

**/\* ##### jQuery Examples / jQuery Örnekleri ##### \*/**  
<https://api.jquery.com/>

**/\* javascript HTML - DOM Erişimi - Temel javascript \*/**

document.getElementById("id") :id (attribute) ile bul,  
document.getElementsByClassName("class")[i] :class (attribute) ile bul,  
document.getElementsByName("name")[i] :name (attribute) ile bul,  
document.getElementsByTagName("element")[i] :tag (element) ile bul,  
document.querySelector("css selectors") :ilk eşleşeni bul,  
document.querySelectorAll("css selectors")[i] :eşleşeni bul,

.nodeName :node adı,  
.nodeType :node tipi (1:element, 2:attribute, 3:text),  
.nodeValue :node değeri,  
.parentNode.nodeName :bir üst node'un node adı  
(node tipi element olanın bir üst node tipi element olacaktır),  
.tagName :tag (element) adı,  
.id :elemente ait id (attribute),  
.className :elemente ait class (attribute),  
.attributes[i].name :elemente ait attribute adı,  
attributes[i].value :elemente ait attribute değeri,

.getAttribute("attribute adı") :attribute değeri,  
.setAttribute("attribute adı", "attribute değeri") :attribute oluştur,  
.removeAttribute("attribute adı") :attribute sil,  
.hasAttribute("attribute adı") :var ise "true", yok ise "false" dondurur,

.style :style (attribute), CSS değerleri,  
get ve set işlemleri için kullanılır, sonuna style object eklenir,

.innerHTML :element içerisindeki html metin,  
.innerText :element içerisindeki metin (style ve script den etkilenir),  
.outerHTML :elementin kendisi ve element içerisindeki html metin,  
.outerText :elementin kendisi ve element içerisindeki metin (style ve script den etkilenir),  
.textContent :element içerisindeki düz metin (style ve script den etkilenmez).

**/\* jQuery Selectors / jQuery Seçicileri \*/**

\$(".\*") :hepsi,  
\$(this) :şimdiki,  
\$("element") :element,  
\$("#id") :id,  
\$(".class") :class,  
\$("element.class") :element.class,  
\$(".class,class") :class veya class,  
\$("element,element") :çoklu secim,  
\$("element element") :torun,  
\$("element > element") :çocuk,  
\$("element + element") :bitişik,  
\$("element ~ element") :kardeş,  
\$("[at]") :attribute,  
\$("element:[at^~\*\$|!='v']") :attribute value,  
\$(" :header") :header h1,h2,h3,h4,h5 ve h6,

\$( "element:first" ) :birinci,  
\$( "element:last" ) :sonuncu,  
\$( "element:even" ) :çift,  
\$( "element:odd" ) :tek,

\$( "element:first-child" ) : ilk çocuk  
\$( "element:nth-child(n)" ) : n. çocuk,  
\$( "element:only-child" ) : tek çocuk,  
\$( "element:nth-last-child(n)" ) : sondan n. çocuk,  
\$( "element:last-child" ) : son çocuk,

\$( "element:first-of-type" ) : ilk tür,  
\$( "element:nth-of-type(n)" ) : n. tür,  
\$( "element:only-of-type" ) : tek tür,  
\$( "element:nth-last-of-type(n)" ) : sondan n. tür,  
\$( "element:last-of-type" ) : son tür,

(Index n 0'dan başlar)

\$( "element:eq(n)" ) :eşit n,  
\$( "element:gt(n)" ) : n'den büyük,  
\$( "element:lt(n)" ) :n'den küçük,

\$( "element:target" ) :hedef,  
\$( "element:not(selector)" ) :değil,  
\$( ":animated" ) :animasyonlu,  
\$( ":focus" ) : odaklanmış,  
\$( ":contains('text')" ) :metni içeren,  
\$( "element:has(selector)" ) :içeren,  
\$( ":empty" ) :bos,  
\$( ":parent" ) :ebeveyn,  
\$( ":hidden" ) :gizli,  
\$( ":visible" ) :görünür,  
\$( ":root" ) :kok(html),  
\$( "element:lang(lng)" ) :dil,

#### **Input:**

\$( ":input" ) : tüm input,  
\$( ":text" ) : text,  
\$( ":password" ) : password,  
\$( ":radio" ) : radio,  
\$( ":checkbox" ) : checkbox,  
\$( ":submit" ) : submit,  
\$( ":reset" ) : reset,  
\$( ":button" ) : button,  
\$( ":image" ) : image,  
\$( ":file" ) : file,

#### **Checkbox:**

\$( ":enabled" ) : aktif,  
\$( ":disabled" ) : pasif,  
\$( ":selected" ) : seçili,  
\$( ":checked" ) işaretli.

