Tutorial Jquery: $.change, $.val and $.css - Developer News

NEWS_PDF_AUTHOR: kaotik

NEWS_PDF_DATE: 2009/11/10 13:01:21

Learn how to detect changes on your forms using jquery and react accordingly using $.change

Jquery is great for detecting user actions and performing reactions based on these.

Let's take a simple form to start:

In this form we have a textfield and a checkbox.

STEP 1 - Check if checked

In our first test, when the checkbox is checked we want some text to appear in the textfield.
There are 2 things we need for this:
1- wait for a change on checkbox "mybox". To acomplish this we will use a jquery function called $.change This detects any change on a particular selector.
2- If change does occur, place text inside textfield "mytext". For this we will combine a simple IF conditional with jquery function $.val used to set the value of a selector.

```
$(document).ready(function() { //Finish loading the entire page before processing any javascript

$("#mybox").change(function () {
if ($('#mybox:checked').val() ) {
$("#mytext").val("yes");
} else {
$("#mytext").val("no");
}
});

});
```

Now, as you click the checkbox, it will change the textfield value between "yes" and "no". Simple isn't it?

STEP 2 - Changing css through jquery

For this next step we will use our checkbox to change some css of our form.

```
$(document).ready(function() { //Finish loading the entire page before processing any javascript

$("#mytext").css("border","medium solid green"); // initial color for my border: green

$("#mybox").change(function () {
if ($('#mybox:checked').val() ) {
$("#mytext").val("yes");
$("#mytext").css("border","medium solid blue"); //if checked apply blue
} else {
$("#mytext").css("border","medium solid red"); //if unchecked apply red
$("#mytext").val("no");
}
});

});
```

You will see 3 new lines here. The first one:
$("#mytext").css("border","medium solid green"); // initial color for my border: green
defines our initial color for the border. I've applied a border to the form element "mytext" but I also could have applied it to any "div", "p", etc.
Note:

You can bind actions to selectores using their ID or class. For an id you would use: ("#mytext") , for a class you would use (".mytext")

Learn how to detect changes on your forms using jquery and react accordingly using $.change

Jquery is great for detecting user actions and performing reactions based on these.

Let's take a simple form to start:

In this form we have a textfield and a checkbox.

STEP 1 - Check if checked

In our first test, when the checkbox is checked we want some text to appear in the textfield.
There are 2 things we need for this:
1- wait for a change on checkbox "mybox". To acomplish this we will use a jquery function called $.change This detects any change on a particular selector.
2- If change does occur, place text inside textfield "mytext". For this we will combine a simple IF conditional with jquery function $.val used to set the value of a selector.

```
$(document).ready(function() { //Finish loading the entire page before processing any javascript

$("#mybox").change(function () {
if ($('#mybox:checked').val() ) {
$("#mytext").val("yes");
} else {
$("#mytext").val("no");
}
});

});
```

Now, as you click the checkbox, it will change the textfield value between "yes" and "no". Simple isn't it?

STEP 2 - Changing css through jquery

For this next step we will use our checkbox to change some css of our form.

```
$(document).ready(function() { //Finish loading the entire page before processing any javascript

$("#mytext").css("border","medium solid green"); // initial color for my border: green

$("#mybox").change(function () {
if ($('#mybox:checked').val() ) {
$("#mytext").val("yes");
$("#mytext").css("border","medium solid blue"); //if checked apply blue
} else {
$("#mytext").css("border","medium solid red"); //if unchecked apply red
$("#mytext").val("no");
}
});

});
```

You will see 3 new lines here. The first one:
$("#mytext").css("border","medium solid green"); // initial color for my border: green
defines our initial color for the border. I've applied a border to the form element "mytext" but I also could have applied it to any "div", "p", etc.
Note:
You can bind actions to selectores using their ID or class. For an id you would use: ("#mytext") , for a class you would use (".mytext")