

19CSCI04I

Internet Programming

**Lab (5)** – JQuery and Form Validation

**JQuery**

Introduction to JQuery:

JQuery is a popular open source JavaScript Library that simplifies the interaction between HTML and JavaScript, it provides many advanced and cross-browser functions that can enhance the web applications.

Moreover, JQuery is a (write less, do more!) JavaScript library, which means that JQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

Installing JQuery:

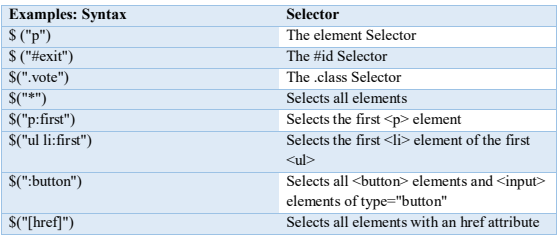
Two ways to add JQuery to web pages:

1. Download the JQuery library from jQuery.com
2. Include JQuery from a Content Delivery Network (CDN), like Google or Microsoft. (You can use these CDNs instead of downloading and hosting JQuery yourself). The following line shows how to include Google CDN:

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>

Selecting Elements with JQuery:

* $ is a JQuery function (also named jQuery), then a selector is passed to this function: $(selector).action( ), the selector is a way to find HTML elements.
* JQuery uses CSS syntax to select elements.
* You can select HTML elements based on their name, id, classes, types, attributes, or values of attributes.
* JQuery has a bunch of custom selectors beside the CSS selectors (such as :first, :last, :has( ), :visible, :hidden)
* JQuery provides a full suite of methods for walking the DOM tree, .parent( ), .next( ), .prev(), .children( ), .siblings( ) and many more.



**Document Ready Event:**

JQuery scripts do not run when loaded, you are able to use JQuery in your Javascript, but you cannot assign any event listeners. Therefore you will need to use **$(document).ready( )**

$(document).ready() launches when the document is ready, and it is loaded at the end of the page. You would generally write all your JQuery in this section in the following manner.

**$(document).ready(function( ){**

**// Your jQuery code goes in here**

**});**

Side Note: .ready() uses a fundamental JS property, that is being a functional language capable of accepting functions as parameters to other functions, you can write your function somewhere else and pass it to .ready() function. You will see this property being commonly repeated in JQuery and other JS framework.

Now the elements we’re looking for have been found, it’s time to change them!

* Attributes (addClass, attr)
* CSS (css)
* Manipulation (append, prepend, remove, clone)
* Events (click, hover, toggle)
* Effects (hide, show, slideDown, fadeOut)

**JQuery Events:**

JQuery has some methods that handle DOM events such as clicking on a button, moving the mouse over some element, double click on some element and much more DOM events.

**For example:**

The following line assigns click event to any element with a class vote.

**$(".vote").click;**

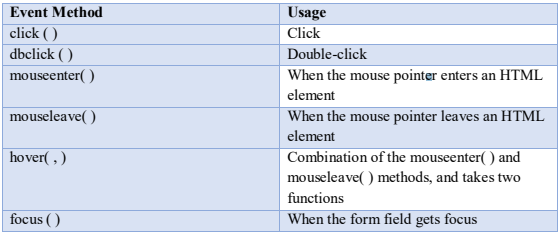
Now we defined what should happen when the click event fires (hide the clicked element) **$(".vote").click(function( )**

**{**

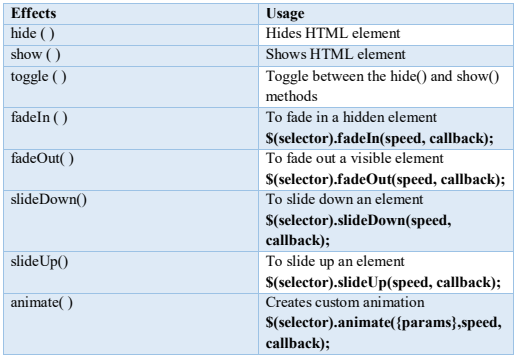
**$(this).hide( );**

**});**

**JQuery has different Event Methods some of them are listed in the following table:**



**JQuery Effects**



**CSS:**

JQuery can also set a specified CSS property using the syntax: **css("propertyname", "value");**   
For example, to change the style of a button:

