|  |
| --- |
| **JQuery** |
| **1. Khái niệm** |
| - Jquery là một thư viện của javascript  - Các module của JQuery bao gồm:   * Ajax - Xử lý Ajax tiện lợi hơn * Atributes - Xử lý thuộc tính của DOM object * Effect - Xử lý hiệu ứng giữa HTML và CSS * Form - Xứ lý sự kiện liên quan đến submit Form * DOM - Truy vấn DOM dễ dàng hơn * Selector - Truy vấn các thẻ HTML thông qua cú pháp selector của CSS |
| **2. Cài đặt** |
| - Truy cập vào: https://code.jquery.com/  - Vào phiên bản minified của bản mới nhất về  - Chèn đoạn script vào file.  Vd: minified của phiên bản: jQuery 3.x  <script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha256-/xUj+3OJU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4=" crossorigin="anonymous"></script>  - Hoặc an toàn hơn down toàn bộ file theo địa chỉ  src=<https://code.jquery.com/jquery-3.6.0.min.js>  lưu lại và dẫn liên kết tới file |
| **3. JQuery Selector** |
| Cú pháp: **$(‘** *css\_selector* **‘ )**   |  |  | | --- | --- | | **Cú pháp** | **Ý nghĩa** | | **$(‘ \* ‘)** | Chọn tất cả thẻ HTML | | **$(‘ .class-name‘)** | Chọn các thẻ theo class-name | | **$(‘element ‘)** | Chọn các thẻ theo element | | **$(‘ #id-name‘)** | Chọn thẻ theo id | | **$(‘ #id-name , .class-name ‘)** | Chọn các thẻ có id **hoặc** class-name , chỉ cần có id hoặc class-name trùng là chọn | | **jQuery Basic Filter Selector**  **(Lọc dựa trên điều kiện lọc)** | | | **$(‘ selector*:animated* ‘)** | Chọn các selector đang chạy hiệu ứng animation | | **$(‘ selector*:eq(*** index ***)* ‘)** | Chọn thẻ có số thứ tự là **index** trong mảng tất cả các selector tìm được | | **$(‘ selector*: qt(*** index ***)* ‘)** | Chọn các thẻ có thứ tự **lớn hơn index** | | **$(‘ selector*: lt(*** index ***)* ‘)** | Chọn các thẻ có thứ tự **bé hơn index** | | **$(‘ selector*:odd* ‘)** | Chọn các thẻ đứng ở vị trí **CHẴN** | | **$(‘ selector*:even* ‘)** | Chọn các thẻ đứng ở vị trí **LẺ** | | **$(‘ selector*:first* ‘)** | Chọn thẻ đứng ở vị trí **ĐẦU TIÊN** | | **$(‘ selector*:last* ‘)** | Chọn thẻ đứng ở vị trí **CUỐI CÙNG** | | **$(‘ selector*:focus* ‘)** | Chọn thẻ đang được **focus** | | **$(‘ selector*:header* ‘)** | Chọn các thẻ là **H1, H2, H3 …** trong selector | | **$(‘ selector*:not(*** filter ***)*‘ )** | Chọn các thẻ **không có filter** trong selector;  vd: $(‘ span:not(.active) ‘)  - Chọn các thẻ span không có class=’active’ | | **$(‘ selector*:lang(*** value ***)* ‘)** | Chọn thẻ trong selector có lang=’value’ | | **jQuery Child Filter Selector**  **(Lọc các thẻ con trong thẻ cha)** | | | **$(‘ selector*:first-child* ‘)** | Chọn thẻ thuộc selector là **first-child**  - Trong selector chỉ có **một :first hoặc :last** nhưng có thể **có nhiều :first-child và nhiều :last-child** | | **$(‘ selector*:last-child* ‘)** | Chọn thẻ thuộc selector là **last child** | | **$(‘ selector*:nth-child(*** index ***)* ‘)** | - Chọn các thẻ trong Selector là thẻ con thứ index | | **$(‘ selector*:last-nth-child(*** index ***)* ‘)** | - Chọn các thẻ trong Selector là thẻ con thứ index từ dưới lên | | **$(‘ selector*:only-child* ‘)** | - Chọn thẻ selector là **con duy nhất** | | **$(‘ Tagname*:first-of-type* ‘)** | - Chọn các thẻ thuộc Tagname là ***first type*** trong thẻ cha của nó | | **$(‘ Tagname*:last-of-type* ‘)** | - Chọn các thẻ thuộc Tagname là ***last type*** trong thẻ cha của nó | | **$(‘ Tagname*:nth-of-type(*** index ***)* ‘)** | - Chọn thẻ vị trí **index** trong tập hợp các thẻ thuộc Tagname nằm trong thẻ cha | | **$(‘ Tagname*:nth-last-of-type(*** index ***)*‘)** | - Chọn thẻ vị trí **index từ dưới lên** trong tập hợp các thẻ thuộc Tagname nằm trong thẻ cha | | **$(‘ Tagname*:only-of-type*‘)** | - Chọn thẻ tagname là con duy nhất | | - Có thể thay thế index bằng các giá trị khác:   * even : số chẵn * odd : số lẻ * 2n+1 : Các vị trí: 1,3,5,7,… ( n chạy từ 0 🡺 ∞) * 3n+1 : Các vị trí: 1,4,7,…. | | | **jQuery content filer selector**  **(Lọc dựa trên nội dung bên trong)** | | | **$(‘ selector*:contain( ‘* value*’ )* ‘)** | Chọn thẻ nội dung có chứa ‘value’  - giống contain trong String | | **$(‘ selector*:empty )* ‘)** | Chọn thẻ có nội dung trống | | **$(‘ selector*:has(*** *Tagname* ***)* ‘)** | Chọn thẻ có chứa thẻ khác | | **$(‘ selector*:parent* ‘)** | Chọn thẻ là không rỗng | | **jQuery form filer selector**  **(Lọc dựa trên các thuộc tính của thẻ trong form)** | | | **$(‘ :button ‘)** | Type=”button” hoặc thẻ <button> | | **$(‘ :checkbox ‘)**  **$(‘ :radio ‘)**  **$(‘ :submit ‘)**  **…….