



jQuery w Visual Studio

- Autor projektu
- Co robi jQuery?
- Podstawowe informacje
- Zalety korzystania z biblioteki
- Przykłady

O Autorze

- W 2005 r. John Resig (obecnie pracownik Mozilla Corporation) pisze bibliotekę.
- Cel: Łatwe odnajdywanie elementów strony i przypisywanie do nich zachowań
- Przed opublikowaniem dodał możliwość modyfikacji drzewa DOM+animacje.
- Nazwa jQuery by podkreślić kluczową rolę odnajdywania elementów strony za pomocą zapytać i wykonania na nich działań za pomocą Java Scriptu

[objeto].**eval**(cadena) Este método, común a todos los navegadores, evalúa una cadena de código JavaScript en el contexto del objeto global. Puede invocarse incluso sin hacer referencia a ningún objeto ejecutando las instrucciones y evaluando los objetos previamente.

```
var x = 2; var y = 10; var z = 20;  
document.write(eval("x * y + z"))
```

Se obtiene: 2051

javascript

objeto.**toString**() La finalidad de este método, común a todos los navegadores, es convertir un objeto en una cadena de texto.

```
var meses = new Array("Enero", "Febrero", "Marzo")
```

Como podemos ver, meses es un objeto, y no es posible concatenarlo directamente con una cadena de texto.

```
document.write("Los meses son: " + meses)
```

Se obtiene: Los meses son: Enero,Febrero,Marzo

=



Co robi jQuery?

- Udostępnia ogólny poziom abstrakcji służący do tworzenia skryptów
- Uzyskanie dostępu do elementów dokumentu.
- W javascript przejście całego drzewa DOM = ok. 50 linii kodu w jQuery = 1

- 
- Modyfikacja wyglądu strony internetowej – pełne wsparcie CSS (rozpoznawanie selektorów)
 - Zmiana zawartości DOM – nie tylko wygląd
 - Reagowanie na interakcję z użytkownikiem (przechwycenia kliknięć itd.) – eliminacja niezgodności przeglądarek

- 
- Animacja zmian wprowadzonych do dokumentu – modyfikacja i tworzenie własnych animacji (nie tylko wbudowanych)
 - Uproszczenie zadań wykonywanych za pomocą JavaScriptu

Dlaczego to działa dobrze?

- Punkt wyjścia dla projektantów
- Obsługa dodatków
- Poziom abstrakcji normalizujący często wykonywane zadania = eliminacja niezgodności przeglądarek
- Praca ze zbiorami = niejawna iteracja = brak zbędnych pętli
- Wiele działań w jednym wierszu.

Główne zadanie jQuery i podstawy

- Podstawowym zadaniem jest wybranie części dokumentu
- Funkcja fabryczna `$()` załatwia wszystko
- Trzy składniki selektorów
- CSS: `#dowolny-id`
- jQuery: `$('#dowolny-id')`