## **/\* jQuery Events (function) / jQuery Olayları \*/**

.click(func) :fare tıklama,  
.dblclick(func) :fare çift tıklama,  
.keypress(func) :düğmeye basma,  
.mouseenter(func) :imleç girme(torunlar hariç),  
.mouseleave(func) :imleç çıkma(torunlar hariç),  
.hover(infunc, outfunc) :imleç girme çıkma(torunlar hariç),  
.mouseover(func) :imleç girme,  
.mouseout(func) :imleç çıkma,  
.mousemove(func) :imleç geçme,  
.mousedown(func) :fare düğme basma,  
.mouseup(func) :fare düğme çekme,  
.keydown(func) :klavye düğme basma,  
.keyup(func) :klavye düğme çekme,

.focus(func) :odaklanma,  
.blur(func) :odaktan çıkma,  
.focusin(func) :odağı kazanma,  
.focusout(func) :odağı kaybetme,

.change(func) :değişim (input, textarea, select için),  
.select(func) :metin seçimi(text filed, text area için),  
.submit(func) :form onaylama,  
.resize(func) :boyutu değişimi,  
.scroll(func) :scroll oynatma,  
.one(event,data,func) :bir olay bir kere tetikleme,

.load() :script yüklendiğinde (resim gibi),  
.unload() :script yüklenmediğinde,  
.ready(func) :dom hazır olduğunda,  
.bind(event,data,func,map) :olay tetikleme,  
.unbind(event,func,map) :son tetiklenen olayı silme,  
.on(event,selector,data,func,map) :birden fazla olay tetikleme,  
.off(event,selector,func,map) :tetiklenen olayları iptal etme,  
.error() :hata olduğunda,  
.trigger(event,eventObj,param1,param2,...) :bir olayı (kendisi hariç) tetikleme,  
.triggerHandler(event,param1,param2,...) :bir olayı tetikleme.

## **/\* jQuery Effects / jQuery Efektler \*/**

.hide(speed,callback) :gizle,  
.show(speed,callback) :göster,  
.toggle(speed,callback) :gizle veya göster,

.fadeTo(speed,opacity,callback) :opacity 0 - 1 arasında soldurma,  
.fadeIn(speed,callback) :keskinleştirme/gösterme,  
.fadeOut(speed,callback) :soldurma/gizleme,  
.fadeToggle(speed,callback) :keskinleştirme veya soldurma,

.slideDown(speed,callback) :açılma/gösterme,  
.slideUp(speed,callback) :kapanma/gizleme,  
.slideToggle(speed,callback) :açılma veya kapanma,

.animate({params},speed,callback) :animasyon uygula,  
.stop(stopAll,goToEnd) :efekti durdur,  
.delay() :beklet(s ver),

.queue(queueName) :animasyon sıraya ekle (varsayılan qN fx),  
.dequeue(queueName) :animasyon sıradan sil,  
.clearQueue(queueName) :animasyon sırayı sil,

callback function: efekt tamamlandığında gerçekleştirilecek devam fonksiyonu,  
speed: milliseconds, "slow" veya "fast seçilebilir.

### **/\* jQuery Manipulating HTML / jQuery HTML İçerik Değişikliği \*/**

.text("text",callback) :elementin metini,  
.html("html text",callback) :elementin html yapısı,  
.val("value",callback) :elementin değeri,

.attr("attribute","attr value",callback) :attribute varsayılan değeri,  
.removeAttr("attribute",callback) :attribute silme,

.replaceAll("selector",callback) :seçili elementleri değiştirme,  
.replaceWith(content,function(index)) :seçili elementleri içerikle değiştirme,

.append("html text",callback) :iç sonuna veri eklemek,  
.prepend("html text",callback) :iç önüne veri eklemek,  
.after("html text",callback) :sonuna veri eklemek,  
.before("html text",callback) :önüne veri eklemek,

.appendTo("selector",callback) :ic sonuna element eklemek,  
.prependTo("selector",callback) :ic önüne element eklemek,  
.clone("selector") :element kopyalamak,  
.insertAfter("selector",callback) :sonuna element eklemek,  
.insertBefore("selector",callback) :önüne element eklemek,

.remove("selector") :seçili elementleri silmek,  
.empty("selector") :seçili elementlerin tortularını silmek,  
.detach("selector") :seçili elementleri hafızaya alarak silmek,

.wrap("element",func) :elementleri tek tek sarmak,  
.unwrap() :sarmayı silmek,  
.wrapAll("element",func) :elementleri topluca sarmak,  
.wrapInner("element",func) :elementleri içeriden sarmak.

