

Question-1: What is JQuery?

Answer. JQuery is a lightweight, "write less, do more", JavaScript library.

The purpose of JQuery is to make it much easier to use JavaScript on your website.

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.

JQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

Question-2: How to Apply CSS Using JQuery, How to Add Class and Remove Class in JQuery, JQuery Animation?

Answer.

=>How to Apply CSS Using JQuery

JQuery `css()` Method

The `css()` method sets or returns one or more style properties for the selected elements.

Return a CSS Property

To return the value of a specified CSS property, use the following syntax:`css("propertyname");`

=>How to Add Class and Remove Class in JQuery

`addClass()` - Adds one or more classes to the selected elements



Module_7 : JQuery Basic, Effects & Advanced

`removeClass()` - Removes one or more classes from the selected elements

`toggleClass()` - Toggles between adding/removing classes from the selected elements

`css()` - Sets or returns the style attribute

=>JQuery Animation

jQuery Animations - The `animate()` Method

The jQuery `animate()` method is used to create custom animations.

Syntax: `$(selector).animate({params},speed,callback);`

The required `params` parameter defines the CSS properties to be animated.

The optional `speed` parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional `callback` parameter is a function to be executed after the animation completes.

Question-3:How to create slider with animation?

Answer.Learn how to create a responsive slideshow with CSS and JavaScript.

Slideshow / Carousel

A slideshow is used to cycle through elements:

1 / 4Caption Text

Create A Slideshow

Step 1) Add HTML:

Example

```
<!-- Slideshow container -->
```

```
<div class="slideshow-container">
```

```
<!-- Full-width images with number and caption text -->
```

```
<div class="mySlides fade">
```

```
  <div class="numbertext">1 / 3</div>
```

```
  
```

```
  <div class="text">Caption Text</div>
```

```
</div>
```

```
<div class="mySlides fade">
```

```
  <div class="numbertext">2 / 3</div>
```

```
  
```

```
  <div class="text">Caption Two</div>
```

```
</div>
```

```
<div class="mySlides fade">
```

```
  <div class="numbertext">3 / 3</div>
```

```
  
```

```
  <div class="text">Caption Three</div>
```

```
</div>
```

Module_7 : JQuery Basic, Effects & Advanced

```
<!-- Next and previous buttons -->
<a class="prev" onclick="plusSlides(-1)">&#10094;</a>
<a class="next" onclick="plusSlides(1)">&#10095;</a>
</div>
<br>

<!-- The dots/circles -->
<div style="text-align:center">
  <span class="dot" onclick="currentSlide(1)"></span>
  <span class="dot" onclick="currentSlide(2)"></span>
  <span class="dot" onclick="currentSlide(3)"></span>
</div>
```

Step 2) Add CSS:

Style the next and previous buttons, the caption text and the dots:

Example

```
* {box-sizing:border-box}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
```

```
}
```

```
/* Hide the images by default */
```

```
.mySlides {  
  display: none;  
}
```

```
/* Next & previous buttons */
```

```
.prev, .next {  
  cursor: pointer;  
  position: absolute;  
  top: 50%;  
  width: auto;  
  margin-top: -22px;  
  padding: 16px;  
  color: white;  
  font-weight: bold;  
  font-size: 18px;  
  transition: 0.6s ease;  
  border-radius: 0 3px 3px 0;  
  user-select: none;  
}
```

```
/* Position the "next button" to the right */
```

```
.next {
```



Module_7 : JQuery Basic, Effects & Advanced

```
right: 0;
border-radius: 3px 0 0 3px;
}

/* On hover, add a black background color with a little bit see-through
*/

.prev:hover, .next:hover {
    background-color: rgba(0,0,0,0.8);
}

/* Caption text */
.text {
    color: #f2f2f2;
    font-size: 15px;
    padding: 8px 12px;
    position: absolute;
    bottom: 8px;
    width: 100%;
    text-align: center;
}

/* Number text (1/3 etc) */
.numbertext {
    color: #f2f2f2;
    font-size: 12px;
```



Module_7 : JQuery Basic, Effects & Advanced

```
padding: 8px 12px;
position: absolute;
top: 0;
}

/* The dots/bullets/indicators */
.dot {
    cursor: pointer;
    height: 15px;
    width: 15px;
    margin: 0 2px;
    background-color: #bbb;
    border-radius: 50%;
    display: inline-block;
    transition: background-color 0.6s ease;
}

.active, .dot:hover {
    background-color: #717171;
}

/* Fading animation */
.fade {
    animation-name: fade;
    animation-duration: 1.5s;
```

```
}
```

```
@keyframes fade {  
  from {opacity: .4}  
  to {opacity: 1}  
}
```

Step 3) Add JavaScript:

Example

```
let slideIndex = 1;  
showSlides(slideIndex);
```

```
// Next/previous controls  
function plusSlides(n) {  
  showSlides(slideIndex += n);  
}
```

```
// Thumbnail image controls  
function currentSlide(n) {  
  showSlides(slideIndex = n);  
}
```

```
function showSlides(n) {  
  let i;
```




Module_7 : JQuery Basic, Effects & Advanced

```
let slides = document.getElementsByClassName("mySlides");
let dots = document.getElementsByClassName("dot");
if (n > slides.length) {slideIndex = 1}
if (n < 1) {slideIndex = slides.length}
for (i = 0; i < slides.length; i++) {
  slides[i].style.display = "none";
}
for (i = 0; i < dots.length; i++) {
  dots[i].className = dots[i].className.replace(" active", "");
}
slides[slideIndex-1].style.display = "block";
dots[slideIndex-1].className += " active";
}
```

Automatic Slideshow

To display an automatic slideshow, use the following code:

Example

```
let slideIndex = 0;
showSlides();

function showSlides() {
  let i;
  let slides = document.getElementsByClassName("mySlides");
  for (i = 0; i < slides.length; i++) {
    slides[i].style.display = "none";
```

```
}  
slideIndex++;  
if (slideIndex > slides.length) {slideIndex = 1}  
slides[slideIndex-1].style.display = "block";  
setTimeout(showSlides, 2000); // Change image every 2 seconds  
}
```

Question-4: Event bubbling tickling example?

Answer:

Code:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta http-equiv="X-UA-Compatible"  
content="IE=edge">  
  <meta name="viewport" content="width=device-width,  
initial-scale=1.0">  
  <title>Document</title>  
  <script  
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7  
.0/jquery.min.js" integrity="sha512-  
3gJwYpMe3QewGELv8k/BX9vcqhryRdzRMxVfq6ngyWXwo03GFEzjsU  
m8Q7RZcHPHkstattq7/GFoxjCVUjkjvPdw=="
```

```
crossorigin="anonymous" referrerpolicy="no-
referrer"></script>
<style>

    .main *
    {
        border:3px solid gray;
        padding: 25px;
        display: block;
    }
</style>
</head>
<body>

    <div class="main">
        <div class="gp">
            grandParent
            <ul>UL
                <li>LI
                    <span>SPAN</span>
                </li>
            </ul>
        </div>
    </div>

    <script>
        $('span').parents().css("border","3px solid
red");
        //$('span').parent().css("border","3px solid
red");
        //$('span').parentsUntil('.main').css("border"
,"3px solid red");
        //$('.gp').children().css("border","3px solid
red");
```

Module_7 : JQuery Basic, Effects & Advanced

```
    //$('.gp').find('ul').css("background","lightcoral");  
    //$('.gp').find('li').css("background","lightblue");  
    //$('.gp').find('span').css("background","pink")  
  </script>  
</body>  
</html>
```

Output:

