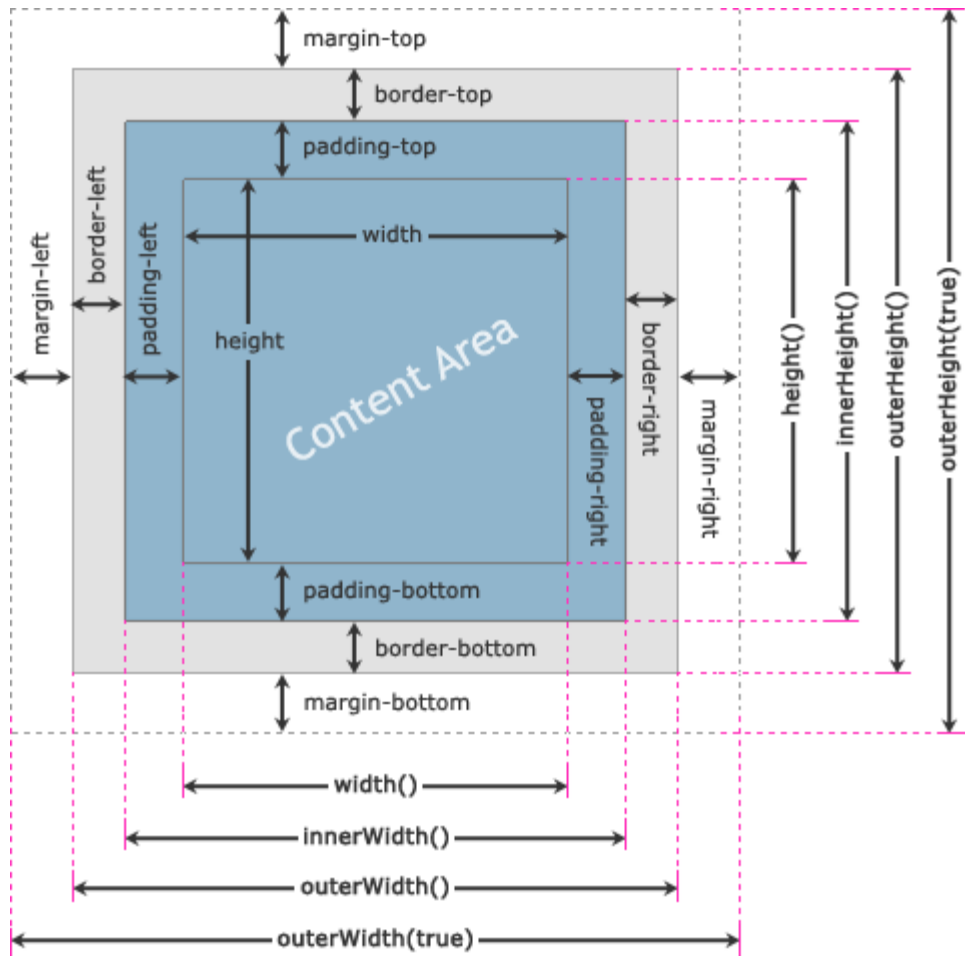
`.css()` - задавання CSS властивостей

```
$(element).css('color', 'red');
```

```
$(element).css({'color': 'red', 'margin-top': '10px'});
```

# розміри і позиціонування

- `.height()` - повертає і задає висоту елемента
- `.innerHeight()` - повертає і задає висоту елемента
- `.outerHeight()` - повертає висоту елемента
- `.width()` - повертає і задає ширину елемента
- `.offset()` - повертає позицію елемента на сторінці
- `.position()` - повертає значення `top` і `left` відносно батьківського елемента
- `.scrollTop()` - повертає і встановлює значення вертикального скролу



# додавання контенту

`.html()` - повертає або змінює html зміст

`.text()` - повертає або змінює текст

`.val()` - повертає або змінює значення елемента форми

# додавання елементів

- `.append()` - вставляє контент в кінець елемента
- `.appendTo()` - вставляє контент в кінець елемента
- `.prepend()` - вставляє контент на початок елемента
- `.prependTo()` - вставляє контент на початок елемента
- `.after()` - вставляє контент після елемента
- `.before()` - вставляє контент перед елементом
- `.insertAfter()` - вставляє контент після елемента
- `.insertBefore()` - вставляє контент перед елементом
- `.clone()` - копія вибраних елементів
- `.replaceWith()` - замінює елементи новими елементами



```
$ ( "<p>Test</p>" ) .appendTo ( ".inner" );
```

```
$ ( ".inner" ) .append ( "<p>Test</p>" );
```

```
$ ( ".inner" ) .after ( "<p>Test</p>" );
```

```
$ ( "<p>Test</p>" ) .insertAfter ( ".inner" );
```

# Практика

- динамічно додати 5 нумерованих строк (не через `<ol></ol>`)

