# Programming in HTML5 with JavaScript and CSS3

## Chapter 07: Working with forms



Q1. What is the purpose of a HTML Form?

1. A form captures, validates user input, processes the input and shows the result of processing.
2. A form allows users to signing in and access personalized information.
3. A form is responsible for collecting data and sending that data is sent back to the server for processing
4. All of the above Answer: C

Q2. Which of the following is or true for a web server?

1. Web servers originally were responsible for only receiving and handling requests from the browsers through HTTP.
2. A Web server hosts web pages and resources of a web application
3. By born, communication between a web server and a browser is stateless
4. With the growth of technology, now web servers also handle requests for pages that contain code that will execute at the server and respond with the results of code execution

Answer: A, B, C, D

Q3. Although HTTP is stateless by born, Web servers now hold state (data) between webpage requests, so the developer can connect many pages to form a web application.

1. True
2. False Answer: A

Q4. What is the role of a web browser?

1. The web browser provides a platform-independent means of displaying webpages that were written with HTML.
2. The web browser responsible for only receiving and handling requests from the browsers through HTTP.
3. The web browser sends user commands to the web servers
4. All of the above Answer: A

Q5. enable the web browsers to talk to the web servers without clearing the existing webpages from the browser window.

1. Silverlight
2. CSS
3. XHTML
4. AJAX Answer: D

Q6. Which of the following is or are true about HTTP?

1. A web browser and a web server communicate using HTTP protocol.
2. HTTP provides a standard for Web browsers and servers to communicate.
3. HTTP protocol is a text-based protocol, so text commands are sent to the server.
4. None of the above Answer: A, B, C

Q7. Which category of status codes indicate server error?

1. 1xx
2. 2xx
3. 3xx
4. 4xx
5. 5xx Answer: D

Q8. Which category of status codes indicate client error?

1. 2xx
2. 3xxx
3. 4xx
4. 5xx Answer: D

Q9. Which http verb is used to retrieve information is identified by the Request-UI?

1. GET
2. POST
3. PUT
4. HEAD Answer: A

Q9. Which http verb is used to Request for the server to accept the data being sent from the client to modify existing server data.

1. GET
2. POST
3. PUT
4. HEAD Answer: B

Q11. All form elements can send data when the Submit button is clicked.

1. True
2. False

Answer: B [Page 316, Not all elements can send data when the Submit button is clicked. For example reset button data is never sent]

Q12. How can to locate all selected <option> elements by using the jQuery selector?

1. $('option:selected')
2. $( ‘option’).is(‘selected’)
3. $(‘option’).has(‘:selected’)
4. $(‘option selected’)

Answer: A

Q12. How can to locate all selected checkboxes by using the jQuery selector?

1. $(‘input:checked’)
2. $('input[type=checkbox]:checked')
3. $(‘checkbox:checked’)
4. $(‘input[type=checkbox] checked’)

Answer: B

Q13. Which of the following will not trigger the submission of a form?

1. <button type='submit' name='submitButton' >Submit</button>
2. <input type='submit' name='submitButton' value='Submit' />
3. <input type='image' src='/images/submit.gif' alt='Submit' />
4. <input type=’button’ name=’submitButton’ value=’Submit’ />

Answer: D

Q14. Which method do you use to decode a URI-encoded querystring?

1. decode
2. decodeURI
3. decodeURIComponent
4. deseialize Answer: C

Q15. How do we pre-populate a form input field with some text?

1. You can't
2. By giving the input tag a value attribute with whatever text you want pre-populated.
3. By typing it in.
4. By giving a placeholder attribute Answer:B

Q16. How do you create a drop-down box on a form?

1. With option tags.
2. With CSS
3. With the select tag.
4. Don’t know

Answer: C

Q17. A form has two required attributes. They are:

1. Method and Action
2. Get and Post
3. Start and Stop
4. Begin and End Answer: A

Q19. The <input> tag has several kinds of controls which are dictated by the attribute.

1. src
2. value
3. name
4. type Answer: D

Q20. You can mask the input typed into a text field by specifying an <input> tag as .

1. password
2. secure
3. invisible
4. hidden Answer: A

Q21. If you wanted to display a list of two hundred countries which control should you use:

1. radio button
2. checkbox
3. select list
4. slide widget Answer: C

Q22. Which of the following is true?

1. The <option> tag does not have a value attribute. If selected, it passes its displayed value.
2. The <option> tag only passes values through its value attributes. The display value is never passed.
3. The <option> tag passes values through the value tag if it is present, and passes the value displayed if the value attribute is not given.
4. The purpose of the <option> tag is to specify the type of control to use, not to pass values.

Answer: C

Q23. The developer can choose to allow more than one option to be selected by choosing the

attribute.

1. multiple
2. many
3. numerous
4. HTML does not provide an attribute that allows the developer to enable more than one option to be selectable on a list

Answer: A

Q24. In order to clear the values in a form, the developer specifies an <input> tag with:

1. the type attribute set to empty
2. the type attribute set to clear
3. the type attribute set to refresh
4. the type attribute set to reset Answer: D

Q25. Defines the caption to the fieldset tag.

1. <legend>
2. <h1>
3. <caption>
4. <header> Answer: C

Q26. A label's for attribute should match the input's attribute.

1. Typ
2. Id
3. Name
4. Class Answer: B

Q27. The action attribute tells the form?

1. How to send the form’s data
2. Where to send the form’s data
3. When to send the form’s data
4. What data in forms to send Answer: B

Q28. The viewer never sees the transfer of data when the form uses <form method="get">.

1. True
2. False Answer: B

Q29. Checkbox buttons let users select one or more of a particular set of choices.

1. True
2. False Answer: A

# Programming in HTML5 with JavaScript and CSS3

## Chapter 08: Websites and services



Q1. What is node.js?

1. Node.js is a JavaScript framework that fills gaps in HTML5 features in browsers
2. Node.js is a JavaScript framework that allows to manipulate HTML DOM in browser- independent way
3. Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications
4. Node.js is a platform for easily building application based on google app Answer: C

Q2. Which platform Node.js is built on?

1. Microsoft JScript engine
2. Edge Trident engine
3. Google Chrome V8 engine
4. ECMAScript engine Answer: C

Q3. How do you import a module in code with node.js?

1. Using require function
2. Using import function
3. Using load function
4. Using open function Answer: A

Q4. Using Node.js, you can write JavaScript that executes on the back-end server.

1. True
2. False Answer: A

Q5. Which of the following is not true for the Node.js platform?

1. The Node.js platform is used for easily building fast, scalable network applications.
2. The Node.js platform implements an event-driven, non-blocking I/O model and is lightweight and efficient.
3. The Node.js platform is perfect for data-intensive, real-time applications that run across distributed devices.
4. Node.js can be plugged to any modern browsers so that it can take advantage of the platform

Answer: D

[Every modern browser has JavaScript runtime built-in. Node.js can replace or do any to those. Node.js is back-end JavaScript platform]

Q6. Which one do you use to install node modules?

1. Node executable
2. npm
3. nuget
4. bower Answer: B

[Node package manager]

Q7. Which Node command do you use to load a module?

1. load
2. exports
3. use
4. require Answer: D

Q8. Which low-level built-in module do you use to create a http server object?

1. express
2. server
3. http
4. socket.io Answer: C

[Your use expressjs to create a server that is built around http and express is not built-in]

Q9. How do create basic HTTP server?

1. using the http module’s create() method
2. using the http module’s createServer() method
3. using the http module’s server() method
4. using the http module’s listen() method

Answer: B

Q10. Which method of http module do you to create a server object?

1. create
2. server
3. createServer
4. request Answer: C

Q11. What is the correct signature of the createServer method of http module?

1. The createServer method accepts a function that has request object parameter
2. The createServer method accepts a function that has response object parameter
3. The createServer method accepts a function that has request object and response object parameters
4. The createServer method accepts a function that has server object parameter Answer: C

Q12. What does the createServer method of http module return?

1. A request object
2. A response object
3. A server object
4. A stream that exposes methods to send response to client Answer: D

Q13. Which method of the server object starts the server and makes it listen for incoming request at a specific port?

1. start
2. run
3. listen
4. execute Answer: C

Q14. Which method of the response object do you use to set http header explicitly?

1. header
2. set
3. write
4. writeHead Answer: D

[You can use setHeader method to set header implicitly. Example

responseObject.setHeader (‘Content-Type’, ‘text/html’)

Q15. Which one sets http header to response stream correctly?

1. response.writeHead({

'Content-Type': 'application/json', 'X-Powered-By': 'bacon'

});

1. response.writeHead(200, { 'Content-Type': 'application/json', 'X-Powered-By': 'bacon'

});

1. response.writeHead(200, {

{ 'Content-Type': 'application/json'},

{'X-Powered-By': 'bacon'}

});

1. response.writeHead(200, [

{'Content-Type': 'application/json'},

{'X-Powered-By': 'bacon'}

}]);

Answer: B

[Correct writeHeader (statuscode, headers) statuscode- 3-digit number

headers -passed as json like {header1: header string,header2:header string... }]

Q16. You are creating a server object using the http module. To test you have written the following code.

Line 01: const http = require (‘http’);

Line 02: var server =http.createServer((r,s)=>{ Line 03:

Line 04:

Line 05: //more code

Line 06: });

Line 07: server.listen (8080);

Which code should you use to send test response to the browser from the server at line 03 and 04?

1. r.writeHead(200, {'Content-Type': 'text/plain'}); r.end('Hello World from Node.js!\n');
2. s.writeHead(200, {'Content-Type': 'text/plain'}); s.end('Hello World from Node.js!\n');
3. server.writeHead(200, {'Content-Type': 'text/plain'}); server.end('Hello World from Node.js!\n');
4. http.writeHead(200, {'Content-Type': 'text/plain'}); http.end('Hello World from Node.js!\n');

Answer: B

[createServer accepts a function

The first parameter of the function wraps request object, the second parameter wraps response object.

To send response you must use the response object methods.

\*\*

Here arrow function notation is used. Every browser now accepts these notation. function f (){statement} is equivalent to ()= statement;

function f (x,y){} is equivalent to (x,y)=>{} For more see lecture notes]

Q17. What happens when you call the listen method on server object by passing a port number?

1. The IP address is set to 0.0.0.0 and the server start listening to the port specified for incoming request
2. Answer random is set to the server and the server start listening to the port specified for incoming request to the set IP
3. The IP address is set to localhost and the server start listening to the port specified for incoming request to the localhost
4. The listen method will produce error if a valid IP is not passed along with the port number Answer: C

[server object.listen (port) results to the server address http://localhost:port The IP for local machine is 127.0.0.1

http://127.0.0.1:port also works]

Q18. Which method of url module should you use parse query QueryString from a url?

1. resolve
2. format
3. toString
4. parse Answer: D

Q19. Which on correctly calls the parse method in order to extract QueryString from a request object?

1. url.parse (requestObject.url)
2. url.parse (requestObject.url,true)
3. url.parse (requestObject.url,false)
4. url.parse (requestObject, false) Answer: B

[The first parameter is url, the second one boolean to indicate whether the query string is to be parsed. If true query string will be parsed. Default is false.]

Q20. Which node module helps you to parse QueryString from a URL?

1. http
2. url
3. formidable
4. utilities Answer: B

Q21. You want to extract QueryString values from a URL. You are using url module and you want QueryString values are passed as an object. You have the code (line numbers are only reference): 1: var url = require('url');

2: var adr = 'http://localhost:8080/default.htm?year=2017&month=february'; 3:

4: console.log(q.year);

Which code should you write at line 2?

1. var q = url.parse(adr);
2. var q = url.parse(adr, true);
3. var q = url.parse(adr, false);
4. var q = url.parse(adr.query); Answer: B

Q22. You have created a Node.js module in Math.js file and the module exported a function add that accepts two numbers returns sum of the two numbers passed as arguments. How can you use the module?

1. var math = require (‘./Math.js’);

var result = math.add(5, 4);

1. var math = require (‘Math.js’);

var result = math.add(5, 4);

1. var math = require (‘Math’);

var result = math.add(5, 4);

1. var math = load (‘./Math.js’);

var result = math.add(5, 4);

Answer: A

[Your own local module must have relative address in require function]

Q23. What is the role of a Node.js package?

1. A Node.js package is a set of code wrapped inside in function and exported so that it can be called from other code.
2. A Node.js package is a collection of modules with a manifest that describes the package and its dependencies and can be publicly and privately published for you and others to use.
3. A Node.js package is a web application created using ExpressJS framework.
4. None of the above Answer: B

Q24. A is a shareable file that promotes reuse.

1. Application
2. Module
3. Package manifest
4. npm command Answer: B

Q25. You are creating a Node.js module using the code (line numbers are for reference only) Line 1: function Add(a, b) {

Line 2: return a+b; Line 3: }

Line 4:

What should be included at line 3?

1. export ( add )
2. exports.add = add;
3. Function.prototype.add = add;
4. Exports = add;

Answer: B [To expose things we use exports or module.exports. For example,

var x = 10; exports.x = x;

exports.y = function () { return ‘Hello’;} same effect will be if written below module.exports.x = x;

module.exports.y = function () { return ‘Hello’;}

]

Q26. Which tool do you use to install node modules?

1. node
2. gulp
3. npm
4. nuget Answer: C

Q27. You are creating a node module. The code is like below: (Line numbers are only for illustrative purpose)

var fs = require(‘fs’);

function getContent(filename, cb){

//code

}

Which code should you add at line 5?

1. exports = getContent;
2. exports.getContent = getContent;
3. exports ( getContent );
4. module.exports (getContent ) Answer: B

Q28. Which file contains metadata that describes your node.js package or application?

1. readme.md
2. package.json
3. package.menifest
4. app.js

Answer: C

Q29. To create package.json for your application, which command do you use?

1. node start
2. npm start
3. node init
4. npm init Answer: D

Q30. Which one is the manifest file in Node.js package?

1. package.json
2. README.md
3. lib\main.js.
4. bin\app.js Answer: A

Q31. What is the purpose of the manifest file in Node.js package?

1. It loads the required dependency modules
2. It describes the package and its dependencies
3. It contains help to get the user started
4. All of the above Answer: B

Q32. What can you publish by using npm?

1. A JavaScript file
2. A module
3. A JavaScript class
4. A package Answer: D

Q33. Which command of the Node Package Manager creates the package manifest file package.json?

1. npm start
2. npm create
3. npm init
4. npm –g install Answer: C

Q34. How do you install a package named contoso for local use?

1. npm install contoso
2. npm -i contoso
3. npm -g contoso
4. npm contoso Answer: A

Q35. You are creating a web server using Node.js with Expressjs. You want to serve static files in public folder under root. You have written code like below (line numbers are for reference only) Line 1: var express = require('express');

Line 2: var app = express();

Line 3:

Line 4: var port = 8080;

Line 3: app.listen(port);

What should you write at line 3?

1. app.use( dirname + '/public');
2. express.static( dirname + '/public');
3. app.use(express.static( dirname + '/public'));
4. app.use(express.get('/public/\*')); Answer: C

Q36. What does the Node.js global object dirname returns?

1. The name of the Node installation directory
2. The name of the directory that the currently executing script resides in
3. The name of the root directory of the web application
4. The name of the user local directory where user packages are installed Answer: B

Q37. Using , you can write JavaScript that makes asynchronous calls the server and processes the result.

1. HTML5
2. AJAX
3. Browser Plug-in
4. Browser Extension Answer: B

Q38. To send requests to a web service from your webpage, you use .

1. JavaScript
2. AJAX
3. ExpressJS
4. Socket.io Answer: B

Q39. REST stands for .

1. Request State Transfer
2. Response State Transfer
3. Real State Transfer
4. Representational State Transfer Answer: D

Q40. attempts to use the standard operations of HTTP (or similar protocols) by mapping create, retrieve, update, and delete (CRUD) operations to HTTP methods.

1. AJAX
2. Node JS
3. jQuery
4. REST Answer: D

Q41. Which one provides the ability to send data to the server and receive data back from the server without requiring a repaint of the browser screen

1. Websocket
2. Node.js
3. Web services
4. ASP.Net Answer: C

[Ref: Page 364]

Q42. Which is one is an implementation of REST Web service?

1. Socket.io
2. Node.js
3. Web API
4. WCF Answer: C [Ref: Page 364]

Q43. Which one is an example of Web service that exposes an arbitrary set of operations?

1. Socket.io
2. Node.js
3. Web API
4. WCF Answer: C [Ref: Page 364

There are two major classes of web services. The first is called Representational State Transfer (REST), in which the primary purpose of the service is to manipulate web resources by using a uniform set of stateless operations. The second is arbitrary web services, in which the service might expose an arbitrary set of operations.

Web API is REST based service

WCF is can be used for services that exposes operations as you need]

Q44. Which one is true for a REST service?

1. REST uses the standard operations of HTTP by mapping create, retrieve, update, and delete (CRUD) operations to HTTP methods
2. REST uses an arbitrary set of operations for create, retrieve, update, and delete (CRUD) operations
3. REST uses SOAP request for create, retrieve, update, and delete (CRUD) operations
4. All of the above Answer: A

[Ref.: Page 365]

Q45. Which HTTP method is used in REST to retrieve data or perform an operation that does not change server-side data?

1. GET
2. POST
3. PUT
4. DELETE Answer: A [Ref.: Page 365]

Q46. Which HTTP method is used in REST to update data or retrieve data when complex parameter values need to be sent to the server?

1. GET
2. POST
3. PUT
4. DELETE Answer: B

[Ref.: Page 365]

Q47. Which HTTP method is used in REST to insert new data?

1. GET
2. POST
3. PUT
4. DELETE Answer: C [Ref.: Page 365]

Q48. Which one retrieves the customer number five from a REST service?

1. HTTP Method: GET, URL: http://localhost:8080/Customer
2. HTTP Method: GET, URL: http://localhost:8080/Customer/5
3. HTTP Method: PUT, URL: http://localhost:8080/Customer/5
4. HTTP Method: POST, URL: http://localhost:8080/Customer/5 Answer: B

Q49. Which one retrieve all orders from a REST service?

1. HTTP Method: GET, URL: http://localhost:8080/Orders
2. HTTP Method: GET, URL: http://localhost:8080/Orders/5
3. HTTP Method: PUT, URL: http://localhost:8080/Orders
4. HTTP Method: POST, URL: http://localhost:8080/Orders Answer: A

Q50. Which middlewire method on expressjs app object do you use to mount a folder as the root of the website?

1. use
2. static
3. get
4. Post Answer: A [Ref.: Page 367]

Q51. Which middlewire method on expressjs do you use tostatic files?

1. express.use
2. express.static
3. express.get
4. express.post Answer: A

[Ref.: Page 367]

Q52. The primary object that makes an AJAX call is ?

1. $.ajax
2. XML
3. XMLHttpRequest.
4. Http Answer: C

Q53. What readyState code of the XMLHttpRequest object indicates all operations are finished upon an AJAX call to the server?

1. 1
2. 2
3. 3
4. 4 Answer: D [Ref: Page 370

0 Uninitialized The open method has not been called yet. 1 Loading The send method has not been called yet.

2 Loaded The send method has been called; headers and status are available. 3 Interactive Downloading; the response properties hold the partial data.

4 Completed All operations are finished.]

Q54. Which event of the XMLHttpRequest object you must subscribe to handle the asynchronous call, you must subscribe to?

1. the onreadystateschange event
2. the onprogress event
3. the onsuccess event
4. the onabort event Answer: A

Q55. Which shorthand jQuery method do you use to load data from the server by using an HTTP GET request?

1. $.ajax();
2. $.get()
3. $.getScript()
4. $.load() Answer: B

Q56. Which shorthand jQuery method do you use to load SON-encoded data from the server by using an HTTP GET request?

1. $.get();
2. $.getJSON()
3. $.getScript()
4. $.load() Answer: B

Q57. When you use jQuery to execute an AJAX call, is returned.

1. an XMLHttpRequestobject
2. a promise object
3. a JSON object
4. a SOAP response Answer: B

[Ref. Page 378]

Q58. Which method of the promise object returned by a jQuery AJAX call to add handlers to be called when the AJAX call has completed, regardless of whether it was successful?

1. always()
2. done()
3. progress()
4. success()

Answer: A [Ref. Page 378

always(): Add handlers to be called when the AJAX call has completed, regardless of whether it was successful

done(): Add handlers to be called when the AJAX call is successful fail(): Add handlers to be called when the AJAX call has failed

progress(): Add handlers to be called when the AJAX call generates progress notifications]

Q59. CORS stands for .

1. cross-origin result sharing
2. common-origin result sharing
3. common-origin resource sharing
4. cross-origin resource sharing Answer: D

Q60. What is CORS?

1. A browser specification to prevent xss attack
2. A browser specification that defines ways for a web server to allow its resources to be accessed by a webpage from the same domain
3. A browser specification that defines ways for a web server to allow its resources to be accessed by a webpage from a different domain
4. A browser specification that defines ways for search engine optimization Answer: C

Q61. You have a web service that provides read-only access to data. You want to expose this web service so it can be called from any other website. What header would you return?

