

Opis biblioteki JQuery

WPROWADZENIE

Szybki rozwój biblioteki jQuery sprawia, że wkrótce jej znajomość będzie niezbędna dla każdego projektanta stron internetowych.

Korzystając z biblioteki jQuery można wzbogacić stronę o proste efekty. Do ich uzyskania można używać wbudowanych ustawień lub definiować własne ustawienia czasu trwania. Można także tworzyć własne animacje dowolnych własności CSS.

POBIERANIE BIBLIOTEKI

Podstawę biblioteki jQuery stanowi JavaScript, który sam w sobie jest niezwykle bogatym i ekspresyjnym językiem.

Odpowiedni plik biblioteki można pobrać ze strony: <http://www.jquery.com>

Staraj pobrać się możliwie najnowszą bibliotekę a następnie dodaj ją w sekcji <HEAD> strony WWW np.:

```
<script type="text/javascript" src="jquery.js"></script>
```

WBUDOWANE EFEKTY

Najczęściej używane efekty są dostępne w jQuery za pomocą wbudowanych metod:

show - Pokazuje wybrany element

hide - Ukrywa wybrane elementy.

fadeIn - Przezroczystość wybranych elementów jest stopniowo zmniejszana do 100%.

fadeOut - Przezroczystość wybranych elementów jest stopniowo zwiększana do 0%.

slideDown - Wybrane elementy są rozwijane w pionie.

slideUp - Wybrane elementy są zwijane w pionie.

slideToggle - Wybrane elementy są pionowo rozwijane lub zwijane, w zależności od tego, czy są widoczne.

Przykład Prosty przykład wykorzystania wbudowanego efektu

```
$('#h1').show();
```

WYBIERANIE ELEMENTÓW FORMULARZA

W bibliotece jQuery znajdziemy kilka pseudoselektorów, które ułatwiają znajdowanie elementów formularzy. Pseudoselektory te są szczególnie przydatne, ponieważ za pomocą standardowych selektorów CSS trudno rozróżnić elementy na podstawie ich stanu czy typu.

:button

Wybiera elementy `button` oraz elementy oznaczone jako `type="button"`

:checkbox

Oznacza pola wejściowe oznaczone jako `type="checkbox"`

:checked

Oznacza wszystkie pola wejściowe, które zostały zaznaczone

:disabled

Wybiera elementy dezaktywowane

:enabled

Wybiera aktywne elementy formularza

:file

Wybiera pola wejściowe oznaczone jako `type="file"`

:image

Wybiera pola wejściowe oznaczone jako `type="image"`

:input

Wybiera elementy `input`, `textarea` oraz `select`

:password

Wybiera pola wejściowe oznaczone jako `type="password"`

:radio

Wybiera pola wejściowe oznaczone jako `type="radio"`

:reset

Wybiera pola wejściowe oznaczone jako `type="reset"`

:selected

Wybiera opcje, które zostały zaznaczone

:submit

Wybiera pola wejściowe oznaczone jako `type="submit"`

:text

Wybiera pola wejściowe oznaczone jako `type="text"`

ZMIENIANIE CZASU TRWANIA WBUDOWANYCH EFEKTÓW

Z wyjątkiem metod `show` i `hide`, animacja wszystkich wbudowanych metod domyślnie trwa 400 ms. Ustawienia te można zmienić w prosty sposób.

Przykład 6.2. Ustawiamy czas trwania efektu

```
$('h1').fadeIn(300);           // element pojawi się w całości w ciągu 300 ms
$('h1').fadeOut('slow');       // jedno ze standardowych ustawień szybkości
```

JQUERY.FX.SPEEDS

W bibliotece jQuery znajduje się obiekt `speeds`, który zawiera domyślną szybkość animacji, a także ustawienia `slow` i `fast` (wolne i szybkie).

```
speeds: {
  slow: 600,
  fast: 200,
  // domyślna szybkość
  _default: 400
}
```

Obiekt ten można również przededefiniować lub dodać do niego nowe składowe, np. jeśli chcemy zmienić domyślny czas trwania efektów lub wprowadzić własne ustawienia szybkości.

