

# Modyfikowanie wyglądu

## Użycie wartości atrybutu

Ostatni przykład na listingu 1.7 pokazuje, jak wybrać element za pomocą selektora, który używa atrybutu. jQuery pozwala wyszukać i pobrać przy użyciu uproszczonych wyrażeń regularnych dowolny atrybut. Możliwe do zastosowania filtry atrybutów przedstawia tabela 1.2.

**Tabela 1.2.** Definicje atrybutu

Nazwa	Opis
[atrybut]	Element o zadanym atrybucie, którego wartość jest dowolna
[atrybut = 'wartość']	Element o zadanym atrybucie, którego wartość jest określona
[atrybut != 'wartość']	Element o zadanym atrybucie, który nie zawiera określonej wartości
[atrybut ^= 'wartość']	Element o zadanym atrybucie, który rozpoczyna się od podanej wartości
[atrybut \$= 'wartość']	Element o zadanym atrybucie, który kończy się podaną wartością
[atrybut *= 'wartość']	Element o zadanym atrybucie, który zawiera podaną wartość
[atrybut ~= 'wartość']	Element o zadanym atrybucie, którego wartość znajduje się na liście (wartości rozdzielone spacjami)

### UWAGA

Stosowanie tego typu rozwiązania może spowolnić działanie strony. Wybór elementów oparty na nazwie, identyfikatorze bądź klasie poprawia wydajność działania kodu JavaScript.

Aby usunąć atrybut (wraz z wartością), należy posłużyć się metodą `.removeAttr()`, np. użycie kodu `$('p').removeAttr('style');` spowoduje usunięcie stylu przypisanego do akapitu.

Dwie inne metody, `.addClass()` i `.removeClass()`, pozwalają dodać lub usunąć nazwy klasy w wartości atrybutu `class`. Są przydatne, ponieważ nie wpływają na zdefiniowane nazwy klas.

### Listing 1.17. Użycie metody `.each()`

*// Pobranie id i numeru elementu*

```
$('#div').each(function(nr) {  
    console.log('numer diva o nazwie', this.id, 'to', nr);  
});
```

*// Użycie przycisku*

```
$('#button').click(function() {  
    $('a').each(function() {  
        alert($(this).text());  
    });  
});
```

Bind to the `change` event instead of `click`. However, you will probably still need to check whether or not the checkbox is checked:

```
$(".checkbox").change(function() {  
    if(this.checked) {  
        //Do stuff  
    }  
});
```

## 3 zadania

Napisz skrypt z wykorzystaniem jQuery, który:

1. Po naciśnięciu przycisku **DODAJ na początku listy** doda się napis: **LISTA**. Wypunktowania na stronie nie ma, pojawi się ono po naciśnięciu przycisku.
2. Przygotuj stronę wyświetlającą pojedyncze pole typu **checkbox** z napisem **Akceptuj** oraz przycisk **Wyślij** (nieaktywny). Napisz skrypt, który aktywuje przycisk **Wyślij** po zaznaczeniu pola **Akceptuję**. Po naciśnięciu przycisku **Wyślij** zniknie pole **checkbox** i pojawi się zamiast tego napis: **WYSŁANO**.  
Pamiętaj, że klasy css powinny mieć co najmniej 2 znaki i nie zaczynać ich od cyfry. **Przypisanie wielu klas:** `class="pierwsza-klasa x2 k3"` – trzy klasy
3. W kodzie strony umieść akapit z dwoma klasami. Pierwsza niech ustawia kolor tła, a druga niech ustawia wielkość czcionki na 150%. Napisz skrypt który usunie te klasy formatujące akapit po naciśnięciu przycisku.
4. W kodzie strony umieść 6 elementów DIV. Następnie wykorzystując funkcję `each()` - która działa jak FOR EACH, dodaj do każdego DIV napis: `div`