1. Allow: \*
2. Access-Control: \*
3. Allow-Origin: \*
4. Access-Control-Allow-Origin: \* Answer: D

# Programming in HTML5 with JavaScript and CSS3

## Chapter 09: Asynchronous operations



Q1. What is the advantage of executing long-running code asynchronously?

1. Asynchronous operations use CPU-cores efficiently so that long-running code can finish quickly
2. Asynchronous operations execute long-running code with higher priority
3. Asynchronous operations create another execution path so that the long-running code can execute on one execution path while the user interface is responsive on another execution path
4. All of the above Answer: C

[Ref. Page 393]

Q2. is executed on a thread that is different from the main thread, a thread being a separate execution path.

Which one correctly fits in the blank space?

1. An asynchronous operation
2. A synchronous operation
3. Every jQuery method
4. Every JavaScript method Answer: A

Q3. jQuery animations are performed asynchronously.

1. True
2. False Answer: A [Ref.: Page 393]

Q4. What is a promise in JavaScript?

1. An alternative mechanism to execute a task asynchronously.
2. A mechanism to schedule work to be done on a value that might not yet be computed.
3. A mechanism to load large JavaScript file asynchronously using a background thread.
4. A mechanism to create full-duplex persistent connection in old browsers.

Answer: B

[Ref.: Page 394]

Q5. Which one is not a valid state of a promise object?

1. pending
2. resolved
3. rejected
4. faulted Answer: D [Ref.: Page 394]

Q6. What are the possible states of a promise object can be?

1. Pending
2. Resolved
3. Rejected
4. Done Answer: A, B, C [Ref.: Page 394]

Q7. What is the state of a promise object when it starts?

1. pending
2. resolved
3. rejected
4. unknown Answer: A

[Ref.: 394]

Q8. What is the benefit of using the promise object?

1. It enables you to execute browser-independent AJAX calls to REST services
2. It enables you to create persistent connection with a remote server easily
3. It enables you to write non-blocking logic that executes asynchronously without having to write a lot of synchronization and plumbing code.
4. All of the above Answer: C

[Ref.: Page 394]

Q9. Which one correctly defines a promise object?

1. A value which may be available now, or in the future, or never.
2. An operation that is not available now for calling but may be available in future
3. A placeholder of object instance which will be created in future
4. A value which can never be computed Answer: A

[Ref.: Page 394]

Q10. Which one is the correct syntax to create a promise object?

1. new Promise (function (resolve, reject) {/\*...\*/});
2. new Promise (resolve, reject) {/\*...\*/};
3. new Promise (function () {/\*...\*/});
4. new Promise ([resolve, reject]) {/\*...\*/}; Answer: A

Q11. Consider the code block below function asyncOperation (){

var promise = new Promise(function(resolve, reject){ var result = callALongOperation();

if(somethingBadHappened) reject(‘Bad happened’)

resolve(result);

});

return promise;

};

Which one correctly calls ayncOperation?

1. var retVal = asyncOperation(); console.log(retVal);
2. try {

var retVal = asyncOperation(); console.log(retVal);

}

catch(e){}

1. asyncOperation()

.then(frunction (retVal){ console.log(retVal);

})

.catch(function(err){});

1. None of the above Answer: C

[Promise is called like below Promise.then(function(..){}).catch(function(..){..}) A, B called synchronously. Will log undefined]

Q12. Which method chains asynchronous operations?

1. pipe()
2. done()
3. resolve()
4. always() Answer: A [Ref.: Page 399]

Q13. Which method creates a resolved promise object?

1. resolve()
2. done()
3. $.when()
4. $.Deferred().

Answer: C [Ref.: Page 402]

Q14. Which method do you call to subscribe to the successful completion of an asynchronous operation?

1. done
2. always
3. complete
4. success Answer: A [Ref.: Page 398]

Q15. Which method do you call on the deferred object to indicate a change in progress?

1. progress
2. notify
3. done
4. resolve Answer: B

[Ref.: Page 403]

Q16. Which one defines a specification for using JavaScript everywhere?

1. jQuery
2. Node.js
3. CommonJS
4. ECMA JavaScript Answer: C

[Ref.: Page 394]

Q17. Which method of a promise object should you use to execute your code upon successful completion?

1. done
2. when
3. always
4. resolve Answer: A [Ref.: Page 397]

Q18. Which method of a promise object should you use to execute your code when the asynchronous call has completed, regardless of success or failure?

1. done
2. when
3. always
4. resolve Answer: C [Ref.: Page 397]

Q19. You want to chain several asynchronous calls together. Which promise method will you use?

1. done
2. pipe
3. always
4. resolve Answer: B [Ref.: 399]

Q20. Which method of the deferred object do you use, when you want to update the progress?

1. resolve
2. reject
3. notify
4. progress Answer: C

[Ref.: Page 400]

Q21. Which method of promise object do you use to execute code when the notify method on deferred object is executed?

1. when
2. done
3. always
4. progress Answer: D

[Ref.: Page 400]

Q22. Which one do you use to create a promise object in jQuery?

1. $.Deferred()
2. $.Promise()
3. $.Ajax()
4. $.Async()

Answer: A [Ref.: Page 395]

Q23. Which one is not a method of jQuery deferred object?

1. resolve ()
2. reject ()
3. notify ()
4. then () Answer: D [Ref.: Page 396]

Q24. With jQuery, the promise object’s method chains asynchronous operations.

1. progress
2. then
3. pipe
4. always Answer: C [Ref.: Page 400]

Q25. Which method do you call on the promise object to indicate a change in progress?

1. progress
2. notify
3. done
4. resolve Answer: A

Q26. Which one is used to execute a script file asynchronously?

1. promise
2. web worker
3. node.js
4. websocket Answer: B

[Ref.: Page 404]

Q27. In the web worker code, you want the asynchronous code to update the DOM. Which method can you use?

1. updateDom().
2. workerUpdate().
3. dom().
4. You cannot update the DOM from within the asynchronous code.

Answer: D [Ref.: Page 406

if you need to access the DOM, the web worker must post a message back to the creator, and the creator must process the message and access the DOM as needed]

Q28. Which object can be used when the web worker and the creator need to reference the same object?

1. Mutex.
2. Semaphore.
3. Closure.
4. The web worker and the creator cannot reference the same object.

Answer: D

[Ref.: Page 404, 414

Web workers lack synchronization locks]

Q29. Web workers supports synchronization locks, critical sections, semaphores, or mutexes.

1. True
2. False Answer: B [Ref.: Page 404

Web workers lack synchronization locks]

Q30. Which method do you use for communicating to and from the web worker?

1. message
2. postMessage
3. onmessage
4. send Answer: B

[Ref.: Page 405]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 10: WebSocket communications



Q1. Which one is an example of two-way communications?

1. A Web service application
2. A single page web application
3. A chat room application
4. A REST services application Answer: C

[Ref.: Page 415]

Q2. Which one of the following provides bidirectional communication support to your web applications?

1. Web API
2. WCF Service
3. WebSocket API
4. jQuery AJAX Answer: C

Q3. Which one is true for a two-way communications pattern between a server and a client in a web application?

1. Only the browser initiates connection and sends messages to the server
2. Only the server initiates connection and sends messages to the server
3. The browser initiates connection and sends messages to the server and also the server can send messages to the browser
4. None of the above Answer: C

[Ref.: Page 415]

Q4. Which of the following is a web technology that provides full-duplex communications over a single TCP connection?

1. RPC
2. XML Web Service
3. DCOM
4. WebSocket protocol Answer: D

[Ref.: Page 416]

Q5. Which one of the following provides a standardized way for the server to send content to the browser and to allow messages to be passed back and forth while keeping the connection open?

1. RPC technology
2. AJAX
3. WebSocket protocol
4. SOAP protocol

[Ref.: Page 416]

Q6. The WebSocket protocol is

1. an independent, TCP-based protocol
2. an independent, UDP-based protocol
3. a HTTP-based protocol
4. dependent on MSMQ protocol Answer: A

[Ref.: Page 416]

Q7. What is the relation of WebSocket protocol with HTTP?

1. Both are independent of TCP-based protocol
2. Both are dependent of TCP-based protocol
3. WebSocket handshake is interpreted by HTTP servers as a request to switch to WebSocket protocol
4. All of the above Answer: C

[Ref.: Page 416]

Q8. The enables a persistent connection between the client (browser) and the server?

1. WebSocket
2. Web worker
3. Geolocation
4. HTTP protocol Answer: A

[Ref.: Page 416]

Q9. Which is not a method of a WebSocket object?

1. WebSocket constructor
2. open
3. close
4. send Answer: B

[Ref.: Page 416]

Q10. Which one of following communication channels the WebSocket is designed for?

1. Multiplexing
2. Simplex
3. Full-duplex
4. Uni-directional Answer: C

[Ref.: Page 415]

Q11. Which of the following is a valid WebSocket instantiation?

1. wsConnection = new WebSocke[t(‘h](http://studygroup.70480.com/)ttp[://studygroup.70480.com](http://studygroup.70480.com/)’);
2. wsConnection = new WebSocket(‘tcp://studygroup.70480.com’,[‘soap’,’xmpp’]);
3. wsConnection = new WebSocket(‘wss://studygroup.70480.com’,[‘soap’,’xmpp’]);
4. wsConnection = new WebSocket(‘ftp://studygroup.70480.com’,[‘soap’,’xmpp’]);

[Ref.: Page 416

A method that requires a URL argument and can optionally accept additional parameters to define the sub-protocol that you’ll use, such as chat or rpc. All of the are valid construction

new WebSocket(‘ws://studygroup.70480.com’) new WebSocket(‘wss://studygroup.70480.com’);

new WebSocket(‘ws://studygroup.70480.com’,[‘soap’,’xmpp’]); new WebSocket(‘wss://studygroup.70480.com’,[‘soap’,’xmpp’]);]

Q12. Which of the following statements properly handles the reception of data from a WebSocket?

1. wsConnection.onpost = function(msg){..};
2. wsConneciton.onreceive = function(msg){...};
3. wsConnection.onmessage = function(msg){...};
4. wsConnection.ongetdata = function(msg){...};

Answer: C [Ref. Page 417]

Q13. Which is true in about the WebSkocket object?

1. The *WebSocket* object, which connects to the server when the connect method is invoked
2. The *WebSocket* object, which connects to the server when first message is sent to server using the send method
3. The WebSocket object, which connects to the server when its constructor is invoked
4. The *WebSocket* object, which connects to the server when you set the URL after its constructor is invoked

Answer: C [Page 417]

Q14. Which of the following is not an event of WebSocket object?

1. onopen
2. onclose
3. onmessage
4. onready Answer: D

[There are four events

onclose: An event property that’s called when the socket is closed. onerror: An event property that’s called when there is an error. onmessage: An event property that’s called when a message is received.

onopen: An event property that’s called when WebSocket establishes a connection.]