** | Type=”checkbok”  Type=”radio”  Type=”submit”  …… | | **$(‘ :checked ‘)** | Chọn các thẻ được check | | **$(‘ :disable ‘)** | Chọn các thẻ bị disable | | **$(‘ :enable ‘)** | Chọn các thẻ bị enable | | **$(‘ :selected ‘)** | Chọn các thẻ được select ( trong option) | | **$(‘ :focus ‘)** | Chọn các thẻ được focus | | **jQuery form attribute selector**  **(Lọc dựa trên thuộc tính)** | | | **$(‘ [ attName = “ value ” ] ’)** | Chọn các thẻ có thuộc tính **attName** có giá trị là **value** | | **$(‘ [ attName |= “ value ” ] ’)** | giá trị của **attName** bắt đầu bằng **value** | | **$(‘ [ attName \*= “ value ” ] ’)** | giá trị của **attName** có chứa chuỗi **value**  vd: “value” & “value-test” & “value test” đều được | | **$(‘ [ attName ~= “ value ” ] ’)** | giá trị của **attName** có chứa chữ **value**  vd: “value” & “value test” được  “value-test” sẽ không được | | **$(‘ [ attName != “ value ” ] ’)** | giá trị của **attName** khác **value** | | **$(‘ [ attName] ’)** | Chọn thẻ chỉ cần có thuộc tính **attName** | | **Vài cái còn lại** | | | **$(‘ :hidden ‘)** | Chọn thẻ bị hidden | | **$(‘ :visible ‘)** | Chọn thẻ đang visible | |
| **4. jQuery Events** |
| Khác với Javascript thuần ta có các sự kiện như kiểu: onclick() , onchange() , ….  thì với JQuery ta bỏ chữ on đi: **click() , change() , ……**  **- Thêm sự kiện trong JQuery : (** tương tự addEventListener() )  **Hàm : on( )**  **$(‘** selector **’).on(** ‘event\_name ‘**,** function( ){ } **)**  vd: **$(‘** #btn **’).on(** ‘click ‘**,** function show( ){ } **)**  **- Xoá sự kiện trong JQuery : (** tương tự removeEventListener() )  **Hàm : off( )**  **$(‘** selector **’).off(** ‘event\_name ‘**, ‘selector’ ,** actionName **)**  vd: **$(‘** #btn **’).off(** ‘click ‘**,** ‘#btn’ , show **)** |
| **5. jQuery Attributes:**  **addClass( ) & removeClass( ) & hasClass( )** |
| **.addClass( ‘***className* **‘)** : Thêm className vào class cho đối tượng  **.removeClass(** ‘*className* ‘) : Xoá className vào class cho đối tượng  **.hasClass(** ‘*className* ‘) : Kiểm tra className có trong đối tượng hay không  Trả về TRUE / FALSE |
| **6.jQuery Attributes:**  **attr( ) & prop( ) & removeAttr( )** |
| **.attr( ‘attributeName ‘)** : Dùng lấy giá trị của thuộc tính  **.attr( ‘attributeName ‘ , ‘ value ‘)** : Dùng gán giá trị của thuộc tính  **.prop( ‘attributeName ‘ )** : Dùng lấy giá trị của thuộc tính  chủ yếu lấy giá trị true / false của các thuộc tính như checked , active , hidden …..  **.removeAttr( ‘attributeName ‘)** : Dùng xoá thuộc tính |
| **7.jQuery Effect: show( ) & hide( )** |
| **.hide( )** : Thiết lập thuộc tính **style:” display = none”** cho đối tượng  **.show( )** : Khôi phục lại trạng thái **display** của đối tượng  Có thể truyền thêm 2 giá trị là **speed** và **callback** vào trong 2 hàm trên:  **.hide( speed , callBack )** và **show( speed , callBack )**  - **speed** : Tốc độ ẩn và hiện của đối tượng  miliseconds , “fast” = 200 | “slow” = 1000  - **callBack** : hàm sẽ thực hiện sau khi hàm hide hoặc show thực hiện xong |
| **8.jQuery Effect: slideUp( ) & slideDown( )** |
| **.slideUp(speed , easing , callBack)** : Biến mất với hiệu ứng trượt lên  **.slideDown(speed , easing , callBack)** : Hiện ra với hiệu ứng trượt xuống  easing - chế độ hiện ra có 2 kiểu là :  **“swing”** : Nhanh ở khúc đầu và cuối  **“linear”** : Tốc độ giống nhau |
| **9.jQuery Effect: fadeIn( ) & fadeOut( )** |
| **.fadeOut(speed , easing , callBack)** : Biến mất với hiệu ứng mờ dần  **.fadeIn(speed , easing , callBack)** : Hiện ra với hiệu ứng hiện dần |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |