

## Bermain Dynamic Tabel Row dengan menggunakan Javascript

20 06 2008

Jika kita adalah orang yang berkecukupan dengan dunia IT terutama bidang aplikasi web programming, pasti akan sering menjumpai proses insert, update, delete terhadap data dalam database, begitu juga dengan manipulasi tampilan table yang menampilkan data – data tersebut dalam script html yang kemudian dapat dilihat dalam browser.

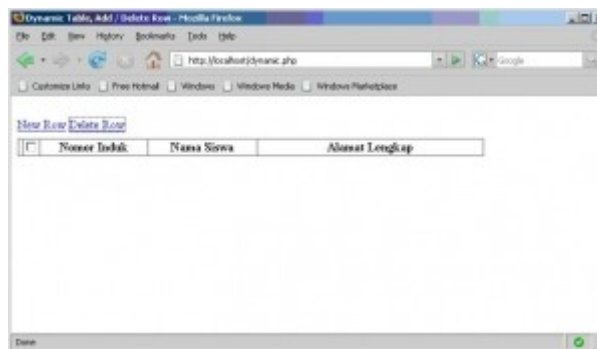
Berikut kita akan coba membahas mengenai proses “Add New Row” dan “Delete Row” terhadap suatu table dalam web-app menggunakan javascript, kenapa kita menggunakan javascript? Karena dengan kelebihan yang dimiliki oleh javascript memungkinkan kita untuk melakukan proses memasukkan baris baru atau menghapus baris dalam table tanpa melalui proses refresh, sehingga effort loading yang diperlukan untuk berinteraksi dengan server cukup sekali, jika proses edit data yang kita lakukan terhadap table tersebut selesai, dan data yang diinginkan untuk masuk ke database sudah benar maka interaksi dengan server dilakukan hanya untuk proses insert data.

Sekarang akan kita coba bahas tahapan yang harus dilakukan untuk membuat proses tersebut, sebagai basic kita akan coba membangunnya diatas php, untuk diketahui sebenarnya proses ini dapat dilakukan dalam berbagai web programming baik itu php, jsp, java ataupun dotnet, tetapi untuk saat ini kita hanya akan membahasnya diatas php. Sebagai permulaan kita buat dulu sebuah file \*.php untuk membentuk tampilan awal dari table dan link action yang akan dilakukan, baik itu link untuk menambah baris dan menghapus baris. Dengan script sebagai berikut:

```
<html>
<head>
<title>Dynamic Table, Add / Delete Row</title>
</head>
<body>
<div class="area">
<div class="toolbar-clean">
<a href="javascript:addNewRow();"><span>New Row</span></a>
<a href="javascript:deleteRow();"><span>Delete Row</span></a>
</div>
<table width="80%" cellpadding="0" cellspacing="0" border="1"
id="lineItemTable">
<tr>
<th width="0%"> </th>
<th width="2%" align="center">
<input type="checkbox" name="checkMaster" id="checkMaster"
onClick="clickAll();" />
</th>
<th width="24%">Nomor Induk</th>
<th width="24%">Nama Siswa</th>
<th width="50%">Alamat Lengkap</th>
</tr>
</table>
</div>
```

```
</body>  
</html>
```

Dari script diatas maka akan menghasilkan tampilan browser sebagai berikut:



Setelah kita dapatkan tampilan awal dari proses yang akan kita inginkan sebagai tahapan berikutnya, akan kita lakukan penambahan kode javascript ke dalam file php tersebut untuk memenuhi kebutuhan dari action "New Row" dan "Delete Row" yang akan kita sisipkan ke dalam tag script, yang pertama akan kita buat adalah function untuk New Row dengan nama function "addNewRow" disertai dengan seluruh function turunannya sebagai berikut:

```
function addNewRow() {  
    var tbl = document.getElementById("lineItemTable");  
    var row = tbl.insertRow(tbl.rows.length);  
  
    var td0 = document.createElement("td");  
    var td1 = document.createElement("td");  
    var td2 = document.createElement("td");  
    var td3 = document.createElement("td");  
    var td4 = document.createElement("td");  
  
    td0.appendChild(generateIndex(row.rowIndex));  
    td1.appendChild(generateCheckBox(row.rowIndex));  
    td2.appendChild(generateNomorInduk(row.rowIndex));  
    td3.appendChild(generateNomorRegister(row.rowIndex));  
    td3.appendChild(generateNamaSiswa(row.rowIndex));  
    td4.appendChild(generateItemName(row.rowIndex));  
  
    row.appendChild(td0);  
    row.appendChild(td1);  
    row.appendChild(td2);  
    row.appendChild(td3);  
    row.appendChild(td4);  
}  
  
function generateIndex(index) {  
    var idx = document.createElement("input");  
    idx.type = "hidden";  
    idx.name = "index[ ]";  
    idx.id = "index["+index+"]";  
    idx.value = index;  
  
    return idx;  
}
```

```
function generateCheckBox(index) {
    var check = document.createElement("input");
    check.type = "checkbox";
    check.name = "check[ ]";
    check.id = "check["+index+"]";

    return check;
}

function generateNomorInduk(index) {
    var idx = document.createElement("input");
    idx.type = "text";
    idx.name = "nomorInduk[ ]";
    idx.id = "nomorInduk["+index+"]";
    idx.size = "15";

    return idx;
}

function generateNomorRegister(index) {
    var idx = document.createElement("input");
    idx.type = "hidden";
    idx.name = "nomorRegister[ ]";
    idx.id = "nomorRegister["+index+"]";

    return idx;
}

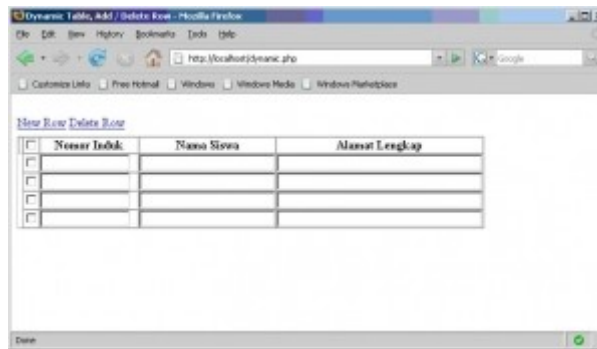
function generateNamaSiswa(index) {
    var idx = document.createElement("input");
    idx.type = "text";
    idx.name = "namaSiswa[ ]";
    idx.id = "namaSiswa["+index+"]";
    idx.size = "25";

    return idx;
}

function generateItemName(index) {
    var itemName = document.createElement("input");
    itemName.type = "text";
    itemName.name = "alamatSiswa[ ]";
    itemName.id = "alamatSiswa["+index+"]";
    itemName.size = "40";

    return itemName;
}
```

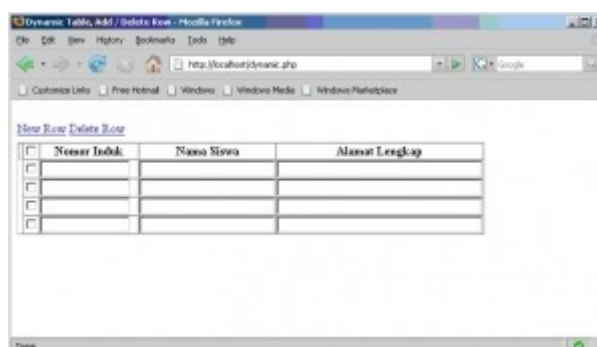
Dari contoh code javascript diatas dapat kita lihat suatu proses untuk membuat baris baru yang akan di generate jika link "New Row" seperti pada gambar diatas di Klik, susunan kolom yang akan generate telah disesuaikan dengan susunan header, sehingga jika proses telah dieksekusi maka tampilan dalam browser akan seperti gambar dibawah, dan baris baru akan terus bertambah jika link "New Row" terus di klik.



Setelah hasil seperti gambar diatas didapat maka tahapan dilanjutkan dengan menambahkan function ceklist semua checkbox yang ada dalam table sehingga jika kita ingin melakukan suatu action terhadap seluruh baris dalam tabel, kita tidak perlu direpotkan dengan harus melakukan ceklist satu per satu semua checkbox yang ada dalam tabel tersebut, function untuk melakukan proses tersebut sebagai berikut.

```
function clickAll() {  
    var checked = false;  
    if (document.getElementById("checkMaster").checked == true)  
        checked = true;  
  
    var tbl = document.getElementById("lineItemTable");  
    var rowLen = tbl.rows.length;  
    for (var idx = 1; idx < rowLen; idx++) {  
        var row = tbl.rows[idx];  
        var cell = row.cells[1];  
        var node = cell.lastChild;  
        node.checked = checked;  
    }  
}
```

Jika function "clickAll" yang sudah disisipkan ke source html dibawah source javascript sebelumnya, maka element checkbox yang terdapat di header dapat diasumsikan sudah berfungsi dengan baik. Dan jika checkbox tersebut diklik maka akan menghasilkan tampilan browser seperti berikut. Dapat kita lihat perbedaan yang dihasilkan dari gambar3 dan gambar2 dimana checkbox nya terclick dan tidak.



Dan tahapan terakhir yang harus dilakukan adalah menambahkan function "deleteRow" yang berfungsi untuk menghapus kembali baris dari tabel yang tidak diperlukan, fungsi ini juga dapat

digunakan untuk menghapus seluruh baris dalam tabel tersebut dengan syarat seluruh checkbox dalam baris tabel dalam kondisi ceklist aktif. Javascript code sebagai berikut.

```
function deleteRow() {
    var tbl = document.getElementById("lineItemTable");
    var error = false;

    if (document.getElementById("checkMaster").checked == false)
        error = true;

    var tbl = document.getElementById("lineItemTable");
    var rowLen = tbl.rows.length;
    for (var idx = 1; idx < rowLen; idx++) {
        var row = tbl.rows[idx];
        var cell = row.cells[1];
        var node = cell.lastChild;
        if (node.checked == true) {
            error = false;
            break;
        }
    }

    if (error == true) {
        alert("Checkbox tidak di cek, proses tidak dapat dilanjutkan");
        return;
    }

    if (document.getElementById("checkMaster").checked == true) {
        deleteAll();
        document.getElementById("checkMaster").checked = false;
    } else {
        var table = document.createElement("table");
        bufferRow(table);
        deleteAll();
        reIndex(table);
    }
}

function deleteAll() {
    var tbl = document.getElementById("lineItemTable");
    var rowLen = tbl.rows.length - 1;
    for (var idx = rowLen; idx > 0; idx--)
        tbl.deleteRow(idx)
}

function bufferRow(table) {
    var tbl = document.getElementById("lineItemTable");
    var rowLen = tbl.rows.length;
    for (var idx=1;idx<rowLen;idx++) {
        var row = tbl.rows[idx];
        var cell = row.cells[1];
        var node = cell.lastChild;
        if (node.checked == false) {
            var rowNew = table.insertRow(table.rows.length);

            var td0 = document.createElement("td");
            var td1 = document.createElement("td");
            var td2 = document.createElement("td");
            var td3 = document.createElement("td");
            var td4 = document.createElement("td");

            td0.appendChild(row.cells[0].lastChild);
```

```
        td1.appendChild(row.cells[1].lastChild);
        td2.appendChild(row.cells[2].lastChild);
        td3.appendChild(row.cells[3].firstChild);
        td3.appendChild(row.cells[3].lastChild);
        td4.appendChild(row.cells[4].lastChild);

        rowNew.appendChild(td0);
        rowNew.appendChild(td1);
        rowNew.appendChild(td2);
        rowNew.appendChild(td3);
        rowNew.appendChild(td4);
    }
}

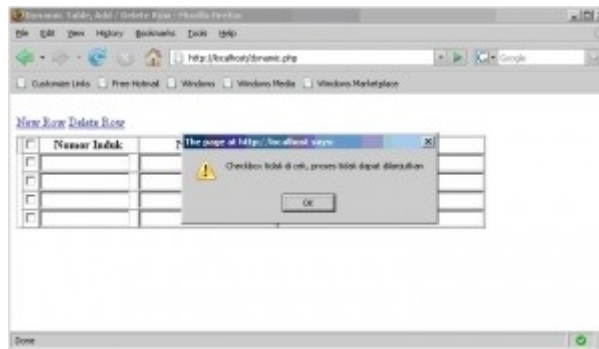
function reIndex(table) {
    var tbl = document.getElementById("lineItemTable");
    var rowLen = table.rows.length;
    for (var idx=0;idx < rowLen;idx++) {
        var row = table.rows[idx];
        var rowTbl = tbl.insertRow(tbl.rows.length);

        var td0 = document.createElement("td");
        var td1 = document.createElement("td");
        var td2 = document.createElement("td");
        var td3 = document.createElement("td");
        var td4 = document.createElement("td");

        td0.appendChild(row.cells[0].lastChild);
        td1.appendChild(row.cells[1].lastChild);
        td2.appendChild(row.cells[2].lastChild);
        td3.appendChild(row.cells[3].firstChild);
        td3.appendChild(row.cells[3].lastChild);
        td4.appendChild(row.cells[4].lastChild);

        rowTbl.appendChild(td0);
        rowTbl.appendChild(td1);
        rowTbl.appendChild(td2);
        rowTbl.appendChild(td3);
        rowTbl.appendChild(td4);
    }
}
```

Dari javascript function "deleteRow" diatas dapat kita lihat sebelum fungsi tersebut melakukan perintah delete pada baris perintah 04 sampai 22 dicodekan perintah validasi untuk memeriksa apakah seluruh element checkbox di ceklist atau tidak, minimal jika 1 saja checkbox kondisi ceklist maka perintah menghapus baris dapat dilaksanakan, tetapi jika tidak ada satupun checkbox yang ceklist maka proses dihentikan dan akan menampilkan tampilan di browser seperti berikut.



Mungkin saat ini hanya sekian ilmu yang bisa saya bagi dengan pembaca semoga bermanfaat, jika ada kekurangan atau masukan yang dapat meningkatkan kemampuan jangan sungkan untuk memberikan komentar, berikut source code lengkap dari proses diatas dalam satu file berupa file pdf yang dapat diambil di [Source Code](#)