Q15. Which of the following are the methods of Websocket object?

1. open
2. close
3. send
4. ready Answer: B, C [Ref.: Page 416

There is no open method on the WebSocket object. When you instantiate the WebSocket object, it automatically attempts to open asynchronously. There is no ready method]

Q16. When working with the WebSocket object, which event can be used to retrieve the data that was received from the server?

1. onopen
2. onclose
3. onmessage
4. onerror Answer: C

[Ref.: Page 417, 419]

Q17. Which property of the WebSocket object indicates the binary data format the onmessage event receives?

1. type
2. format
3. messageType
4. binaryType Answer: D

[Ref.: Page 417]

Q18. The WebSocket URL begins with .

1. http:// or https://
2. ws:// or wss://
3. tcp:// or http://
4. http:// or ftp:// Answer: B

[Ref.: Page 417

wss:// for secure WebSocket protocol]

Q19. How do you open an WebSocket connection?

1. Create a WebSocket object and call the open method when you want to open a WebSocket connection
2. Create a WebSocket object and call the connect method when you want to open a WebSocket connection
3. There is no method to open WebSocket connection. When we instantiate the WebSocket object, it automatically attempts to open asynchronously
4. None of the above Answer: C

[Ref.: Page 419]

Q20. Which property of the WebSocket object do you check to obtain the state of the connection?

1. state
2. ready
3. readyState
4. status Answer: C [Ref.: Page 417]

Q21. Which value of readyState attribute of the WebSocket object indicates that the connection has been closed or could not be opened?

1. 0
2. 1
3. 2
4. 3 Answer: D [Ref.: Page 419]

Q22. You want to ensure that the WebSocket connection is not disconnected as a result of inactivity. How can you accomplish this?

1. Add code to the onclose event to reopen the connection when it’s closed.
2. Add code to send an empty message periodically before the connection is closed.
3. Set the keepAlive property on the WebSocket object to true.
4. Create a new WebSocket object each time you send a message Answer: B

[Ref.: Page 420]

Q23. Which of the following is or are websocket libraries?

1. Socket.io
2. SignalR
3. Express
4. AngularJS Answer: A, B [Ref.: Page 423]

Q24. Which library would you use if you are creating a Node.js website and want to write browser- independent code that uses WebSocket?

1. SignalR
2. Socket.IO
3. FarmSockets
4. AgnosticSocket Answer: B

[Ref.: Page 424]

Q25. How to transmit WebSockets data?

1. push(data)
2. socketConnection.push(data)
3. socketConnection.send(data)
4. send(data) Answer: C

Q26. Which of the following activities usually WebSockets perform?

1. Use a persistent connection
2. Permit server-side push
3. Do not permit client to send request to server
4. More overhead Answer: A, B

Q27. Which one do you use to broadcast the message to all clients?

1. socket object
2. socket.broadcast object
3. io.sockets object
4. io object Answer: C [Ref.: Page 427

socket.emit – for the current client

socket.broadcast.emit – for all clients except the current client io.sockets.emit – for all clients]

Q28. Which one do you use to broadcast the message to all clients except the current client?

1. socket object
2. socket.broadcast object
3. io.sockets object
4. io object Answer: B [Ref.: Page 427

socket.emit – for the current client

socket.broadcast.emit – for all clients except the current client io.sockets.emit – for all clients]

Q29. You are working with socket.io. You the code like below (Line numbers are illustrative purpose only)

1. io = require('socket.io').listen(server);
2. io.sockets.on('connection', function (socket) { 3.

4.

5. });

You want that whenever a client is connected an event will triggered on the client. Which code block will you insert at line 3?

* 1. io.sockets.emit(‘connectionSuccess’,’You are connected.’);
  2. socket.emit(‘connectionSuccess’,’You are connected.’)
  3. socket.broadcast.emit(‘connectionSuccess’,’You are connected.’)
  4. socket.on((‘connectionSuccess’,’You are connected.’)

Answer: B

[Ref.: Page 428]

Q30. You are working with socket.io. You the code like below (line numbers are only for reference)

1. io = require('socket.io').listen(server);
2. io.sockets.on('connection', function (socket) { 3.

4.

5. });

You want that whenever a client is connected an event will triggered on all the clients except the current client. Which code block will you insert at line 3?

* 1. io.sockets.emit(‘newConnected’,’You are connected.’);
  2. socket.emit(‘newConnected’,’You are connected.’)
  3. socket.broadcast.emit(‘newConnected’,’You are connected.’)
  4. socket.on((‘newConnected’,’You are connected.’)

Answer: C

[Ref.: Page 428]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 11: HTML5 supports multimedia



Q1. Why the developer must provide formats that are available across most browsers when incorporating media element such audio or video in web pages?

1. So that browsers can decide which plugin or extension they should use to play the media
2. So that browsers allow user to choose which format they want to load
3. So that browsers can decide which format they want to support
4. All of the above Answer: C

[Ref. Page 437]

Q2. What one of the following is not HTML5 supported video format?

1. Ogg/Theora (.ogv extension)
2. WebM/VP8 (.webm extension)
3. MPEG-4/H.264 (.mp4 extension)
4. Flash Video (.flv extension)

Answer: D [Ref. Page 438]

Q3. Which one is the most popular video format based on performance when comparing data- stream size to picture quality?

1. Ogg/Theora
2. WebM/VP8
3. MPEG-4/H.264
4. video/x-ms-wmv Answer: C

[Ref. Page 438]

Q4. Why should you should include more than one <source> element in <video> element?

1. To allow users to play one video after another
2. To provide many sources so that the browser can choose the most appropriate video codec.
3. To provide user a playlist from where users can choose which video to play
4. It is not possible to add more than one <source> element Answer: B

[Ref. Page 440]

Q5. Which one does not correctly embeds video?

1. <video width="320" height="240" controls="controls">

<source src="movie.mp4" />

You need a browser that supports HTML5!

</video>

1. <video src="movie.mp4" width="320" height="240" controls="controls"> You need a browser that supports HTML5!

</video>

1. <video width="320" height="240" controls="controls">

<source url="movie.mp4" />

You need a browser that supports HTML5!

</video>

1. <video width="320" height="240" controls="controls">

<source src="movie.mp4" />

</video>

Answer: C [Ref. Page 438

At a minimum, you need to set the src attribute to the URL of the video.

Syntax: <video src=”source path” width=”..” height=”..” ></video>

You should include more than one <source>

<video …….>

<source src=”source path” ….>

<source src=”source path” ….>

<source src=”source path” ….>

Fallback text

</video>]

Q6. How do the browsers choose the format of the audio/video when you provide multiple sources of various formats?

1. The browser checks all the sources and selects one which has the minimum size
2. The browser checks all the sources and selects one which takes the minimum time to download
3. The browser checks all the sources and selects one which provides the best performance
4. The browser starts looking at the top and stops when it finds a file that it can display Answer: D

[Ref. Page 439]

Q7. When providing multiple source of video, what is the recommended order?

1. Start with .mp4, then .ogv and then .webm
2. Start with .webm, then .ogv and then .mp4
3. Start with .ogv, then .webm and then .mp4
4. Start with .ogv, then .mp4 and then .webm Answer: B

[Ref. Page 439]

Q8. Which attribute of the video element control how the video should be loaded when the page loads?

1. controls
2. loop
3. preload
4. autoplay Answer: C

[Ref. Page 441]

Q9. How do you specify the video in a web page starts loading when the page loads?

1. By setting the preload attribute to true
2. By setting the preload attribute to auto
3. By setting the preload attribute to metadata
4. By setting the preload attribute to none Answer: B

[Ref. Page 441

The preload attribute specifies how the video should be loaded when the page loads. It can be set to auto, metadata, or none. The auto setting starts loading the video when the page loads. The metadata setting loads only the metadata, and the none setting doesn’t load anything.]

Q10. Which attribute should you add to <video> element so that the video will repeat when it has reached the end of its stream?

1. controls
2. autoplay
3. loop
4. peload Answer: C [Ref. Page 441]

Q11. Which attribute of <video> element specifies that the URL of an image is to be shown when the video is not playing?

1. image
2. img
3. src
4. Poster Answer: D [Ref. Page 441]

Q12. Which one is the developed and recommended by W3C for subtitling video?

1. SRT
2. WebVTT
3. VTT
4. XML Answer: B [Ref. Page 441

New standard by W3C for subtitling is WebVTT (Web Video Text Tracks) Popular one is SubRip Text (SRT) ( not a statndard from W3C]

A less-used standard called Timed Text Markup Language (TTML) is an XML-based format W3C recommends WebVTT and TTML ]

Q133. Which of the following is or are HTML5 standards for providing captions in video?

1. SRT
2. WebTT
3. TTML
4. XHTML Answer: B, C [Ref. Page 441]

Q14. You want to add caption to a video which will start from at 10th second and end at 20th second. Which one is the valid content of your WebVTT file?

1. WebTT

For more information visit: [www.idb-bisew.org](http://www.idb-bisew.org/) 00:00:10.000 --> 00:00:20.000

1. WebTT

00:00:10.000 --> 00:00:20.000

For more information visit: [www.idb-bisew.org](http://www.idb-bisew.org/)

1. WebTT

For more information visit: [www.idb-bisew.org](http://www.idb-bisew.org/) 00:00:10.000 --> 00:00:20.000

1. WebTT

From: 00:00:10.000 To: 00:00:20.000

For more information visit: [www.idb-bisew.org](http://www.idb-bisew.org/) Answer: B

[Ref. Page 442]

Q15. You want to include a video on your webpage that has an .avi file extension. If you supply the codec information, what will be displayed in browsers that support the <video> element but don’t have the codec?

1. The browser will show the <video> element, but will not play the video.
2. The browser will automatically download the codec and play the video.
3. The browser will display the contents of the <video> element.
4. The browser will automatically download a version of the browser that supports the codec.

Answer: A [Ref. Page 443]

Q16. You don’t want to supply an .ogg video because you find that the .ogg format is choppy and low quality. What other video formats can you include to be compatible with most browsers? (Choose all that apply.)

1. .webm
2. .avi
3. .mpg
4. .mp4
5. .mjpg Answer: A, D [Ref. Page 443]

Q17. Which one repeats the audio continuously when it completes?

1. <audio controls>

<source src="media.mp3" />

</audio>

1. <audio loop>

<source src="media.mp3" />

</audio>

1. <audio autoplay>

<source src="media.mp3" />

</audio>

1. <audio continuous>

<source src="media.mp3" />

</audio>

Answer: B

[Ref. Page 445]

Q18. Which object do the <audio> and <video> elements inherit from?

1. HTMLControl
2. HTMLElement
3. HTMLMediaElement
4. Media Answer: C [Ref. Page 447]

Q19. Which one of the following is not a method of the HTMLMediaElement object?

1. load ()
2. play ()
3. pause ()
4. stop () Answer: D [Ref. Page 447

Methods available

addTextTrack() Adds a new text track to the audio or video

canPlayType() Determines whether the browser can play the specified audio or video type load() Reloads the audio or video

play() Plays the audio or video

pause() Pauses the currently playing audio or video]

Q20. Which property of the HTMLMediaElement gets the URL of the current audio or video?

1. src
2. currentSrc
3. played
4. videoTracks Answer: B

[Ref. Page 448]

Q21. Which property of the HTMLMediaElement can be used to determine whether the media is playing?

1. played
2. muted
3. paused
4. readyState Answer: C

[Ref. Page 451]

Q22. You want to trigger some code when the media is loaded. Which event should you subscribe to?

1. onstalled
2. onseeked
3. onplay
4. onloadeddata Answer: D

[Ref. Page 449]

Q23. The W3C standardized using which format for video?

1. .webm
2. .mpg
3. .mp4
4. No standard has been selected.

Answer: D [Ref. Page 438]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 12: Drawing with HTML5



Q1. Which element of the following brings ability to draw in HTML5 without requiring any plug-in?

1. <figure>
2. <canvas>
3. <svg>
4. <object> Answer: B [Page 459

canvas is drawing. To draw you have use javascript

svg is for creating and displaying vector graphics. JavaScript is not required to work with svg]

Q2. Which element of the following to create and display Scalable Vector Graphics (SVG) in HTML5 without requiring any plug-in?

1. <figure>
2. <canvas>
3. <svg>
4. <object> Answer: C [Page 459

canvas is for drawing. To draw you have to use JavaScript

svg is for creating and displaying vector graphics. JavaScript is not required to work with svg]

Q3. What is SVG?

1. It is a bitmap web image format
2. It is a image format which supports multiple layers.
3. It is a new language to show animations in web pages
4. SVG is a language by which to define two-dimensional graphics in XML Answer: D

[SVG is a language by which to define two-dimensional graphics in XML, and the XML can be rendered by the browser by using the <svg> element]

Q4. Which one allows defining define two-dimensional graphics in XML?

1. canvas
2. svg
3. flash
4. sliverlight Answer: B

[Page 459]

Q5. How do you create drawings on canvas element in HTML5?

1. Using drawing definitions in XML inside canvas element
2. Using javascript through the canvas context
3. Using plug-in such as Adobe flash
4. None of the above Answer: B

[Page 459]

Q6. Which one is not the member of the <canvas> element?

1. width
2. height
3. toDataUrl()
4. fill() Answer: B [Page 460

fill() is the method of the context object]

Q7. What is the toDataUrl() method of the canvas element used for?

1. To serialize the canvas drawings so that it can be sent to server for save as a web image format
2. To create a URL that can be used with an element that requires an image URL
3. to send the canvas content to a different web page
4. to encode canvas data into url-encoded form Answer: B

[Page 460]

Q8. You want to use the drawings on the canvas in an img element. Which method should you use?

1. getContext()
2. getDataUrl()
3. toDataUrl()
4. toImageUrl()

Answer: C [Page 460]

Q9. Which one of the following is to draw on the canvas?

1. Methods of the canvas
2. Methods of the document object
3. Methods of the window object
4. Methods of the context object that is returned from the getContext method of the canvas Answer: D

[Page 460]

Q10. What does the getContext method of the canvas which accepts a parameter 2d return?

1. Image
2. Bitmap
3. CanvasRenderingContext
4. CanvasRenderingContext2D Answer: D

[Page 463]

Q11. Which method of the context object of the canvas is used to create a new, blank ImageData object?

1. createImageData()
2. getImageData()
3. putImageData()
4. drawImage()

Answer: A [Page 461]

Q12. Which method of the context object of the canvas moves the path to the specified point in the canvas without creating a line?

1. lineTo
2. moveTo
3. beginPath
4. stroke Answer: B [Page 462]

Q13. Which method of the context object of the canvas adds a new point and creates a line from that point to the last specified point in the canvas?

1. lineTo
2. moveTo
3. beginPath
4. stroke Answer: A [Page 461]

Q14. Which method of the context object of the canvas starts a path or resets the current path?

1. restore
2. moveTo
3. beginPath
4. closePath Answer: C

[Page 461]

Q15. Which method of the context object of the canvas creates a path from the current point back to the starting point?

1. lineTo
2. moveTo
3. beginPath
4. closePath Answer: D

[Page 461]

Q16. Which does the getContext method of the canvas accepts the value as parameter to return a context object that is supported by all browsers including Internet Explorer?

1. 2
2. 2d
3. 3d
4. webgl Answer: B

[Page 462]

Q17. What is the proper parameter to pass to the getContext method on the canvas to create two- dimensional drawings?

1. Webgl
2. 2d
3. 3d
4. No parameter Answer: B

[Page 462]

Q18. Which method of the graphics context of a canvas element should you use to draw an unfilled rectangular area?

1. rect()
2. fillRect()
3. strokeRect()
4. stroke() Answer: C [Page 462]

Q19. Which method of the rendering context object of a canvas element do you use to clear the specified rectangular area?

1. clearRect(x, y, w, h)
2. clear(x, y, w, h)
3. strokeRect(x, y, w, h)
4. rect(x, y, w, h)

Answer: A [Page 463]

Q20. You have a canvas element in your page, the HTML you write

<canvas id=’mycanvas’>

</canvas>

How can you get a reference to the canvas context?

1. var canvas = document.getElementById('mycanvas'); var ctx = canvas.getContext();
2. var canvas = document.getElementById('mycanvas'); var ctx = canvas.getContext2D();
3. var canvas = document.getElementById('mycanvas'); var ctx = canvas.getRenderingContext()
4. var canvas = document.getElementById('mycanvas'); var ctx = canvas.getContext('2d');

Answer: D [Page 463]

Q21. You are drawing on HTML5 canvas you have the code like below: (Line numbers are illustrative purpose only)

1. var canvas = document.getElementById('canvas');
2. var ctx = canvas.getContext('2d');

To draw a 100X100, which code segment after line 2 will do the purpose?

* 1. ctx.fillStyle = 'red'; ctx.rect(0, 0, 100, 100); ctx.fill();
  2. ctx.fillStyle = 'red'; ctx.fillRect(100, 100, 100, 100);
  3. ctx.fillStyle = 'red'; ctx.rect(0, 0, 100, 100); ctx.stroke();
  4. ctx.fillStyle = 'red'; ctx.strokeRect(100, 100, 100, 100);

Answer: B [Page 463]

Q22. How can you control the opacity when filling shapes?

1. By setting the appropriate value to the opacity property of the context object
2. By setting the appropriate value to the alpha property of the context object
3. By creating gradient color using createLinearGradient or createRadialGradient method and then setting the fillStyle property of the context object to the created gradient color
4. By setting a rgba color to the fillStyle property of the context object Answer: D

[Page 465]

Q23. Which one draws a rectangle which a blue filled with opacity .25?

1. var canvas = document.getElementById('myCanvas'); var ctx = canvas.getContext('2d');

ctx.fillStyle = "rgb(0, 0, 255)";

ctx.shadowBlur = .25;

ctx.fillRect(0, 0, 100, 100);

1. var canvas = document.getElementById('myCanvas'); var ctx = canvas.getContext('2d');

ctx.fillStyle = "rgb(0, 0, 255)";

ctx.shadowColor = .25;

ctx.fillRect(0, 0, 100, 100);

1. var canvas = document.getElementById('myCanvas'); var ctx = canvas.getContext('2d');

ctx.fillStyle = "rgb(0, 0, 255)";

ctx.fillRect(0, 0, 100, 100);

ctx. stroke(. 25);

1. var canvas = document.getElementById('myCanvas'); var ctx = canvas.getContext('2d');

ctx.fillStyle = "rgba(0, 0, 255, 0.25)";

ctx.fillRect(0, 0, 100, 100);

Answer: D [Page 465]

Q24. You want to create a gradient color using the createLinearGradient(x1, y1, x2, y2) method of the context object.

To create a horizontal gradient, how should you change the parameter values?

1. Change the values of x1, x2 and set the values of y1, y2 equal
2. Change the values of y1, y2 and set the values of x1, x2 equal
3. Set the values of x1, y1 and set different values to x2, y2
4. Set the same to all four parameters Answer: A

[Page 490

Change x values and keep y values same to create horizontal gradient Change y values and keep y values same to create vertical gradient Change all to create diagonal gradient]

Q25. Which code example can be used to draw a rectangle on a <canvas> element that has an id of myCanvas?

1. var canvas = document.getElementById(‘myCanvas’);

var ctx = canvas.getContext(); ctx.rect(10, 10, 50, 75);

1. var canvas = document.getElementById(‘myCanvas’);

canvas.rectangle(10, 10, 50, 75);

1. var canvas = document.getElementById(‘myCanvas’); var ctx = canvas.getContext(‘2d’); ctx.fillRect(10, 10, 50, 75);
2. var canvas = document.getElementById(‘myCanvas’);

var ctx = canvas.getContext(); ctx.fillRect(10, 10, 50, 75);

Answer: C [Page 463

The following code will do the same

var canvas = document.getElementById(‘myCanvas’);

var ctx = canvas.getContext(); ctx.rect(10, 10, 50, 75); ctx.fill();

You can do it using path too.

]

Q26. Which one is not valid value of the lineJoin property of the canvas rendering context object?

1. square
2. round
3. bevel
4. miter Answer: A [Page 471

Allowable values round, bevel, or miter.]

Q27. Which one is the dafault value of the lineJoin property of the canvas rendering context object?

1. square
2. round
3. bevel
4. miter Answer: D

Allowable values round, bevel, or miter. Default is miter.]

Q28. You want to draw an arc that is approximately three-quarters of a circle. Which method is the easiest to use to accomplish this task?

1. arcTo()
2. arc()
3. circle()
4. dot() Answer: B [Page 485, 487

Using arcTo method you cannot draw more than the half of a circle Using arc method you can draw a circle or any part of a circle.]

Q29. You are drawing on canvas. You start the code like below

1. var canvas = document.getElementById('myCanvas');
2. ctx = canvas.getContext('2d');
3. ctx.strokeStyle = 'blue';
4. ctx.fillStyle = 'yellow';
5. ctx.lineWidth = 5; 6.

7.

8.

To draw an outline there quarter of a circle of radius 100 centered at point (400,300), which code should you add after line 4?

