**Chapter 7: How to work with images and timers**

**Murach's JavaScript and jQuery (4th Edition)**

**MULTIPLE CHOICE**

1. Why might you want to preload images on a page that uses rollovers?

| a. | To prevent delays displaying the images when the swapped images are large. |
| --- | --- |
| b. | To prevent delays displaying the images when the connection is slow. |
| c. | To prevent delays when the page is being loaded. |
| d. | All of the above. |
| e. | A and B only. |

2. Assuming that the href attribute of each <a> element contains the URL for an image, what does the code that follows do?

**const flowers = document.querySelectorAll("a");**

**for (let flower of flowers) {**

**const image = new Image();**

**image.src = flower.href;**

**}**

| a. | It stores all <a> elements in an array named flowers. |
| --- | --- |
| b. | It creates a new Image object for each image in the flowers array. |
| c. | It preloads each Image object. |
| d. | All of the above. |
| e. | A and B only. |

3. Given this HTML:

**<main>**

**<h1>Image Swap</h1>**

**<p>Click on an image to enlarge.</p>**

**<ul id="image\_list">**

**<li><a href="images/pic1.jpg" title="dogs">**

**<img src="thumbnails/t1.jpg" alt=""></a></li>**

**<li><a href="images/pic2.jpg" title="cats">**

**<img src="thumbnails/t2.jpg" alt=""></a></li>**

**</ul>**

**<h2 id="caption">Animals</h2>**

**<p><img id="main\_image" src="images/pic1.jpg" alt=""></p>**

**</main>**

Why are the img elements for the thumbnails coded within the <a> elements?

| a. | So the user can swap the main image by pressing Enter when a thumbnail has the focus. |
| --- | --- |
| b. | So the user can access the thumbnail links by pressing the Tab key. |
| c. | So the application is easy to use by the motor-impaired. |
| d. | All of the above. |

**Code Example 7-1**

JavaScript code for an Image Swap

**1 const $ = selector => document.querySelector(selector);**

**2**

**3 document.addEventListener("DOMContentLoaded", () => {**

**4 const caption = $("#caption");**

**5 const mainImage = $("#main\_image");**

**6 const imageLinks = $("#image\_list").querySelectorAll("a");**

**7**

**8 for (let link of imageLinks) {**

**9 const image = new Image();**

**10 image.src = link.href;**

**11**

**12 link.addEventListener("click", evt => {**

**13 mainImage.src = link.href;**

**14 mainImage.alt = link.title;**

**15 caption.textContent = link.title;**

**16 evt.preventDefault();**

**17 });**

**18 }**

**19 imageLinks[0].focus();**

**20 });**

4. (Refer to Code Example 7-1) Which line or lines of code attach an event handler for the click event of each link?

| a. | lines 3-6 | c. | lines 12 - 17 |
| --- | --- | --- | --- |
| b. | lines 8-10 | d. | line 19 |

5. (Refer to Code Example 7-1) On line 10, what does the link variable refer to?

| a. | the image that’s currently being processed |
| --- | --- |
| b. | the <a> element that’s currently being processed |
| c. | the <img> element that the user clicked on |
| d. | the <a> element that the user clicked on |

6. (Refer to Code Example 7-1) What would happen if line 16 was omitted?

| a. | The code would only work in older versions of IE. |
| --- | --- |
| b. | Clicking on a link would open a new browser window or tab and display the image specified by the href attribute of the link. |
| c. | Nothing would happen when the user clicks on one of the links. |
| d. | Clicking on a link would cause the caption, but not the larger image, associated with the link to be displayed. |

7. Which of the following statements runs the displayAlert() function once after 3 seconds?

| a. | **setTimeout(displayAlert, 3);** | c. | **setInterval(displayAlert, 3);** |
| --- | --- | --- | --- |
| b. | **setTimeout(displayAlert, 3000);** | d. | **setInterval(displayAlert, 3000);** |

8. Which of the following statements runs the displayAlert() function every 3 seconds?

| a. | **setTimeout(displayAlert, 3);** | c. | **setInterval(displayAlert, 3);** |
| --- | --- | --- | --- |
| b. | **setTimeout(displayAlert, 3000);** | d. | **setInterval(displayAlert, 3000);** |

9. When creating a timer, the delay or interval is specified in

| a. | seconds | c. | minutes |
| --- | --- | --- | --- |
| b. | milliseconds | d. | tenths of a second |

**Code Example 7-2**

**The JavaScript:**

**1 const $ = selector => document.querySelector(selector);**

**2**

**3 let timer = null;**

**4 let count = 10;**

**6 const updateCounter = () => {**

**7 $("#counter").firstChild.nodeValue = count;**

**8 if (count <= 0) {**

**9 clearInterval(timer);**

**10 $("#counter").firstChild.nodeValue = "Blastoff!";**

**11 }**

**12 count--;**

**13 };**

**15 document.addEventListener("DOMContentLoaded", () => {**

**16 updateCounter();**

**17 timer = setInterval(updateCounter, 1000);**

**18 });**

**The HTML:**

**<h3>Countdown: <span id="counter">Starting...</span></h2>**

10. (Refer to Code Example 7-2) Which of the following statements about this code is true?

| a. | It creates a timer that displays a counter on the page that replaces the text “Countdown”. |
| --- | --- |
| b. | It creates a timer that uses the setInterval() method to change the updateCounter value every 1000 milliseconds . |
| c. | It creates a timer that counts down from 10 to 1 and displays "Blastoff!" in the span element when the counter variable reaches 0. |
| d. | It will fail because a timer cannot count down. |

11. (Refer to Code Example 7-2) The timer in this code will update the counter

| a. | every 10 seconds |
| --- | --- |
| b. | once, after a delay of 1000 milliseconds |
| c. | immediately, when the page loads |
| d. | every second for 10 times |

12. (Refer to Code Example 7-2) Why does line 16 call the updateCounter() function?

| a. | To stop the timer |
| --- | --- |
| b. | To start the timer |
| c. | So the countdown will begin immediately when the page loads |
| d. | So the countdown start one second after the page loads |

13. (Refer to Code Example 7-2) Which line contains the statement that stops the timer?

| a. | 3 |
| --- | --- |
| b. | 9 |
| c. | 16 |
| d. | 17 |