$("button").css("border", "3px solid red")

You can also set multiple CSS properties using the syntax:

**css("{propertyname" : "value", "propertyname" : "value", …..} );**

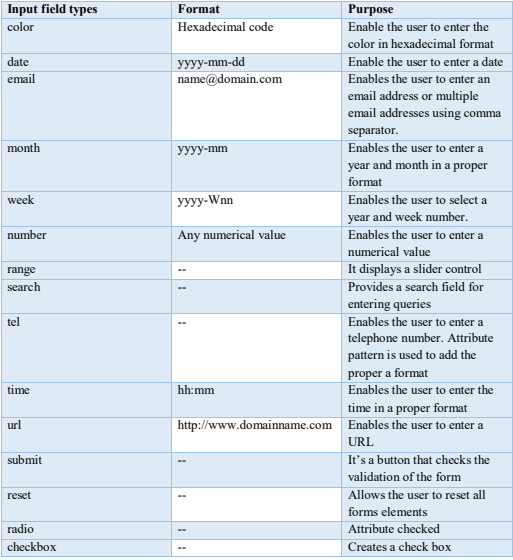
**For example:** **$("p").css ({"background-color": "black", "font-size": "200 %"});**

You can also add or remove entire premade CSS classes using **addClass(), removeClass(), toggleClass().**

**For example:**  **$("div").addClass("MenuBox");**

**HTML5 form validation**

It is difficult to validate user input such as email, URL, or time. However, in HTML5 the input types became self-validating on the client side.



**Input field attributes**

1) **Formnovalidate**

There is an attribute called formnovalidate it bypasses the validation for this form, and it is placed in the input type submit as an attribute.

2) **Placeholder**

It enables the user to display a temporary text in the input field.

3) **Autofocus**

It is an optional attribute that can be used to focus on a specific input element, and it can’t be used on multiple input fields.

4) **Required**

It forces the user to enter a value in the required field before submitting the form.

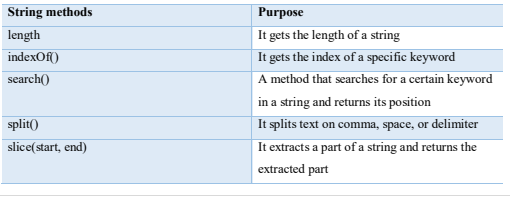
5) **Autocomplete**

It allows the user to automatically fill the user’s information based on previous input.

6) **Maxlength** It specifies the maximum allowed length for an input

7) **Pattern (regex)**

It is used to specify a pattern for the input field, for example, pattern = “[A-Za-z]{3}” The 3 here means that the user should enter 3 letters only. We use a method called match() to search a string for a match against a certain expression.



**Exercise (1) Website Graphics**

With the help of the HTML and CSS files provided in the Exercise1 folder, manipulate the DOM object to produce the following.

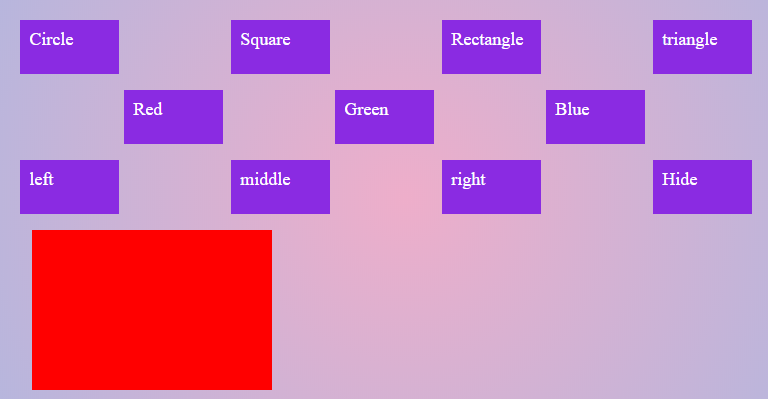


Figure Intial page

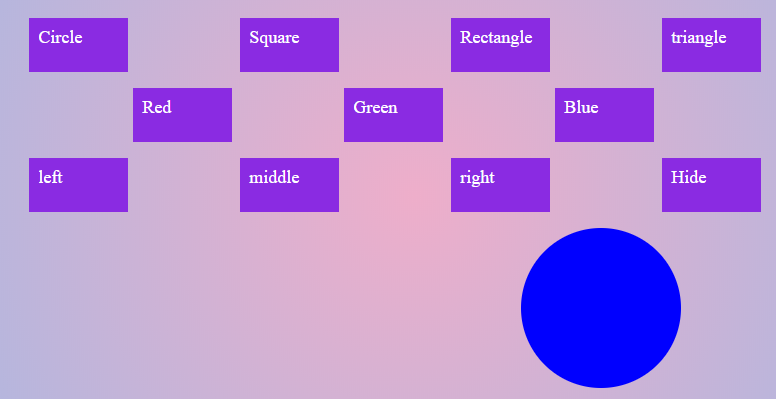
The initially red rectangle is given the ID “subject”, the other squares in violet are the commands that you need to implement, the first row consist of shapes that the subject needs to change into. The second consists of colors that it needs to change into. The third consists of minor animations to move the subject left and right (feel free to add any animation other than the given). 

Figure Clicked Circle, blue, and right

Hint: CSS classes for the shapes are premade (except triangle), so you will only need to add and remove class. For the rest you will change CSS and use animate. Hide is a little different though.

**Exercise (2) Cleaning Up**

Building on the previous exercise, can you slide make everything be kept tidily in a sliding bar that only opens when required? Please do it in a smooth animation.

