



Microsoft

70-480 EXAM

Microsoft Programming in HTML5 with JavaScript and CSS3

Product: **Demo**

For More Information:

<https://www.dumpsplanet.com/70-480-dumps>

Question: 1

You are troubleshooting an application. Users report that the UI is slow to respond.

You need to improve UI responsiveness by moving application tasks to web workers. Which two tasks can you move to achieve this goal? (Each correct answer presents a complete solution. Choose two.)

- A. A function that loops through the Document Object Model to update the style of page elements
- B. A long-running calculation that requires looping through an array
- C. A function that performs graphic-intensive animation
- D. A function that stores large amounts of data to local storage

Answer: B, C

Note:

* Despite the improvements in JavaScript engines, it is not uncommon for users to encounter frozen user interfaces as the browser works through resource intensive tasks. This leads to a horrible user experience. The purpose of Web Workers is to give developers a way of instructing the browser to process large tasks in the background; therefore preventing the UI from freezing up.

* The [Web Workers](#) specification defines an API for spawning background scripts in your web application. Web Workers allow you to do things like fire up long-running scripts to handle computationally intensive tasks, but without blocking the UI or other scripts to handle user interactions

Question: 2

You develop an interactive scalable vector graphic (SVG) application.

You write the following code (Line numbers are included for reference only.):

```
01 <script>
02
03 </script>
04 <svg height="150" width="150" id="myGraphic">
05   <circle cx="25" r="20" cy="20" fill="orange" id="myCircle"/>
06 </svg>
07 <button id="zoom" onclick="zoomIn();">Zoom In</button>
```

You need to increase the size of the circle by 50 percent. Which code segment should you insert at line 02?

- A.

```
function zoomIn() {
    var myGraphic = document.getElementById("myGraphic");
    myGraphic.createSVGTransform(1.5);
}
```
- B.

```
function zoomIn() {
    var myGraphic = document.getElementById("myGraphic");
    myGraphic.setAttribute("currentScale", 1.5);
}
```
- C.

```
function zoomIn() {
    var myCircle = document.getElementById("myCircle");
    myCircle.setAttribute("currentScale", 1.5);
}
```
- D.

```
function zoomIn() {
    var myCircle = document.getElementById("myCircle");
    myCircle.r = myCircle.r * 1.5;
}
```

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Increase the radius (the r property) of the circle (not the graphic) by a factor 1.5. Incorrect:
CurrentScale is used for zooming.
Reference: The HTML <svg> Element

Question: 3

You develop an HTML5 webpage. You have the following HTML markup:

```
<label for="personname">Given and Surname:</label>
<input type="text" name="personname" /><br />

<label for="businessname">Business Name:</label>
<input type="text" name="businessname" /><br />

<label for="emailaddress">Email Address:</label>
<input type="text" name="emailaddress" />
```

You need to change the background color for all of the elements whose name attribute ends with the word name.

Which code segment should you add to the webpage?

- A. `$('input [name!="name"]') .css ({ 'background-color' : ' #E0ECF8' });`
- B. `$('#input [name=~"name"] ') .css ({ 'background-color' : ' #E0ECF8' });`
- C. `$('#input[name="*name"]') .css ({ 'background-color' : ' #E0ECF8' });`
- D. `$('input [name="$name"] ') .css ({ 'background-color' : ' #E0ECF8' });`

Answer: C

The string pattern `"*name"` matches all strings that ends with name.

Question: 4

You develop an HTML application that is located at `www.adventure-works.com`. The application must load JSON data from `www.fabrikam.com`. You need to choose an approach for loading the data. What should you do?

- A. Design a REST URI scheme with multiple domains.
- B. Configure Cross-Origin Resource Sharing (CORS) on the servers.
- C. Load the data by using WebSockets.
- D. Use the jQuery `getJSON` method.

