

8. Додаток

Таблиця 1. CSS-селектори jQuery

| Селектор | Опис | Приклад |
|----------------------------|---|--|
| Прості селектори | | |
| * | усі елементи | \$("*") |
| .className | елементи з класом className | \$(".bg-color") |
| #idName | елемент (один!) з ідентифікатором idName | \$("#test") |
| tagName | елементи із заданим ім'ям тегу | \$("h2") |
| Комбіновані селектори | | |
| first, second, ... | елементи, що відповідають будь-якому з селекторів first, second, ... | \$("h2, p") |
| outer inner | елементи з inner, які є нащадками (тобто містяться всередині) елементів з outer | \$(".test p") |
| parent > child | елементи з child, які є прямими нащадками елементів з parent | \$("#wrap > div") |
| prev + next | елементи з next, які йдуть безпосередньо за елементами з prev | \$("label + input") |
| prev ~ next | елементи з next, які йдуть за елементами з prev | \$(".test ~ .block") |
| Селектори атрибутів | | |
| [name] | елементи з атрибутом name | \$("[data-url]") \$("[data-dismiss]") |
| [name = value] | елементи, в яких значення атрибута name співпадає з value | \$("[name = 'radio-bg']") \$("[data-dismiss = 'true']") |
| [name != value] | елементи, в яких значення атрибута name не співпадає з value | \$("[class != 'test']") |
| [name ^= value] | елементи, в яких значення атрибута name починається з value | \$("a[href ^= 'https://']") |
| [name \$= value] | елементи, в яких значення атрибута name закінчується на value | \$("[class \$= '-bg']") |
| [name *= "value"] | елементи, в яких значення атрибута name містить value | |
| [first][second][...] | елементи, що відповідають усім заданим умовам на атрибути одночасно | \$("[name='bg'] [type='radio']") |
| Фільтри дочірніх елементів | | |
| :first-child | елементи, які йдуть першими у своїх безпосередніх предках | \$(".block:first-child") |
| :last-child | елементи, які йдуть останніми у своїх безпосередніх предках | \$(".block:last-child") |

| | | |
|----------------------|---|-----------------------------------|
| :nth-child() | елементи, які розташовані у своїх безпосередніх предках за певними умовами | \$(".block:nth-child(3n)"); |
| :only-child | елементи, єдині у своїх безпосередніх предках | \$(".block img: only-child") |
| :only-of-type | елементи, які є єдиними нащадками у своїх батьківських елементах | \$("button: only-of-type") |
| :first-of-type | ті з вибраних елементів, які першими зустрічаються у своїх батьківських елементах | \$("span: first-of-type") |
| :last-of-type | ті з вибраних елементів, які останніми зустрічаються у своїх батьківських елементах | \$('.text:last-of-type') |
| :nth-last-of-type() | ті з вибраних елементів, які і є визначеними по черзі відносно останнього елемента у своєму елементі-батьку | \$("ul li:nth-last-of-type(2)") |
| :nth-of-type() | дочірні елементи певного типу у певній послідовності всередині батьківського елемента | \$(".column:nth-of-type(3n)") |

Таблиця 2. jQuery-селектори

| Селектор | Опис | Приклад |
|-----------------------|---|------------------------|
| <i>Simple Filters</i> | | |
| :first | перший знайдений елемент | \$(".block: first") |
| :last | останній знайдений елемент | \$(".block: last") |
| :eq(n) | елемент, який йде під заданим номером серед обраних | \$(".block: eq(3)") |
| :not(selector) | усі знайдені елементи, крім зазначених у selector | \$(":not(.color)") |
| :even | елементи з парними номерами позицій в наборі вибраних елементів | \$(".block: even") |
| :odd | елементи з непарними номерами позицій в наборі вибраних елементів | \$(".block: odd") |
| :gt() | елементи з індексом, що перевищує n | \$(".block: gt(2)") |
| :lt() | елементи з індексом меншим, ніж n | \$(".block: lt(3)") |
| :header | елементи, які є заголовками (з тегами h1, h2 і т. д.) | \$(":header") |
| :animated | елементи, які у даний момент задіяні в анімації | \$(".block: animated") |
| :hidden | невидимі елементи сторінки | \$("form: hidden") |
| :visible | видимі елементи сторінки | \$("form: visible") |
| :lang(language) | елементи, в яких вказані мови вмісту | \$("html: lang(ru)") |
| :root | елемент, який є кореневим у документі | \$(":root") |

Таблиця 3. Селектори форм

| Селе-ктор | Опис | Приклад |
|-----------|--|--|
| :button | елементи з тегом <button> або типом button (<input type="button">) | <code>\$(':button'). addClass('shadow');</code> |
| :checkbox | елементи з атрибутом type="checkbox" | <code>\$(':checkbox'). addClass('shadow');</code> |
| :checked | вибрані елементи (зі статусом/атрибутом checked). Це можуть бути елементи з атрибутом type="checkbox" або type="radio" | <code>\$(':checked'). addClass('shadow');</code> |
| :disabled | неактивні елементи форми (зі статусом disabled) | <code>\$(':disabled'). prop('disabled', false);</code> |
| :enabled | активні елементи форми (зі статусом enabled або без атрибуту disabled) | <code>\$(':enabled'). addClass('bg-gray');</code> |
| :focus | елементи форми, що знаходяться у фокусі | <code>\$(':focus').addClass('bg-red');</code> |
| :file | поля завантаження файлів (<input type="file">) | <code>\$(':file').addClass('red-border');</code> |
| :image | елементи, які є зображеннями для надсилання форми (<input type="image">) | <code>\$(':image').prop('disabled', true);</code> |
| :input | елементи, які є елементами форми (з тегами input, textarea або button) | <code>\$(':input').addClass('bg-green shadow')</code> |
| :password | поля для введення пароля (<input type="password">) | <code>\$(':password'). addClass('bg-red')</code> |
| :radio | елементи-перемикачі, тобто які мають type = "radio" | <code>\$(':radio'). addClass('shadow')</code> |
| :reset | кнопки для очищення форми (<input> або <button>) | <code>\$(':reset'). addClass('shadow bg-gray');</code> |
| :selected | вибрані елементи (зі статусом selected), як правило, елементи типу <option> | <code>\$(':selected'). addClass('gray-border');</code> |
| :submit | кнопки для відправлення форми (<input> або <button>) | <code>\$(':submit'). addClass('shadow');</code> |
| :text | будь-які текстові поля | <code>\$(':text').addClass('bg-blue');</code> |

Приклади вибору елементів сторінки, залежно від застосованих селекторів, ви знайдете на ресурсі codepen.io або у файлі *examples/jquery-selectors.html*.

Для того щоб побачити, як діє селектор, клікніть на будь-яку радіо-кнопку і зверніть увагу на стилі, що змінилися. В деяких випадках над вибраним елементом з'являтиметься стрілка, яка вказує на те, що змінилося.

Рисунок 41

Щоб зрозуміти, які саме селектори були використані, доведеться заглянути у JS-код.

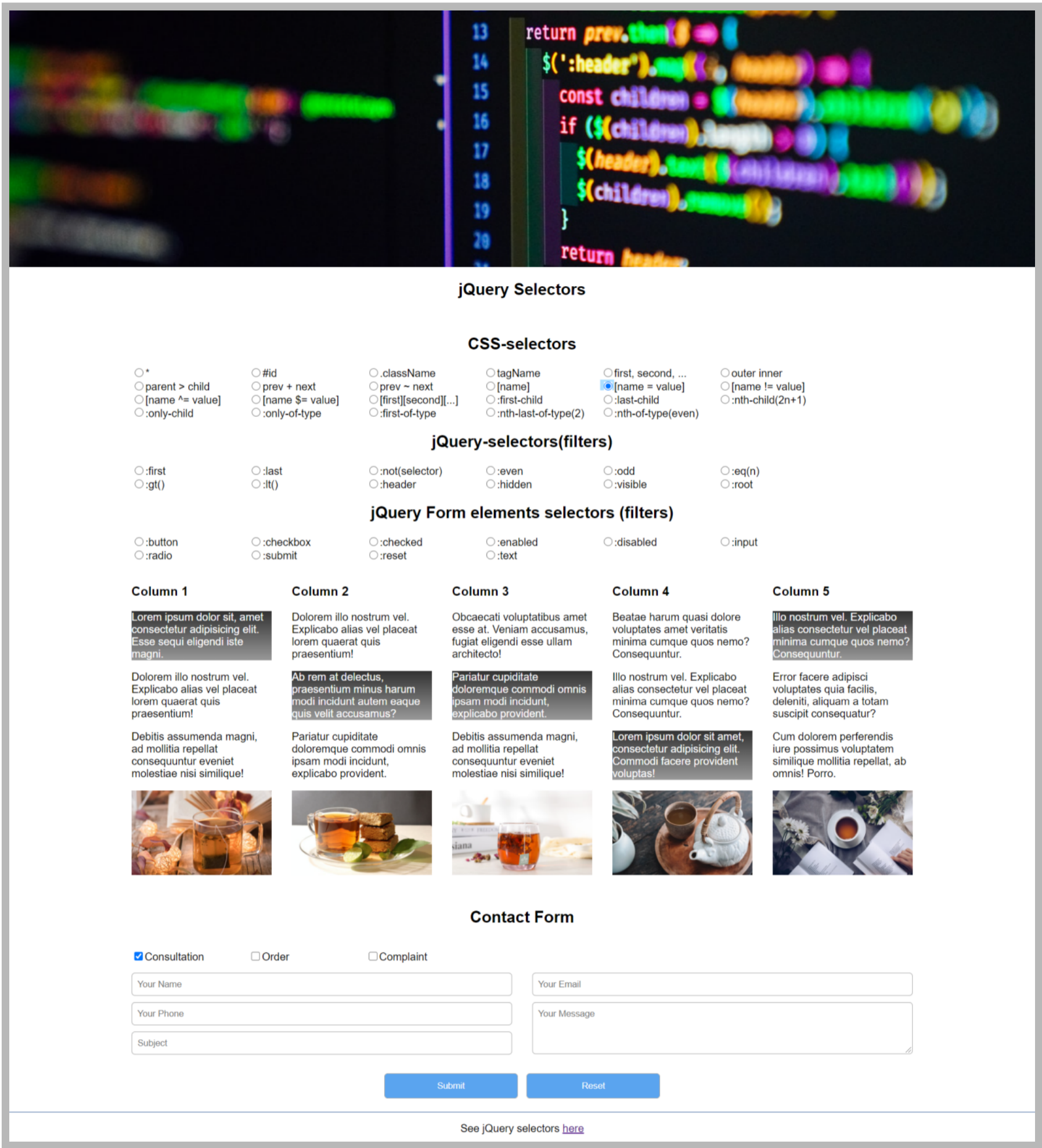


Рисунок 42

Таблиця 4. Методи керування стилями та класами jQuery

| Метод | Опис | Приклад |
|--|--|---|
| Методи керування стилями jQuery | | |
| <code>\$('.selector').css('property')</code> | Отримати значення вказаної CSS-властивості | <code>\$('#block').css('display');</code> |

| | | |
|--|---|---|
| <code>\$('#selector').css('property', 'value');</code> | Встановити 1 значення | <code>\$('#block').css('color', 'red');</code> |
| <code>\$('#selector').css({prop1: 'value1', 'prop2': 'value2', prop3: 'value3'});</code> | Встановити декілька значень. CSS-стилі можна записувати в лапках у нотації CSS, або без лапок у нотації JavaScript за допомогою camelCase | <code>\$('.example').css({color: 'red', 'background-color': 'pink', 'fontSize: '18px'});</code> |
| Методы керування класами jQuery | | |
| <code>\$('#selector').addClass('class_name')</code> | Додає клас до списку класів елемента. Можна додати декілька через пробіл | <code>\$('#a').addClass('read-more'); \$('.btn').addClass('btn-outline btn-secondary');</code> |
| <code>\$('#selector').removeClass('class_name')</code> | Видаляє клас зі списку класів елемента. Можна видалити декілька через пробіл | <code>\$('.nav-links.active').removeClass('active'); \$('.products').removeClass('col-md-4 col-sm-6');</code> |
| <code>\$('#selector').toggleClass('class_name')</code> | Додає/видаляє клас(-и) зі списку класів елемента залежно від того, чи призначений відповідний клас(-и) елементу | <code>\$('.nav-links').toggleClass('active'); \$('.products').toggleClass('col-md-4 col-sm-6');</code> |
| <code>\$('#selector').hasClass('class_name')</code> | Перевіряє, чи призначений вказаний клас елементу. Якщо так, то повертає true, в іншому випадку – false | <code>if(\$('.nav-links').hasClass('active')) { \$('.nav-links').parent().addClass('menu-active'); }</code> |

Таблица 5. Traversing. Методи обходу DOM

| Метод | Опис | Приклад |
|-------------------------|---|---|
| <code>add()</code> | Додавання елементів до набору вибраних елементів | <code>\$('.column h3').add('p').addClass('red-border');</code> |
| <code>addBack()</code> | Після пошуку дочірніх елементів додає попередній набір елементів до поточного для призначення стилів або обробників подій | <code>\$('.column').find('p').addBack().addClass('red-border');</code> |
| <code>children()</code> | Повертає усі прямі дочірні елементи вибраного елемента | <code>\$('.column-selected').children().addClass('red-border');</code> |
| <code>closest()</code> | Повертає перший предок вибраного елемента | <code>\$('.column').closest('.container').addClass('gray-border');</code> |
| <code>contents()</code> | Повертає усі прямі дочірні елементи вибраного елемента (включаючи вузли тексту та коментаря) | <code>\$('.column-selected').contents().addClass('shadow');</code> |
| <code>each()</code> | Виконує функцію для кожного вибраного елемента | <code>\$('.column').each(function(ind, elem){ let columnText = \$(elem).find('p');</code> |