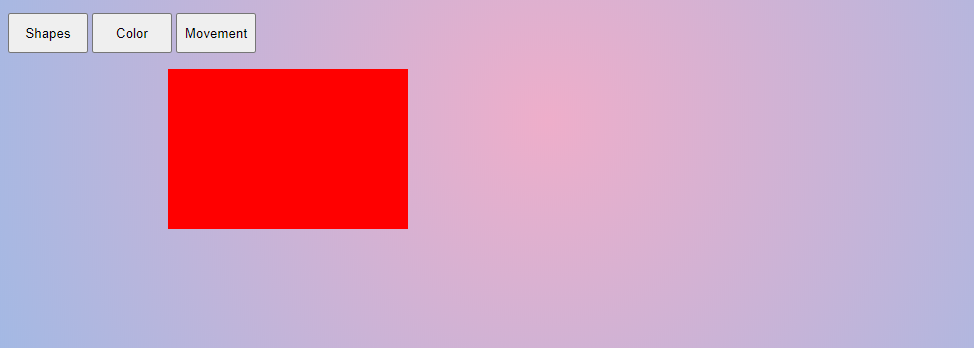


Figure New Initial Page

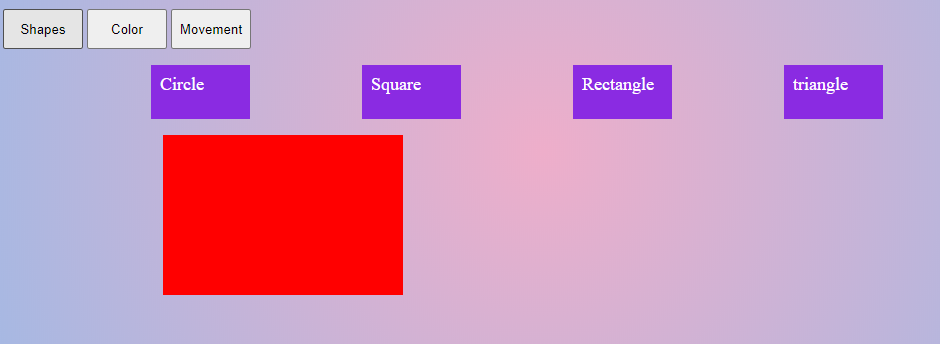


Figure Clicked Shapes

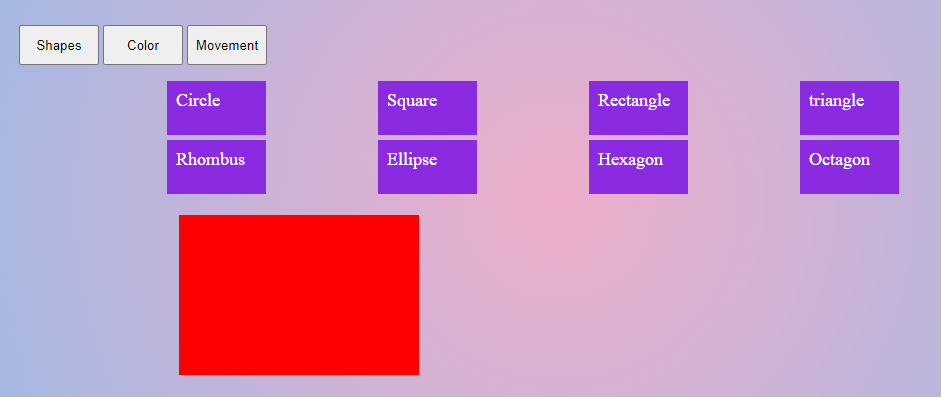
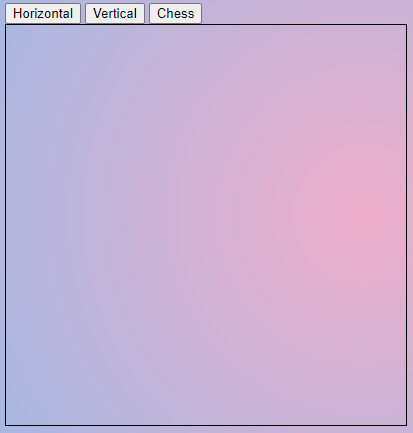
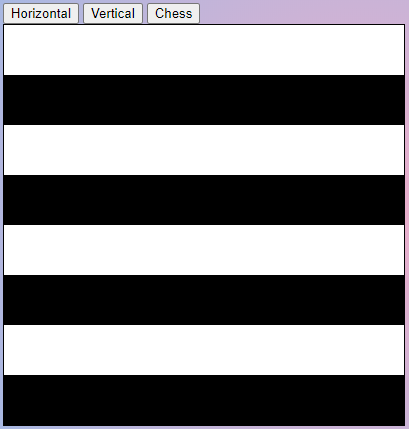
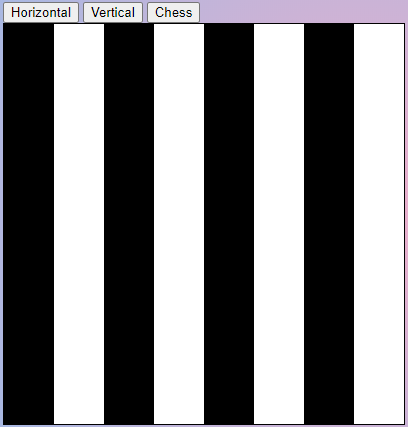
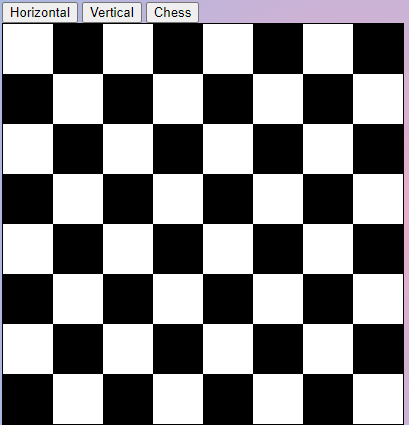


Figure if you want to, you can now add as many Commands as you want and still keep it compact, think how you can leverage that in your project

**Exercise (3) Evens and Odds**

The following four screenshots show index page from Exercise3 folder, the first screenshot is the initial clear page, the next 3 show the result of adding mouseover/mouseenter events to the 3 buttons on top, implement those events.