Answer: B

* Cross-origin resource sharing (CORS) is a mechanism that allows Javascript on a [web page](#) to make [XMLHttpRequests](#) to another [domain](#), not the domain the Javascript originated from. Such "cross- domain" requests would otherwise be forbidden by [web browsers](#), per the [same origin security policy](#). CORS defines a way in which the browser and the server can interact to determine whether or not to allow the cross-origin request.^[2] It is more powerful than only allowing same-origin requests, but it is more secure than simply allowing all such cross-origin requests.

* You must use Cross Origin Resource Sharing

It's not as complicated as it sounds...simply set your request headers appropriately...in Python it would look like:

```
self.response.headers.add_header('Access-Control-Allow-Origin', '*');
self.response.headers.add_header('Access-Control-Allow-Methods', 'GET, POST,
```

<https://www.dumpsplanet.com/>

```
OPTIONS'); self.response.headers.add_header('Access-Control-Allow-Headers', 'X-
Requested-With'); self.response.headers.add_header('Access-Control-Max-Age',
'86400');
```

Question: 5

You are building a web page for a newspaper publisher. You have the following requirements:

The web page should split the content into columns that are at least 100 pixels wide. The number of columns displayed must be set by the browser.

You need to implement the correct CSS properties.

Which line of code should you use?

- A. `<div id="outer" style="width: 100px; column-fill: balance;">...</div>`
- B. `<div id="outer" style="width: 100px; column-gap: 10px;">...</div>`
- C. `<div id="outer" style="column-width: 100px; "> . . </div>`
- D. `<div id="outer" style="width: 100px; columns: 100px 3">...</div>`

Answer: C

column-width

The width attribute specifies the width of a `<col>` element.

Normally, a `<col>` element takes up the space it needs to display the content. The width attribute is used to set a predefined width of a `<col>` element.

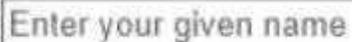
Syntax:

```
<col width="pixels|%|relative_length">
```

Question: 6

You develop a webpage with a standard input control by using HTML5.

The input control must display the text Enter your given name, as shown below:



When a user selects the input control, the text must disappear.

You need to create the input control.

Which input control should you use?

- A. `<input name="GivenName" value=" Enter your given name" />`
- B. `<input name ="GivenName" default=" Enter your given name" />`
- C. `<input name="GivenName" text=" Enter your given name" />`
- D. `<input name="GivenName" placeholder=" Enter your given name" />`

Answer: D

The placeholder attribute specifies a short hint that describes the expected value of an input field (e.g. a sample value or a short description of the expected format).

The short hint is displayed in the input field before the user enters a value.

The placeholder attribute works with the following input types: text, search, url, tel, email, and password.
Reference: HTML <input> placeholder Attribute

Question: 7

You develop an HTML5 webpage. You have the following HTML markup:

```
<section id="news">
  <h1>World News</h1>
  <article id="news1">...</article>
  <article id="news2">...</article>
  <article id="news3">...</article>
</section>
<section id="sports">
  <h1>Sports</h1>
  <article id="sports1">...</article>
  <article id="sports2">...</article>
  <article id="sports3">...</article>
</section>
```

You need to add a background color to the first article in each section. Which code segment should you add to the webpage?

- A. \$ ("section article:first-child").css("background-color", "#f2f2f2");
- B. \$ ("section:first-child").css ("background-color", "#f2f2f2");
- C. \$ ("article:first-of-type") .css("background-color", "#f2f2f2");
- D. \$ ("section:first-of-type").css("background-color", "#f2f2f2");

Answer: C

The :first-of-type selector matches every element that is the first child, of a particular type, of its parent. Reference: CSS3 :first-of-type Selector

Question: 8

You are creating a custom object as described by the following code.

```
function square(side) {
  this.side = side;
  this.area = calcArea;
}
```

You need to implement the calcArea method.
Which code should you use?

- A.

```
function calcArea () {  
    return me.side * me.side;  
}
```
- B.

```
function calcArea (obj) {  
    return obj.side * obj.side;  
}
```
- C.

```
function calcArea () {  
    return side * side;  
}
```
- D.

```
function calcArea () {  
    return this.side * this.side;  
}
```

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Need to refer to .this in the calculation.