Przykład 6.3. Dodawanie własnych ustawień szybkości do obiektu `speeds`

```
jQuery.fx.speeds.blyskawicznie = 100;
jQuery.fx.speeds.zolw = 2000;
```

WYKONYWANIE KODU PO ZAKOŃCZENIU EFEKTU

Często chcemy, aby po zakończeniu animacji został wykonany jakiś blok kodu — jeśli zostanie wykonany wcześniej, może obniżyć jakość animacji, bądź usunąć któreś z jej elementów.

W tym przypadku zdarzeniem, do którego się będziemy odwoływać jest koniec animacji. Słowo kluczowe `this` wewnątrz funkcji wywołania zwrotnego odwołuje się do elementu, na którym został wywołany efekt. Podobnie jak w przypadku funkcji obsługi zdarzeń, element ten może zostać przekształcony w obiekt jQuery za pomocą konstrukcji `$(this)`.

Przykład 6.4. Wykonanie kodu po skończeniu animacji

```
$('#div.stary').fadeOut(300, function() { $(this).remove(); });
```

Jeśli selektor nie zwróci żadnych elementów, wywołanie zwrotne nigdy nie zostanie wykonane! Aby uniknąć tego problemu, należy sprawdzić, czy selektor wybrał jakieś elementy — jeśli nie, wywołanie zwrotne może zostać wykonane natychmiast.

Przykład 6.5. Kod wykonujący wywołanie zwrotne nawet jeśli nie ma żadnych elementów, które można by animować.

```
var $cos = $('#nieIstnieje');

var wz = function() {
    console.log('zrobione!');
};

if ($cos.length) {
    $cos.fadeIn(300, wz);
} else {
    wz();
}
```

TWORZENIE WŁASNYCH EFEKTÓW ZA POMOCĄ METODY `$.fn.animate`

Dostępna w bibliotece jQuery metoda `$.fn.animate` umożliwia animację dowolnych własności CSS. Możemy ustawić konkretną wartość parametru animacji, bądź wartość względną do ustawionej już wartości.

Przykład 6.6. Tworzenie własnych efektów za pomocą metody `$.fn.animate`

```
$('#div.trocheZabawy').animate(
{
```

```

        left : "+=50",
        opacity : 0.25
    },
    300, // czas trwania animacji
    function() { console.log('zrobione!'); // wywołanie zwrotne
});

```

Przy użyciu czystego jQuery animacja właściwości kolorów metodą `$.fn.animate` jest niemożliwa. Ten sam efekt można jednak osiągnąć za pomocą [wtyczki color](#). Wtyczki zostaną omówione w dalszej części podręcznika.

DYNAMIKA ANIMACJI

Definicja: *Dynamika animacji* to sposób w jaki wykonywany jest dany efekt — określa, czy zmiany zachodzą równomiernie, czy też w miarę upływu czasu animacja zwalnia lub przyspiesza. W bibliotece jQuery dostępne są tylko dwie metody pozwalające modyfikować dynamikę animacji: `swing` oraz `linear`. Jeśli zależy ci na płynniejszym przejściu animacji, możesz skorzystać z wielu dostępnych wtyczek.

W wersji jQuery 1.4 istnieje możliwość ustawienia dynamiki każdej własności z osobna poprzez metodę `$.fn.animate`.

Przykład 6.7. Ustawianie dynamiki animacji dla dwóch własności z osobna

```

$('div.trocheZabawy').animate(
    {
        left : [ "+=50", "swing" ],
        opacity : [ 0.25, "linear" ]
    },
    300
);

```

Odpowiedz na Pytania (wyszukaj odpowiedzi korzystając z podręcznika bądź odpowiednich stron WWW):

1. Jakie zadania można realizować na stronie za pomocą biblioteki JQuery?
2. Podaj metody dołączania biblioteki JQuery do skryptu.
3. Co może być metodą biblioteki JQuery.
4. Jaka rolę pełnią filtry JQuery.
5. Do jakich celów możemy wykorzystać obsługę zdarzeń?
6. Opisz działanie wtyczki **validation**