Hint1: You will change the background of every tr or td tag

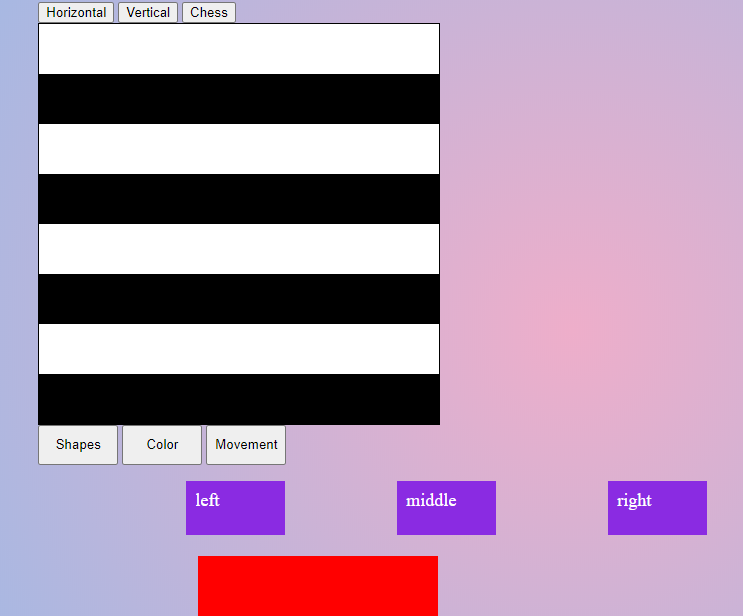
Hint2: Use :odd and :even selectors with either the tr and td tag, remember to remove the other’s background effects so they wouldn’t override each other.

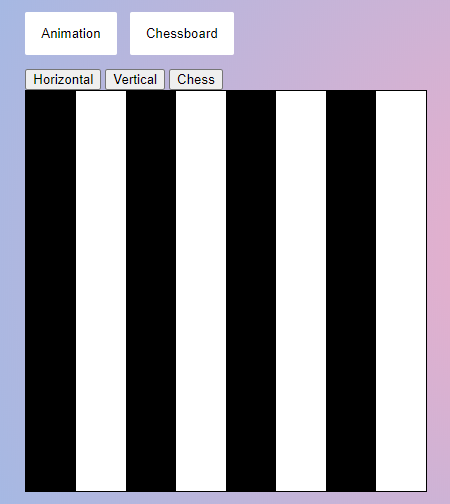
Hint3: Chess is more challenging, you will have to make some minor changes the original generated tags you are provided and change the selector :odd.

**Exercise (4) Tabs**

The last 2 exercises were combined as a fun repository of games in one place, but they are untidy. Put them into tabs in any way you please. **Hint** you will change the display attribute.

It is preferable if you copy and paste your answers from the previous exercises to test the functionality of the tabs with the existing games, but this step is optional.





**Exercise (5) Form Validation**

Write an html code for the following form and its validation using HTML one time, and then write a script which validates this form. The form will include the following input fields:

* Username
* Age
* Address
* Email
* Gender
* Password
* Confirm password

The username fields should accept only letters, while the age fields accepts only numbers. The username shouldn’t be less than 8 letters. The email field should contain “@” and finally the password field should be matching with the retyped one, it should also contain a special character, and be mix of numbers and letters. Make sure that ALL input fields are required. You can add any necessary validation on this form.

**Hint**: use onclick="validateForm()" or onsubmit="return validateForm();"

