8.4 jQuery Event Methods

- All the different visitors' actions that a web page can respond to are called **events**.
- An event represents the precise moment when something happens.
  1. Moving a mouse over an element
  2. Selecting a radio button
  3. Clicking on an element
- The term "fires/fired" is often used with events. Example: "The keypress event is fired, the moment you press a key".
- In jQuery, most DOM events have an equivalent jQuery method.
- To assign a click event to all paragraphs on a page, you can do this:

```
$("p").click();
```

- The next step is to define what should happen when the event fires. You must pass a function to the event:

```
$("p").click(function(){
   // action goes here!!
});
```

Events	Keyboard events	Triggered by	Sample code	Code explanation

click	The function is executed	Mouse	`$("p").click(function(){`
			`  $(this).hide();`
			`});`
				• When a \<p> element is clicked, it hides (removes) that specific \<p>

	when the user clicks on the HTML element.		element from the page.

`dblclick`	The function is executed when the user double-clicks on the HTML element.	`$("p").dblclick(function(){`
		`  $(this).hide();`
		`});`
			• When a `<p>` element is double-clicked, it hides (removes) that specific `<p>` element from the page.

`mouseenter`	The function is executed when the mouse pointer enters the HTML element.	`$("#p1").mouseenter(function(){`
		`  alert("You entered p1!");`
		`});`
			• When the mouse cursor enters the `id="p1"` element, it displays an alert with the message "You entered p1!"

`mouseleave`	The function is executed when the mouse pointer leaves the HTML element.	`$("#p1").mouseleave(function(){`
		`  alert("Bye! You now leave p1!");`
		`});`
			• When the mouse cursor leaves the `id="p1"` element, it displays an alert with the message "Bye! You now leave p1!"

`mouseup`
 	
The function is executed, when the
		`$("#p1").mousedown(function(){`
		`  alert("Mouse down over p1!");`
		`});`
			• When the mouse button is pressed down over `id="p1"`, it

	left, middle or right mouse button is released, while the mouse is over the HTML element.		triggers an alert with the message "Mouse down over p1!"
`mousedown`	The function is executed, when the left, middle or right mouse button is pressed down, while the mouse is over the HTML element.	```$("#p1").mouseup(function(){
alert("Mouse up over p1!");
});```	• When the mouse button is released over id="p1", it triggers an alert with the message "Mouse up over p1!"
`hover`	The first function is executed when the mouse enters the HTML element, and the second function is	```$("#p1").hover(function(){
alert("You entered p1!");
},
function(){
alert("Bye! You now leave p1!");
});```	• When the mouse button is released over `id="p1"`, it triggers an alert with the message "Mouse up over p1!"

	executed when the mouse leaves the HTML element.			

`focus`	The function is executed when the form field gets focus.	Form interaction	`$("input").focus(function(){`
			`  $(this).css("background-color", "#cccccc");`
			`});`
				• When an input element gains focus (is clicked or selected), it changes its background color to light gray (`#cccccc`) using CSS.

`blur`	The function is executed when the form field loses focus.		`$("input").blur(function(){`
			`  $(this).css("background-color", "#ffffff");`
			`});`
				• When an input element loses focus (e.g., when the user clicks outside of the input), it changes its background color to white (`#ffffff`) using CSS.

`submit`	This function is executed when the user is attempting to submit a form.		`<form id="register">` `  <label for="name">Name:</label>` `  <input type="text" id="name" name="name">` `   ` `  <label for="email">Email:</label>` `  <input type="email" id="email" name="email">` `   ` `  <input type="submit" value="Submit">` `</form>` `<div id="output"></div>`	• When the form is submitted, it retrieves the values entered for name and email, and then displays them in the "output" `<div>` as a concatenated string.

		```<script>``` ```$(document).ready(function() {``` ```$("#register").submit(function(event) {``` ```event.preventDefault();``` ```let name = $("#name").val();``` ```let email = $("#email").val();``` ```$("#output").text("Name: " + name + "\nEmail: " + email);``` ```});``` ```});``` ```</script>```	
```change```	This function is executed when an element's value changes.	```<label for="color_select">Select a color:</label>``` ```<select id="color_select">``` ```<option value="red">Red</option>``` ```<option value="blue">Blue</option>``` ```<option value="green">Green</option>``` ```<option value="yellow">Yellow</option>``` ```</select>``` ```<script>``` ```$(document).ready(function() {``` ```$("#color_select").change(function() {``` ```var selected_color = $(this).val();``` ```$("#output").text("Selected color: " + selected_color);``` ```});``` ```});```	• When a different color is selected from the dropdown, it captures the selected color's value and displays it in an element with ```id="output"``` as "Selected color: ```[selected_color]```."

			`</script>`	

`keydown`	This function is executed when a keyboard key is pressed down.	Keyboard	`<label for="kb_input">Press a key: </label>`
			`<input type="text" id="kb_input">`
			
			`<div id="output">`
			   `<p>Keydown: <span id="kd_value"></span></p>`
			   `<p>Keypress: <span id="kp_value"></span></p>`
			   `<p>Keyup: <span id="ku_value"></span></p>`
			`</div>`
				• When a key is pressed down, during the keypress, and when it's released, the respective event handler updates the content of three different elements in the `<div>` element with `id="output"` with the corresponding key code or character information.

`keypress`	This function is executed when a button is pressed down.		```
			<script>
			    $(document).ready(function() {
			
			$("#kb_input").keydown(function(event) {
			
			$("#kd_value").text(event.keyCode);
			        });
			
			$("#kb_input").keypress(function(event) {
			
			$("#kp_value").text(event.which);
			        });
			``	

`keyup`	The function is executed when a keyboard key is released.		```
			$("#kb_input").keyup(function(event) {
			
			$("#ku_value").text(event.keyCode);
			        });
			    });
			</script>
			``	

Additional Material

- **Learn more**
  - W3Schools