| | | |
|-----------------------------|---|---|
| | | <pre>columnText.html(ind+' ' + columnText.text()); });</pre> |
| <code>end()</code> | Завершує останню операцію фільтрації (пошуку) вкладених елементів у поточному ланцюгу і повертає набір елементів у попередній стан | <pre>\$('.column').find('.text'). addClass('shadow').end().find('img'). addClass('shadow');</pre> |
| <code>eq()</code> | Повертає елемент із зазначеним індексом у списку вибраних елементів. Індeksi рахуються з 0 | <pre>\$('.column').eq(1).addClass('bg-blue');</pre> |
| <code>filter()</code> | Повертає набір елементів, що збігаються, які відповідають зазначеному селектору або функції | <pre>\$('.column img').filter('.snow'). addClass('gray-border shadow');</pre> |
| <code>find()</code> | Повертає елементи-нащадки вибраного елемента за вказаним селектором | <pre>\$('.column').find('.text'). addClass('shadow');</pre> |
| <code>first()</code> | Повертає перший елемент серед обраних | <pre>\$('.column').first().addClass('bg-red');</pre> |
| <code>has()</code> | Повертає усі елементи, які мають один або декілька елементів всередині них із зазначеним селектором | <pre>\$('.column').has('img.snow'). addClass('bg-blue');</pre> |
| <code>is()</code> | Перевіряє набір елементів, що збігаються, на об'єкт Selector/елемент/jQuery та повертає значення true, якщо хоча б один з цих елементів відповідає заданим аргументам | <pre>\$('.column').each(function(i, elem){ if(\$(this).is(':nth-child(2n)')) \$(this). addClass('bg-red'); });</pre> |
| <code>last()</code> | Повертає останній елемент із вибраних елементів | <pre>\$('.column').last().addClass('bg-green');</pre> |
| <code>map()</code> | Передає кожен елемент із набору функцією, роблячи новий об'єкт jQuery зі значеннями, що повертаються | <pre>\$('.container:eq(1)').prepend('<p class = "text-center">Photo titles: ' + \$('.column img').map(function(ind, elem){ return \$(elem).attr('alt'); }).get().join(', ') + '</p>');</pre> |
| <code>next()</code> | Повертає наступний споріднений елемент вибраного елемента | <pre>\$('.column-selected').next(). addClass('bg-green');</pre> |
| <code>nextAll()</code> | Повертає усі сусідні елементи вибраного елемента | <pre>\$('.column-selected').nextAll(). addClass('bg-red');</pre> |
| <code>nextUntil()</code> | Повертає усі сусідні елементи одного рівня між двома заданими аргументами | <pre>\$('.column p').nextUntil('img'). addClass('shadow');</pre> |
| <code>not()</code> | Повертає елементи, що не збігаються за певними критеріями | <pre>\$('.column').not(':nth-child(2)'). addClass('bg-green');</pre> |
| <code>offsetParent()</code> | Повертає перший позиційний батьківський елемент | <pre>\$('.column-selected').offsetParent(). addClass('gray-border');</pre> |
| <code>parent()</code> | Повертає прямий батьківський еле- | <pre>\$('.column').parent().addClass('red-</pre> |

| | | |
|----------------|--|---|
| | мент для вибраного елемента | border'); |
| parents() | Повертає усі батьківські елементи обраного елемента | \$('.column').parents('article').addClass('bg-red shadow'); |
| parentsUntil() | Повертає усі батьківські елементи заданого елемента — не тільки безпосереднього, а й прабадька, прапрабадька і так далі до кореневого елемента (тобто до тега html). | \$('.column').parentsUntil('.container').addClass('bg-blue'); |
| prev() | Повертає попередній споріднений елемент вибраного елемента (-ів) | \$('.column-selected').prev().addClass('bg-blue'); |
| prevAll() | Повертає усі попередні споріднені елементи вибраного елемента | \$('.column-selected').prevAll().addClass('bg-red shadow'); |
| prevUntil() | Повертає усі попередні елементи одного рівня між двома заданими аргументами | \$('.column p').prevUntil('img').addClass('shadow'); |
| siblings() | Повертає усі споріднені елементи вибраного елемента | \$('.column p:first-of-type').siblings().addClass('gray-border'); |
| slice() | Вибирає з набору елементів тільки ті, індекси яких вказані в діапазоні у дужках | \$('.column').slice(1,4).addClass('bg-red shadow'); |
| wrap() | Поміщає вибрані елементи всередину заданого елемента | \$('.column').wrap('<div class = "bg-green shadow"></div>'); |
| wrapAll() | Обертає усі вибрані елементи у загальний елемент-обгортку | \$('.column').wrapAll('<div class = "bg-red shadow"></div>'); |

У прикладі, який міститься в папці *examples/jquery-traversing-methods.html* або на ресурсі codepen.io, ви можете поклацати радіо-перемикачі, щоб переглянути роботу усіх описаних методів. У код додаються різні класи зі стилями, що виділяють якісь елементи залежно від вибраного методу. По центру виділено колонку, відносно якої визначаються елементи такими методами, як `prev()`, `prevAll()`, `next()` та ін.

Код цього файлу використаний у таблиці з описом методів у вигляді прикладів.