

## 15/5/20 Adding Event Handling to DOM Elements through JQuery methods

following are the pre-defined methods provided by the jquery which can be applied on DOM elements to handle events.

element.click();  
element.focus();  
element.change();  
element.blur();  
element.mouseover();  
etc.

eg: `$("#abc").click(<callback method>);`

"on()" is a method been provided by jquery through which we can add event handling dynamically to DOM elements.

Syntax: `element.on(<type of event>, <call back method>);`

eg: `$("#abc").on("click", function(event) {  
...  
// code to handle event  
})`

\* Using "on" method we could able to add multiple events at a time to single DOM element.

Syntax:

```
element.on({  
  "eventtype": callback,  
  "eventtype2": call back,  
  etc.  
});
```

Eg: `Element.on({  
 "click": function(event) {  
 ...  
 },  
 "focus": function(event) {  
 ...  
 },  
});`

### Jquery Animation:

Adding animation to DOM using jquery methods  
following are the different ways we could  
add animation to DOM elements.

1. Sliding
2. fade
3. hide/show/toggle

#### ① Hide/show:

```
element.hide(<delay>, <optional call back>);
```

Eg: `$("#abc").show(5000, function() {`



...  
// this will be called after the animation  
is completely done.

})

### ② Sliding:

```
element.slideUp(delay, callback);  
element.slideDown(delay, callback);  
element.slideToggle(delay, callback);
```

### ③ fading:

```
element.fadeIn(delay, callback);  
element.fadeOut(delay, callback);  
element.fadeToggle(delay, callback);
```

### Master Animation method.

• animate method is supported in jquery,  
which can take multiple CSS properties and  
add to DOM elements with provided delay  
time.

### Syntax:

```
element.animate({  
  css prop 1 : value;  
  css prop 2 : value;  
  ...  
}, delay time);
```

Eg:

```
element.animate({  
  width : 200px;  
  height : 100px;
```

Opacity: 0.8px;

}, 5000);

16/5/20

## Query Traversing Methods

following are pre-defined methods through which we could traverse between DOM elements.

### ① Traversing Ancestors:

- `parent()` - Returns the immediate parent of any DOM element.
- `parents()` - Returns all parents until HTML tag.
- `parentsUntil("pattern")` - Returns all the parents until it finds a parent with given pattern.

### ② Traversing Siblings:

- `siblings()` - Returns the list of all siblings of any DOM element.
- `next()` - Returns reference of next sibling element.
- `prev()` - Returns reference of previous sibling element.
- `prevAll()` - Returns reference of all previous elements.
- `nextAll()` - Returns reference of all next siblings.



- nextUntil (T<sup>pattern</sup>) - Returns all next siblings until the provided pattern.
- prevUntil (pattern) - Returns all previous sibling elements.

### ③ Traversing Decendants:

- children () - Returns the children of any DOM element.
- find ("\*") - Returns every children of a DOM element.