# Видалення

`.detach()` - видаляє елемент із сторінки

`.empty()` - видаляє контент у вибраних елементах

`.remove()` - видаляє елемент із сторінки

# Події

`.ready()` - готовність дерева DOM

`.on()` - вставляює обробник події на вибраний елемент

```
$(document).ready(function() {});
```

```
$(element).on(event, function() {});
```

<https://jquery-docs.ru/category/events/>

# Події

`.resize()` - обробник на зміну розмірів екрана

`.scroll()` - обробник прокрутки екрана

`.trigger()` - викликає подію у вибраного елемента

`.submit()` - вставляє обробник події відправки форми або запускає цю

подію

# Приклади

scroll click [https://jsfiddle.net/blog\\_code/g9t5k7bm/10/](https://jsfiddle.net/blog_code/g9t5k7bm/10/)

resize [https://jsfiddle.net/blog\\_code/h246fgad/1/](https://jsfiddle.net/blog_code/h246fgad/1/)

# Практика

- Зробити попап(модальне вікно). Показувати його при кліку на кнопку і при зменженні екрану менше 768px

<https://getbootstrap.com/docs/3.3/javascript/#modals>

# Анімація

- `.animate()` - виконує анімацію CSS властивостей
- `.fadeIn()` - показує елементи через плавну зміну прозорості
- `.fadeOut()` - ховає елементи через плавну зміну прозорості
- `.fadeToggle()` - показує/ховає елементи через плавну зміну прозорості
- `.hide()` - ховає елементи через плавну зміну прозорості і розміру
- `.show()` - показує елементи через плавну зміну прозорості і розміру
- `.slideDown()` - показує елементи через плавну зміну розміру
- `.slideUp()` - ховає елементи через плавну зміну розміру
- `.slideToggle()` - показує/ховає елементи через плавну зміну розміру
- `.toggle()` - показує/ховає елементи



# Приклад

```
$("button").click(function(){  
  
    $("div").animate({top: '700px'});  
  
});
```

[https://jsfiddle.net/blog\\_code/vdp32cft/2/](https://jsfiddle.net/blog_code/vdp32cft/2/)

<https://www.bitdegree.org/learn/jquery-animate/>

<https://codepen.io/knard/pen/iDsAt>

# Практика

зробити анімацію завантажуючого прогресбару

# ajax через jquery

Для реалізації технології використовується метод `$ .ajax` або `jQuery.ajax`:  
`$ .ajax (властивості)` або `$ .ajax (url [, властивості])`

`url` (рядок) - адреса URL для запиту.

`success` (функція) - дана функція викликається після успішного завершення запиту. Функція отримує від 1 до 3 параметрів (в залежності від використовуваної версії бібліотеки). Але перший параметр завжди містить повертаються з сервера дані.

`data` (об'єкт / рядок) - призначені для користувача дані, які передаються на запитувану сторінку.

`dataType` (рядок) - можливі значення: `xml`, `json`, `script` або `html`. Опис типу даних, які очікуються у відповіді сервера.

`type` (рядок) - тип запиту. Можливі значення: `GET` або `POST`. За замовчуванням: `GET`.

```
$.ajax({  
  type: 'GET',  
  dataType: 'json',  
  url: 'адреса запыту',  
  success: function(data) {  
    $('<div>.results</div>').html(data);  
  }  
});
```

<https://jquery-docs.ru/jQuery.ajax/>

# Бібліотеки

## Sliders

<http://idangero.us/swiper/>

<https://owlcarousel2.github.io/OwlCarousel2/>

<http://sachinchoolur.github.io/lightslider/examples.html>

<https://alvarotrigo.com/fullPage/ru/>

<http://kenwheeler.github.io/slick/>

## Gallery

<http://sachinchoolur.github.io/lightslider/examples.html>

## Data

<http://underscorejs.org/>

## Chart

<https://d3js.org/>

<https://www.highcharts.com/demo>

## Svg

<https://maxwellito.github.io/vivus/>

## Parallax and animation

<https://mynameismatthieu.com/WOW/>

<http://scrollmagic.io/examples/index.html>

<https://greensock.com/>

<http://matthew.wagerfield.com/parallax/>

## Grid and filter

<https://isotope.metafizzy.co/>

<https://masonry.desandro.com/>

# Практика

додавання слайдера slick на сайт

<https://kenwheeler.github.io/slick/>



# Посилання

<https://api.jquery.com/>

[http://jquery.page2page.ru/index.php5/%D0%97%D0%B0%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D0%B0%D1%8F\\_%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0](http://jquery.page2page.ru/index.php5/%D0%97%D0%B0%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0)

<https://www.w3schools.com/jquery/>

# Відео

<https://www.youtube.com/watch?v=DaYyTDZDCQk&list=PLvItDmb0sZw964PmBjUcB75x17RK7M5ZA&index=1>

<https://www.youtube.com/watch?v=uUccmFUDd5s&t=403s>