### **/\* jQuery Manipulating CSS / jQuery CSS İçerik Değişikliği \*/**

.addClass("class",callback) :class eklemek,  
.removeClass("class",callback) :class silmek,  
.toggleClass("class",callback) :class eklemek veya silmek,  
.hasClass("class") :class var ise true, false,

.css("propertyname","value",callback) :style değeri,  
.css({"propertyname":"value","propertyname":"value",...}) :çoklu style,  
.prop(property,value,callback) :değer atama,  
.removeProp(property,callback) :atanan değeri silme,

### **Genislik Degeri:**

.width() :element,  
.innerWidth() :element + padding,  
.outerWidth() :element + padding + border,  
.outerWidth(true) :element + padding + border +margin,

### **Yukseklk Degeri:**

.height() :element,  
.innerHeight() :element + padding,  
.outerHeight() :element + padding + border,  
.outerHeight(true) :element + padding + border + margin,

.position() :konum tespiti,  
.offset({top:value,left:value}) :konum atama,  
.offsetParent() :ana elemanın konumu,

.scrollLeft() :yatay scroll pozisyonu,  
.scrollTop() :düşey scroll pozisyonu,  
.val(value) :input değeri,

\$(document): doküman (html body),  
\$(window) :aktif pencere,  
\$(document).ready(function(){...});  
Kısaltması: \$(function){...};

### **/\* jQuery Manipulating DOM / jQuery DOM İçerik Değişikliği \*/**

.parent() :ebeveyn  
.parents() :ebeveyn ve atalar,  
.parentsUntil("selector") :seçili olana kadar(hariç) ebeveyn ve atalar,  
.closest("selector") :en yakın ebeveyn veya ata,  
.children() :çocuklar,  
.find("selector") :torunlar,  
.find("\*") :tüm torunlar,  
.siblings() :kendisi hariç kardeşler,  
.next() :sonraki kardeş,  
.nextAll() :sonraki kardeşler,  
.nextUntil("selector") :seçili olana kadar(hariç) sonraki kardeşler,  
.prev() :önceki kardeş,  
.prevAll() :önceki kardeşler,  
.prevUntil("selector") :seçili olana kadar(hariç) önceki kardeşler,  
.first("selector") :ilk,  
.last("selector") :son,  
eq("n") :indeks seç (indeks n 0'dan baslar),  
.slice(startn,stopn) :indeks arası seç,  
.even() :çift indeks numaralı seç,  
.odd() :tek indeks numaralı seç,  
.each(func) :her bir,  
.has() :içeriğinde bulunduran,  
.is("selector",func) :kendisi,  
.filter("selector") :kriterleri seç,  
.not("selector") :olmayanı seç.

### **/\* jQuery Misc / jQuery Cesitli \*/**

\$.noConflict(); veya var jQuery = \$.noConflict(); :"\$" isareti yerine "jQuery" kelimesi kullanarak kısaltma yapmak,

\$.jquery :jQuery versiyon numarası,  
.data(name,value) :elemente data ekleme,  
.removeData(name) :elementten data silme,  
.get(n) :indekse göre dom bilgisi alma,  
.index() :indeks numarasını tespit eder,  
.length :element sayısı.

### **/\* jQuery Ajax / jQuery Veri Gönderme \*/**

```
$("#form#id").submit(function() {
    $.ajax({ type: "POST", url: "send.php", data: $("#form#id").serialize(), success: function(result) {

        // "form#id" yerine formun id'si verilebilir. Başka bir seçici kullanılabilir.
        // "POST" (gizli) yada "GET" (açık) seçilebilir. "send.php" yerine verilerin gönderilmesi talep edilen
        adres girilebilir.

        // console.log($("#form#id").serialize()); // Test amaçlı yazılır.
        // serialize consolda bu şekilde string olarak gözükür:
        name1=value1&name2=value2&name3=value3&name4=value4

        // serialize yerine bu şekilde kullanılabilir:
        "name1=value1&name2=value2&name3=value3&name4=value4"
        // serialize yerine bu şekilde de kullanılabilir: { name1: "value1", name2: "value2", name3:
        "value3", name4: "value4" }

        // serialize devamına ilave yapılabilir: $("#form#id").serialize()+"&name5=value5&name6=value6"
        // ilaveli hali bu şekilde olacaktır:
        "name1=value1&name2=value2&name3=value3&name4=value4&name5=value5&name6=value6"

        $("#div-response#id").html(result); // Cevabı ekrana getirir.

        // "div-response#id" yerine cevabin görüntülenmesi talep edilen elementin id'si verilebilir. Başka
        bir seçici kullanılabilir.

        // Talebe göre diğer işlemler eklenebilir. Cevap A olarak geldi ise A işlemini yap.

        // Buton isleme duyarlı hale getirilebilir.

        $("#button[type='submit']").attr("disabled", "disabled"); // Butonu pasif yap
        setTimeout(function() {
            $("#button[type='submit']").removeAttr("disabled"); // Butonu 1 saniye sonra aktif yap
        }, 1000);

    });
});
```