1. ctx.beginPath();

ctx.arc(400, 300, 100, 0, 3\*90);

ctx.stroke();

1. ctx.beginPath();

ctx.arc(400, 300, 100, 0, 270);

ctx.stroke();

1. ctx.beginPath();

ctx.arc(400, 300, 100, 0, 3\* Math.PI/2); ctx.stroke();

1. ctx.beginPath();

ctx.arc(400, 300, 100, 0, Math.PI);

ctx.stroke();

Answer: C [Page 486, 487

Angle is passed as radian]

Q30. How can you control the vertical alignment of the text in relation to a point when drawing text using the fillText method?

1. Using the textAlign property of the rendering context object
2. Using the verticalAlign property of the rendering context object
3. Using the textBaseline property of the rendering context object
4. Using the font property of the rendering context object Answer: C

textBaseline - Sets the vertical alignment of the text in relation to the coordinate that is passed into the fillText or strokeText method. Can be top, hanging, middle, alphabetic, ideographic, or bottom.]

Q31. What can be passed as source image as the first parameter to the drawImage method?

1. an <img> element
2. a <video> element
3. a <canvas> element
4. an image created using JavaScript Answer: A, B, C, D

[Page 490]

Q32. What happens when you pass a <video> element to the drawImage method as the source image?

1. a series of snapshot frames at intervals of 1 sec are displayed
2. a snapshot of last frame of the video is used as the image.
3. a snapshot of the frame that is currently displayed is used as the image.
4. A blank frame is drawn as a video is not supported by the method Answer: C

[Page 490]

Q33. is a container for the XML-based commands.

1. The <svg> element
2. The <canvas> element
3. The context rendering object of the <canvas> element
4. The context rendering object of the <svg> element Answer: A

[Page 496]

Q34. The content of the <svg> element is in which format?

1. SGML
2. HTML
3. XHMTL
4. XML Answer: D [Page 496]

Q35. Which attribute of the <path> element is used define the coordinates of points?

1. m
2. d
3. a
4. p Answer: B [Page 496]

Q36. How do you indicate the coordinates are relative while defining a path in svg?

1. Using the lowercase command
2. Using the the uppercase command
3. Using the first command in lowercase
4. By adding relative attribute to the <path> element Answer: A

[Page 496

When the lowercase command is used, it indicates that the coordinates are relative. When the uppercase command is used, it indicates that the coordinates are absolute.

L 100 100 moves to (100,100) from the top-left corner (0,0) l 100 100 moves to (100,100) from the current position]

Q37. Which one is the command in svg for closing a path from the current position to the beginning of the path?

1. M or m
2. C or c
3. L or l
4. Z or z Answer: D [Page 497]

Q38. What is the purpose of the viewBox attribute?

1. It define the maximum and minimum zoom size
2. It describes the default zoom scale of the drawing
3. It describes the part of the drawing that you want the user to see.
4. It describes the part of the drawing how to adjust when window is resized.

Answer: C [Page 501]

Q39. On the <svg> element, what attribute provides a window into the drawing and enables zoom capabilities?

1. zoom
2. window
3. viewBox
4. zoomWindow Answer: C

[Page 500]

Q40. You are drawing on HTML5 canvas you have the code like below: (Line numbers are illustrative purpose only)

* 1. var canvas = document.getElementById('canvas');
  2. var ctx = canvas.getContext('2d'); You want to draw an 200X100 rectangle like below



Which code segment you should add after Line 2?

1. var lg = ctx.createLinearGradient(0, 0, 200, 100);

lg.addColorStop(0, 'white'); lg.addColorStop(1, 'black'); ctx.fillStyle = lg; ctx.strokeStyle = 'black'; ctx.lineWidth = .5;

ctx.fillRect(0, 0, 200, 100);

ctx.strokeRect(0, 0, 200, 100);

1. var lg = ctx.createLinearGradient(0, 0, 100, 0); lg.addColorStop(0, 'white'); lg.addColorStop(1, 'black');

ctx.fillStyle = lg; ctx.strokeStyle = 'black'; ctx.lineWidth = .5;

ctx.fillRect(0, 0, 200, 100);

ctx.strokeRect(0, 0, 200, 100);

1. var lg = ctx.createLinearGradient(0, 0, 200, 0); lg.addColorStop(0, 'white'); lg.addColorStop(1, 'black');

ctx.fillStyle = lg; ctx.strokeStyle = 'black'; ctx.lineWidth = .5;

ctx.fillRect(0, 0, 200, 100);

ctx.strokeRect(0, 0, 200, 100);

1. var lg = ctx.createLinearGradient(0, 0, 200,200); lg.addColorStop(0, 'white');

lg.addColorStop(1, 'black'); ctx.fillStyle = lg; ctx.strokeStyle = 'black'; ctx.lineWidth = .5;

ctx.fillRect(0, 0, 200, 100);

ctx.strokeRect(0, 0, 200, 100);

Answer: C

[A – gradient will progress diagonally

B – gradient will be half of the width, the later half will be filled blacck C- correct

D – out of range]

Q41. You are drawing on HTML5 canvas. You have the canvas rendering object ctx in you your code. You want to fill a 400X400 rectangle with an image pattern.gif. Which code fragment should you use?

1. var img = new Image(); img.src = 'images/pattern.gif';

var ptrn = ctx.createPattern(img, 'repeat'); ctx.fillStyle = ptrn;

ctx.fillRect(0, 0, 400, 400);

1. var img = new Image(); img.src = 'images/pattern.gif'; img.onload = function () {

var ptrn = ctx.createPattern(img, 'repeat');

ctx.fillStyle = ptrn; ctx.fillRect(0, 0, 400, 400);

}

1. var img = new Image(); img.src = 'images/pattern.gif'; img.onload = function () { ctx.fillStyle = img; ctx.fillRect(0, 0, 400, 400);

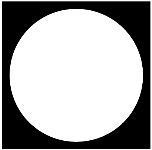
}

1. var img = new Image(); img.src = 'images/pattern.gif'; ctx.fillStyle = img; ctx.fillRect(0, 0, 400, 400);

Q42. Which method of canvas rendering object do you use to draw circle?

1. circle()
2. arc()
3. arcTo()
4. ellipse() Answer: B

Q43. Which of the following code draws the shapes like below?



1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<rect x="0" y="0" width="400" height="400" fill="back" />

<circle cx="200" cy="200" r="180" fill="white" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<circle cx="200" cy="200" r="180" fill="white" />

<rect x="0" y="0" width="400" height="400" fill="back" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<rect x="0" y="0" width="400" height="400" fill="back" />

<arc cx="200" cy="200" r="180" fill="white" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<rectangle x="0" y="0" width="400" height="400" fill="back" />

<circle cx="200" cy="200" r="180" fill="white" />

</svg> Answer: A

Q45. Look at the following shape



Which of the following svg definition does not render the above shape properly?

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<path d="M0 0 L100 0 L100 100 Z" fill="black" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<path d="M0 0 l100 0 l100 100 Z" fill="black" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<path d="M0 0 H100 V100 Z" fill="black" />

</svg>

1. <svg width="400" height="400" xmlns[="http](http://www.w3.org/2000/svg):/[/www.w3.org/2000/svg](http://www.w3.org/2000/svg)">

<path d="M0 0 h100 v100 Z" fill="black" />

</svg> Answer: B

Q46. On the <svg> element, what attribute provides a window into the drawing and enables zoom capabilities?

1. zoom
2. window
3. viewBox
4. zoomWindow Answer: C

[Page 500]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 13: Drag and drop



Q1. How do you indicate to the browser that an element can be dragged?

1. Set draggable attribute to true
2. Set draggable attribute to false
3. Set draggable attribute to auto
4. Add the draggable attribute Answer: A

[Page 509]

Q2. To specify to the browser that an element can be dragged, use the attribute.

1. draggable
2. allowdrag
3. drag
4. data-drag Answer: A

[Page 509]

Q3. What does it mean when you set the draggable attribute of an element to auto?

1. The element can be dragged
2. The element cannot be dragged
3. The browser will decide whether the element should be draggable
4. The element can be dragged to user desktop Answer: C

[Page 509

For most browsers, the auto setting is the default, which means that the browser decides whether the element should be draggable. For example, the <img> element is usually draggable by default, but a <div> element is not draggable by default]

Q4. How can you control whether the text of an element can be selected?

1. Using css user-select property
2. Using css display property
3. Using JavaScript onselect event handler
4. Using JavaScript onselect event handler Answer: A

[Page 509

The user-select property specifies whether the text of an element can be selected.

In web browsers, if you double-click on some text it will be selected/highlighted. This property can be used to prevent this.

Syntax

user-select: auto|none|text|all;

auto - Default. Text can be selected if the browser allows it

none - Prevent text selection

text - The text can be selected by the user

all - Text selection is made with one click instead of a double-click Example

Prevent text selection of a <div> element:

div {

-webkit-user-select: none; /\* Safari 3.1+ \*/

-moz-user-select: none; /\* Firefox 2+ \*/

-ms-user-select: none; /\* IE 10+ \*/

user-select: none; /\* Standard syntax \*/

}

For details https://developer.mozilla.org/en-US/docs/Web/CSS/user-select]

Q6. Which of the following is not the event of a dragged element?

1. dragstart
2. drag
3. dragend
4. drop Answer: D [Page 510

drop is the event of drop target element]

Q7. Which of the following is not an event of the drop target?

1. dragstart
2. dragenter
3. dragover
4. dragleave Answer: A [Page 510

dragstart is the event of drag source element]

Q8. Which of the following is not an event of the dragged element?

1. dragstart
2. drag
3. drop
4. dragend Answer: C [Page 510]

Q9. When the drag event is triggered?

1. Triggers when the drag is started
2. Triggers continuously as the element is being dragged
3. Triggers continuously as the element is dragged over the drop zone
4. Triggers when the drag enters a drop zone Answer: B

[Page 510]

Q10. Which of the following is not an event of the drop target element?

1. dragenter
2. drop
3. drag
4. dragleave Answer: C

[Page 511]

Q11. The dragenter and dragover events default to rejecting dragged items. How can you enable dropping?

1. By adding allowdrop attribute to the drop target
2. By cancelling the default action on these events
3. By setting css user-select property to none
4. By returning false from the event handler method Answer: B

[Page 511

Default action must be cancelled otherwise you can't drop. you should code like

document.getElementById(‘targetid’)

.ondragenter = (e) =>{

e.preventDefault();

}

document.getElementById(‘targetid’)

.ondragover = (e) =>{

e.preventDefault(); return true;

}

returning true is important in most browsers]

Q12. You are implementing HTML5 drag and drop. You have subscribed to dragenter event of the drop target like below

document.getElementById(‘drop-target’).addEventListener(‘dragenter’, function (evt) {

evt.preventDefault();

}, false);

Why do you call the preventDefault() on event object?

1. To prevent the rejection of the dragged items
2. To accept the dragged items
3. To cancel event bubble
4. None of the above Answer: A

[Page 512

The dragenter and dragover events default to rejecting dragged items, which is why you can’t

currently drop an item. You can enable dropping by cancelling the default action on these events.]

Q13. How can you pass data from the dragstart event to the drop event?

1. Using the DataTransfer object
2. Using a global variable
3. Using a local variable
4. All of the above Answer: A

[Page 513]

Q14. Which property of the dataTransfer object is used to set the type of drag operation and the cursor type?

1. dropEffect
2. effectAllowed
3. types
4. cursor Answer: A [Page 514]

Q15. Which one is not a valid value for the DataTransfer object’s effectAllowed property?

1. copy
2. move
3. paste
4. link Answer: C [Page 514

effectAllowed – type of operation copy/move/link/none]

Q16. Using the DataTransfer object, what kind of data can you pass to the drop event?

1. Any valid string, number, date/time, or Boolean value
2. Any URL that is within the same domain as the webpage
3. Any JSON object
4. Any object that can be represented as a string or URL Answer: D

[Page 517]

Q17. You are coding for HTML5 drag and drop in JavaScript. You have handled dragstart and drop event like below: (Line numbers are illustrative purpose only)

Line 1: draggableElement.on(‘dragstart’ function(evt){

Line 2: var id = evt.target.id;

Line 3: evt.dataTransfer.setData('text/plain', id); Line 4: });

Line 5: dropTrarget.on(‘drop’, function(evt){

Line 6:

Line 7: //code

Line 8: });

You want to capture the data passed in dragstart event at line 2 in drop event. Which code should you add at line 6?

1. var id = evt.dataTransfer.getData();
2. var id = evt.dataTransfer.getData(‘id’);
3. var id = evt.dataTransfer.getData(‘plain’);
4. var id = evt.dataTransfer.getData(‘text/plain’);

Answer: D [Page 515]

Q18. In which event, do you set the desired drag effect for the drag source?

1. dragstart
2. drag
3. dragend
4. dragmove Answer: A

[Page 510]

Q19. Which of the following is not a property on the File object?

1. name
2. path
3. type
4. size Answer: B [Page 517]

Q20. Which one is not a property of the File object?

The File object provides the name, type, and size properties

1. name
2. type
3. size
4. extension Answer: D

[Page 517]

Q21. You want to allow users to drop files on a division element in your web page. The html for the division is like below

<div id=’target’>

<p>Drop files here...</p>

</div>

You want to access the files a user dropped in drop event. You have code like below: (Line numbers are illustrative purpose only)

Line 1: document.getElementById(‘target’) Line 2: .addEventListener(‘drop’, function(evt){ Line 3:

Line 4: //mode code Line 4: }, false);

Which code should you add at line 3?

1. var files = evt.files
2. var files = evt.target.files
3. var files = evt.data.files
4. var files = evt.dataTransfer.files Answer: D

[Page 520]

Q22. Which object provides access to the dropped files?

1. DataTransfer
2. Data
3. Target
4. Window Answer: A

[Page 521]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 14: Making your HTML location-aware



Q1. What are the underlying location information sources of Geolocation?

1. locations inferred from network
2. radio frequency identification (RFID)
3. Wi-Fi and Bluetooth MAC addresses
4. All of the above Answer: D

[Page 539

Underlying location information sources such as the Global Positioning System (GPS) and locations inferred from network signals such as IP address, radio frequency identification (RFID), Wi-Fi and Bluetooth MAC addresses, and cellular IDs and from user input]

Q2. Which one of the following provides a way to integrate location services into a web page?

1. The Web storage API
2. The Geolocation API
3. The AppCache API
4. The Media Query API Answer: B

[Page 539]

Q3. Using the Geolocation API, you can get .

1. the current position of the server from which the application is serving the web page to the user device
2. the current position of the user or the device in which the application is running
3. the current position of the nearest GPS device available to the user device
4. the current position of the network server that provides user or the device internet gateway Answer: B

[Page 539]

Q4. You are creating a web application that will help users finding nearby points of interest or checking into their favorite social community sites. Which feature you will use?

1. The Web storage API
2. The Geolocation API
3. The AppCache API
4. The Media features Answer: B

[Page 539]

Q5. The Geolocation API is designed to support ?

1. one-shot position requests only.
2. repeated position updates only.
3. both one-shot position requests and repeated position updates.
4. None of above Answer: C

[Page 539]

Q6. You get a reference to the Geolocation API using

1. geolocation
2. window.geolocation
3. navigator.geolocation
4. window[‘geolocation’]

Answer: C [Page 540]

Q7. The Geolocation object is accessible by using the global variable?

1. window.geolocation
2. navigator.geolocation
3. browser.geolocation
4. location.geolocation Answer: B

[Page 540]

Q8. You want to determine whether the user’s browser provides support Geolocation API. Which of the following code does not accomplish it properly?

1. if('geolocation' in navigator) {

//code

}

1. if(navigator.geolocation) {

//Code

}

1. if(navigator['geolocation']) {

//Code

}

1. if(geolocation) {

//Code

}

Answer: D [Page 540

Property can be accessed using key in JavaScript window.location is equivalent to window[‘location’] Property can be enumerated using foreach in JavaScript foreach (var prop in window) {//work with prop}

So ‘geolocation’ in navigator will return true if supported otherwise false (null which evaluate to

false)]

Q9. Which one of the following is not a valid call to getCurrentPostion method of the Geolocation object?

1. getCurrentPosition()
2. getCurrentPosition(positionCallback)
3. getCurrentPosition(position Callback, positionErrorCallback)
4. getCurrentPosition(positionCallback, positionErrorCallback, positionOptions)

Answer: A [Page 541

getCurrentPosition(positionCallback, [positionErrorCallback], [positionOptions])

First parameter is mandatory, other two are optional

position Callback - is called after the current position is determined. positionErrorCallback - is called if any errors occur when trying to get the current position

position Options - lets you set some special options that control how the getCurrentPosition method behaves.]

Q10. Which of the following is not the methods of the Geolocation object?

1. getCurrentPosition()
2. watchPosition()
3. clearWatch()
4. LatLng() Answer: D [Page 540

LatLang() is the method of Google map API, actually it is a object constructor]

Q11. What is not a parameter of the Geolocation object’s getCurrentPosition method?

1. A success callback
2. An error callback
3. options array
4. timestamp Answer: D

[Page 540]

Q12. Which method continuously monitors your current location from the Geolocation object?

1. watchPosition()
2. watchLocation()
3. getCurrentPosition()
4. getCurrentLocation()

Answer: A [Page 541, 546]

Q13. What is the type of the parameter that the success callback of the getCurrentPostion or watchPostion method accepts?

1. Geoposition
2. Coordinate
3. Latitude
4. Longitude Answer: A

[Page 540]

Q14. Which property of the Geopostion object allows to access information about current position like longitude, latitude etc.?

1. position
2. longitude
3. latitude
4. coords

Answer: D [Page 540]

Q15. Which of the following are the members of Geoposition object?

1. coords
2. timestamp
3. longitude
4. latitude Answer: A, B [Page 541]

Q16. What does the coords property of the Geoposition return?

1. a Coordinates object
2. a LatLng object
3. a Map object
4. a JSON object Answer: A

[Page 541]

Q17. Which one is not a member of the Coordinate object?

1. latitude
2. longitude
3. altitude
4. enableAccuracy Answer: D

[Page 541

latitude - Gets the latitude in decimal degrees longitude - Gets the longitude in decimal degrees altitude - Gets the height in meters

accuracy - Gets the accuracy of the coordinates in meters altitudeAccuracy Gets the accuracy of the altitude in meters heading Gets the direction of travel in degrees

speed - Gets the speed of travel in meters/second]

Q18. Which property of the Geoposition of the direction of travel?

1. direction
2. travelDirection
3. heading
4. travel Answer: C [Page 541]

Q19. How do you control how the getCurrentPosition method behaves?

1. By setting a PositionOptions object to the Geolocation object’s option property
2. By passing a PositionOptions object as the third parameter while calling the getCurrentPosition method
3. By setting a PositionOptions object to the options property of the Geoposition before calling the getCurrentPosition method
4. By passing a PositionOptions object as a parameter to the success callback for the getCurrentPosition method

Answer: B [Page 544]

Q20. Which property of the PositionOptions object is used to indicate to use a cached result if available?

1. timeout
2. enableHighAccuracy
3. maximumAge
4. cache Answer: C [Page 543

enableHighAccuracy This causes the method to be more resource intensive if set to true. The default is false. If true, the getCurrentPosition method tries to get as close as it can to the actual location.

timeout This specifies a timeout period for how long the getCurrentPosition method can take to complete. This number is measured in milliseconds and defaults to zero. A value of zero represents infinite.

maximumAge If this is set, the API is being told to use a cached result if available, rather than make a new call to get the current position. The default is zero, so a new call is always be made. If maximumAge is set to a value and the cache isn’t older than the allowable age, the cached copy is used. This value is measured in milliseconds.]

Q21. What is the default value of the maximumAge property of PositionOptions object?

1. -1
2. 0

C. 1000

D. 5000

Answer: B [Page 545]

Q22. Which one always get the updated current position?

1. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{});
2. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{}, {timeout:0, maximumAge:1000} );
3. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{},{maximumAge:5});
4. navigator.geolocation.getCurrentPosition((pos) =>{} , (err)

=>{},{timeout:1000,maximumAge:5000});

Answer: A [Page 544]

Q23. Which one correctly get the new position if cache is older than 5 seconds?

1. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{});
2. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{}, 5000);
3. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{},{maximumAge:5});
4. navigator.geolocation.getCurrentPosition((pos) =>{} , (err) =>{},{maximumAge:5000});

Answer: D [Page 545]

Q24. The default value of the maximumAge of the PositionOptions object is set to 0. What does it indicate?

1. It means that a cached location is used if available.
2. It means that a cached location is not used.
3. It means that it will prompt user to decide whether a cached location is to be used or not
4. None of the above Answer: B

[Page 545]

Q25. Which of the following calls the getCurrentPosition method of the Geolocation object?

1. navigator.geolocation.getCurrentPosition(function (position) {

// Code to access co-ordinates

},

function (err) {

//Error handling

},

{enableHightAccuracy: true, timeout: 3000, maximumAge:20000 });

1. navigator.geolocation.getCurrentPosition({enableHigtAccuracy: true, timeout: 3000, maximumAge: 20000 },

function (position) {

//Code to access co-ordinates

}, function (err) {

//Error handling

});

1. navigator.geolocation.getCurrentPosition(function (position) {

// Code to access co-ordinates

},

{enableHighAccuracy: true, timeout: 3000, maximumAge: 20000 });

1. navigator.geolocation.getCurrentPosition(function () {

// Code to access co-ordinates ;

}, function (err) {

//Error handling

},

{enableHighAccuracy: true, timeout: 3000, maximumAge: 20000 });

Answer: A [Page 545]

Q26. You are working with Geolocation object. You want to know the user’s current position. You

have the following code: (line numbers are illustrative purpose only) Line 1: if(navigator['geolocation']) {

Line 2: navigator.geolocation.getCurrentPosition(function (position) { Line 3:

Line 4:

Line 5: //code to work with latitude and longitude Line 6: }, function (err) {

Line 7: //Error handling code

Line 8: });

Line 9: }

Which code segment should you add after line 2?

1. var lat = position.latitude; var lng = position.longitude;
2. var lat = position.coords.latitude; var lng = position.coords.longitude;
3. var lat = position.LatLng.latitude; var lng = position. LatLng.longitude
4. var latlng = new LatLng(position.coords); var lat = latlng.latitude;

var lng = latlng.longitude;

Answer: B [Page 542]

Q27. Which method of the Geolocation object is used to retrieve continuous position updates?

1. getCurrentPosition
2. watchPosition
3. clearWatch
4. watch Answer: B [Page 546]

Q28. Which of the following is or are true?

1. getCurrentPosition() of the Geolocation object calls one-time the success callback with the current position
2. getCurrentPosition() of the Geolocation object calls continuously calls the success callback with the current position
3. watchPosition() of the Geolocation object calls one-time the success callback with the current
4. watchPosition() of the Geolocation object calls continuously calls the success callback with the current position

Answer: A, D [Page 540]

Q29. Which of the following formulas can you use to calculate the distance between two samples?

1. haversine
2. Pythagorean theorem
3. quadratic
4. hyperbolic Answer: A

[Page 548]

Q30. What does the watchPosition method return?

1. A Geoposition object
2. A Coordinate object
3. A PositionOptions object
4. An id which is used in clearWatch method to stop monitoring user’s location

Answer: A

[Page 546, 549]

Q31. Which method monitors your location?

1. getCurrentPosition
2. watchPosition
3. clearWatch
4. watch Answer: B [Page 549]

Q32. How do create a Map object using Google Map API?

1. By using google.Map class
2. By using google.maps.Map class
3. By using google.maps.LatLng
4. By using Map class Answer: B

[Page 512]

Q33. You want to show Google map pointing user position in your application. You have the following code

1. navigator.getCurrentPosition ((position)=>{ 2.

3. });

Which code should you write at line 2?

* 1. var map = new google.maps.Map(document.getElementById('map'),position);
  2. var map = new google.maps.Map(document.getElementById('map'), position.latitude,position.longitude);
  3. var map = new google.maps.Map(document.getElementById('map'), {lat: position.latitude, lng: position.longitude });
  4. var map = new google.maps.Map(document.getElementById('map'), { center: {lat: position.latitude, lng: position.longitude}});

Answer: D [Page 512

Google Map constructor google.maps.Map(mapcontainer, [mapOption]) MapOption center property is LatLng set objects LatLng.lat - latitude

LatLng.lng - longitude]

# Programming in HTML5 with JavaScript and CSS3

## Chapter 15: Local data with web storage



Q1. Which one is an API for storing web page data locally?

1. Geolocation API
2. Web storage API
3. AppCache API
4. Filesystem API Answer: B

[Page 555, 558]

Q2. What are the two forms of storage the Web storage API provides?

1. localStorage
2. sessionStorage
3. cookie
4. appCache Answer: A, B [Page 564]

Q3. Which one is also known as DOM storage?

1. Geolocation
2. Web storage
3. AppCache
4. Filesystem Answer: B

[Page 555]

Q4. Which one has been for years for storing small bits of information on client device?

1. Cookie
2. Web storage
3. Web sql
4. Filesystem Answer: A

[page 556]

Q5. What is maximum size of data for storing cookie data?

1. 1 KB
2. 2 KB
3. 4 KB
4. Unlimited Answer: C

[Page 557]

Q6. Which one use

s key/values pairs to store data on browser?

1. Web storage
2. Web SQL
3. Indexeddb
4. Filesystem API Answer: A

[Page 558]

Q7. Which one provides full power of relational database for storing complex data on browser?

1. Web storage
2. Web SQL
3. Indexeddb
4. Filesystem API Answer: B

[Page 558]

Q8. Which one provides way to store complex data on browser as a non-relational database but with indexing and transaction support?

1. Web storage
2. Web SQL
3. Indexeddb
4. Filesystem API Answer: C

[Page 558]

Q9. What is the difference between the localStorage and sessionStorage?

1. localStorage is cleared when the session is closed, whereas sessionStorage is still be accessible after a session closes
2. sessionStorage is cleared when the session is closed, whereas localStorage is still be accessible after a session closes
3. sessionStorage is accessible within the current is closed, whereas localStorage is accessible in the all open tab
4. localStorage is accessible within the current is closed, whereas sessionStorage is accessible in the all open tab

Answer: B [Page 564]

Q10. Which of following URLs can access that data storage that was created using the URL [http://www.example.com/area1/page1.html?](http://www.example.com/area1/page1.html)

1. <http://store.example.com/area1/page1.html>
2. <http://example.com/area1/page1.html>
3. <http://www.example.com/area1/page2.html>
4. <http://www.example.com/area2/page1.html> Answer: C, D

[Page 559]

Q11. Which method removes all the item that has been saved in localStorage?

1. clear()
2. removeAll()
3. abandon()
4. reset() Answer: A [Page 560]

Q12. Which one should you use to retrieve the stored at index position 3?

1. localStorage.get(3);
2. localStorage.getItem(3);
3. localStorage[3];
4. localStorage.getItem(localStorage.key(3));

Answer: D [Page 561]

Q13. How much minimum size for localStorage most browsers allows?

1. 5 KB
2. 1 MB
3. 5 MB
4. Unlimited Answer: C

[Page 561]

Q14. Which property of localStorage do you use to determine whether any value is stored local storage?

1. length
2. size
3. value
4. key Answer: A [Page 560]

Q15. You want to read all items from localStorage if any value stored. Which code should you use?

1. if(localStorage){ //code}
2. if(localStorage>0){ //code}
3. if(localStorage.length>0){ //code}
4. if(localStorage.key>0){ //code}

Answer: C [Page 560]

Q16. You have stored data to localStorage with key x, which of the following cannot retrieve that data?

1. localStorage.getItem(‘x’)
2. localStorage[‘x’];
3. localStorage.x
4. localStorage.get(‘x’)

Answer: D [Page 560

localStorage.getItem("x"),localStorage.x, localStorage["x"] does the same but localStorage.get("x") does not]

Q17. What type of data can be stored in web storage?

1. object
2. json
3. string
4. any type of data Answer: C

[Page 562

Currently, only string values can be stored in web storage]

Q18. To store complex JSON object in web storage which method is used?

1. JSON.toString()
2. JSON.parse()
3. JSON.toObject()
4. JSON.stringify()

Answer: D [Page 562]

Q19. How do you store JSON object to local storage?

1. Simply store it with a key
2. Use JSON.stringify() method to convert the object to a string that can then be placed in web storage
3. Use toString()to convert the object to a string that can then be placed in web storage
4. None of the above Answer: B

[Page 562]

Q20. Which type of storage data is sandboxed to only the current tab or window and is cleared when closed?

1. localStorage
2. sessionStorage
3. offline cache
4. cookie Answer: B [Page 564]

Q21. What is the web storage limit currently recommended by the World Wide Web Consortium (W3C)?

1. 4 KB
2. 5 MB
3. 500 MB
4. 10 MB Answer: B [Page 561]

Q22. What is the correct syntax for removing all values existing in localStorage?

1. localStorage.clear();
2. localStorage.removeAll();
3. localStorage.abandon();
4. localStorage.reset(); Answer: A

Q23. Which of the following storage mechanisms has the highest level of cross-browser support?

1. Web storage
2. Web SQL
3. IndexDB
4. FileSystem API Answer: A

[Page 558]

Q24. Which of the following features does web storage support?

1. Indexing
2. Transactions
3. Asynchronous read/write
4. Simple key/value pair storage Answer: D

[Page 558]

Q25. Which one is not true web storage?

1. Only string can be stored
2. operates asynchronously
3. no transaction support
4. Does not support indexing Answer: B

[Page 564]

Q26. How can you keep in sync all the open tab with localStorage data so that if onetab modified data other tabs are updated accordingly?

1. periodically check storage update if any modification found
2. use insert, update callback
3. use StorageEvent
4. Any one of the above Answer: C

[Page 566]

Q27. Which of the following is not a property of the StorageEvent object?

1. oldValue
2. key
3. changeType
4. storageArea Answer: C

[Page 566]

Q28. Which of the following is the correct way to cancel a storage event?

1. event.returnValue = false;
2. event.preventDefault();
3. event.stopPropagation();
4. Storage events cannot be canceled after they are triggered. Answer: D

Q29. Which web storage is accessible to only the current tab or window and is cleared when closed?

1. HTTP Cookie
2. FileSystem API
3. localStorage
4. sessionStorage Answer: D

[Page 564]

Q30. Web storage supports advanced features such as transactions or indexing?

1. True
2. False Answer: B [Page 564]

Q31. What is the correct syntax for removing all values existing in *localStorage*?

1. localStorage.clear();
2. localStorage.removeAll();
3. localStorage.abandon();
4. localStorage.reset();

Answer: A [Page 560]

Q32. Which of the following features does web storage support?

1. Indexing
2. Transactions
3. Asynchronous read/write
4. Simple key/value pair storage Answer: D

[Page 564]

Q33. Which web storage provide the power of a full relational database, including support for SQL commands, transactions, and performance tuning?

1. Web sql
2. indexedDB
3. localStorage
4. sessionStorage Answer: A

[Page 558]

Q34. Which of the following is true about Session Storage in HTML5?

1. HTML5 introduces the sessionStorage attribute which would be used by the sites to add data to the session storage.
2. It will be accessible to any page from the same site opened in that window i.e. session.
3. As soon as you close the window, session would be lost.
4. All of the above.

Answer: C [Page 562]

1. Cookies are included with every HTTP request, thereby slowing down your web application by transmitting the same data.
2. Cookies are included with every HTTP request, thereby sending data unencrypted over the internet.
3. Cookies are limited to about 4 KB of data . Not enough to store required data.
4. All of the above.

Answer: C [Page 557]

Q36. You would like to store the user’s name after he authenticates on your site, but he will need to authenticate again on his next visit, at which time you would reload his information (including name). Which storage mechanism should you use??

1. Web sql
2. indexedDB
3. localStorage
4. sessionStorage Answer: D

[Page 565]