Question: 9

DRAG DROP

You are developing a web page that includes the following HTML.

```
<input name="ITSupport" type="checkbox" value="IT" id="choc"  
    onclick="enable();" /><label for="IT">IT Support</label>  
<input type="email" placeholder="IT@Contoso.com" id="eMail"  
    name="eMail" disabled /><label for="eMail" >Your eMail </label>
```

You need to ensure that the email element is enabled only if the user selects the IT Support check box. Which CSS selectors should you use? (To answer, drag the appropriate selector to the correct location. Each selector may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.)

Answer:

```


input: {
  background-color: #e0e0e0;
}

<script type="text/javascript">
  function enable(e) {
    if (document.getElementById) {
      if (document.getElementById('IT').checked) {
        document.getElementById('eMail').disabled = false;
      } else {
        document.getElementById('eMail').disabled = true;
      }
    }
  }
</script>

```

The disabled property sets or returns whether an email field should be disabled, or not. A disabled element is unusable and un-clickable. Disabled elements are usually rendered in gray by default in browsers.

Example

Disable an email field:

`document.getElementById("myEmail").disabled = true;` Reference: Input Email disabled Property

Question: 10

You are developing an HTML5 web form to collect feedback information from site visitors. The web form must display an INPUT element that meets the following requirements: Allow numeric values between 1 and 10. Default to the value of 5.

Display as a slider control on the page.

You need to add the INPUT element to the form. Which HTML element should you add?

- Rating (Between 1 and 10): `<input type="number" name="rating" min="1" max="10">`
- Rating (Between 1 and 10): `<input type="number" name="rating" min="1" max="10" default="5">`
- Rating (Between 1 and 10): `<input type="range" name="rating" min="1" max="10" default="5">`
- Rating (Between 1 and 10): `<input type="range" name="rating" min="10" max="10" value="5">`

Answer: C

`input type="range"`

The `<input type="range">` is used for input fields that should contain a value within a range. Depending on browser support, the input field can be displayed as a slider control.

Example

<form>

<input type="range" name="points" min="0" max="10">

</form>

Question: 11

You develop an HTML5 application. The application uses an image that is returned from an HTTP POST request.

You develop the following code

: (Line numbers are included for reference only.)

```
01 var xhr = new XMLHttpRequest();
02 xhr.open('POST', '/Images/heroAccent.png', true);
03 var img = document.createElement('img');
04
05 xhr.send();
```

You need to display the loaded image in the application. Which code segment should you insert at line 04?

- A.

```
xhr.responseType = 'blob';
xhr.onload = function (e) {
  if (this.status == 200) {
    img.src = window.URL.createObjectURL(this.responseText);
    document.body.appendChild(img);
  }
};
```
- B.

```
xhr.responseType = 'blob';
xhr.onload = function (e) {
  if (this.status == 200) {
    img.src = window.URL.createObjectURL(this.response);
    document.body.appendChild(img);
  }
};
```
- C.

```
xhr.responseType = 'document';
xhr.onload = function (e) {
  if (this.status == 200) {
    img.src = window.URL.createObjectURL(this.responseText);
    document.body.appendChild(img);
  }
};
```
- D.

```
xhr.responseType = 'document';
xhr.onload = function (e) {
  if (this.status == 200) {
    img.src = window.URL.createObjectURL(this.response);
    document.body.appendChild(img);
  }
};
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: B

As an image will be returned the response type must be blob, not document. Similarly, this.response must be used not this.responseText.

Reference: createObjectURL method

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual **70-480** Exam Questions With Answers.

Dumps Planet also provides you extra 20% discount on the purchase of **70-480** exam. Use Coupon code like “dp20%”

<https://www.dumpsplanet.com/70-480-dumps>

Money Back Guarantee



<https://www.dumpsplanet.com/>