

Pływająca ramka zamiast AJAX

Wpisany przez Patryk yarpo Jar
sobota, 28 listopada 2009 18:21

W ostatnim czasie AJAX bije wszelkie rekordy popularności. Tak - jest to wygodna "technologia" (w cudzysłowie, ponieważ to zbitek istniejących wcześniej technologii, a nie coś nowego).

Jednak czy zawsze trzeba używać Ajaksa? Nie. Choć często jest rozwiązaniem lepszym od tego, co mam zamiar tu zaprezentować.

Na swoim blogu niedawno opisałem dużo dokładniej ten pomysł - wraz z działającym kodem. Zapraszam: [Ajaks w oparciu o pływającą ramkę](#) .

Pływająca ramka - iframe

Pływająca ramka to taki "wycinek" strony, na który możemy wstawić inną stronę... Brzmi zawile. Może kod powie więcej:

```
{codecitation class='brush: html'}...<!-- tu nagłówek, sekcja head itp --> <body> <p>To jest na tej stronie</p> <iframe name=&quot;example&quot; src=&quot;druga_strona.htm&quot; width=&quot;200&quot; height=&quot;100&quot;>Ten tekst będzie widoczny, jeśli twoja przeglądarka nie obsługuje ramek</iframe> </body> </html>{/codecitation}
```

W prostokąt o rozmiarach 200x100px zostanie wczytana strona `druga_strona.htm`. I będzie żyć swoim własnym życiem. Czy możemy się do tej ramki jakoś dostać? Tak - wykorzystując JS (dla skrócenia kodu używam innerHTML, nie znaczy to, że to jest najlepszy i najwłaściwszy sposób).

Pobieranie danych

Aby pobrać dane z pływającej ramki można wykorzystać taki kod JS:

```
{codecitation class='brush: js'}window.frames['example'].document.innerHTML{/codecitation}
```

Przykładowe zastosowanie można zobaczyć na mojej stronie (wstęp do własnego edytora typu TinyMCE itp.): yarpo.republika.pl/studia/editor.

Nowy request HTTP

Pływająca ramka zamiast AJAX

Wpisany przez Patryk yarpo Jar
sobota, 28 listopada 2009 18:21

Po prostu otwieramy w ramcę nową stronę, np. tak:

```
{codecitation class='brush: html'}<a href="skrypt.php?id=2132&a=sdsd&b=323" target="example">Wykonaj</a>{/codecitation}
```

Ta strona zostanie wywołana w pływającej ramce. Następnie kodem, który pokazałem chwilę wcześniej odczytujemy wszystko co się tam wyświetliło i odpowiednio interpretujemy.

Ale to widać!

Starczy zatem zmniejszyć rozmiar pływającej ramki do 1x1 [nie jestem pewien, czy w każdej przeglądarce 0x0 będzie akceptowane], a następnie dodać taką regułę CSS (do [zewnętrznego pliku, albo do definicji stylów w w sekcji head](#)

):

```
{codecitation class='brush: css'}iframe[name="example"] { position: absolute; top: 0; left: 0; visibility: hidden; }{/codecitation}
```

i problem zniknął - razem z pływającą ramką.

Atrybut target

Pozwala nam sterować gdzie powinna się pokazać strona. Jest zakazany w XHTML-u oraz w HTML 4.01 strict. Słusznie, z resztą. Nie powinno się decydować za usera, gdzie On chce otworzyć stronę. Tu podajemy nazwę naszej ramki, dlatego skrypt php (a raczej wynik jego działania) pojawi się w pływającej ramce.

Czy warto to stosować?

Najczęściej nie. Dlaczego? Ponieważ nie mamy tu takich informacji jak status requesta, nie możemy wywołać tego tak banalnie asynchronicznie jak w przypadku Ajaksa. Czasem jednak może się okazać przydatna wiedza, że coś takiego istnieje.