What will be the output of the following code:  
var arr = [10, 20, 30, 40];  
var i = 1;  
do {  
console.log(arr[i] / arr[i - 1]);  
i++;  
} while (i < arr.length);  
\*

1/1

2

1.5 2

Error

2 1.5

What will be the output ?  
<div class="container">  
  <p>Hello World</p>  
  <p>Goodbye World</p>  
</div>  
<script>  
  console.log($("div p").slice(1,2).text());  
</script>  
\*

1/1

undefined

World

Goodbye World

Hello World

What is the correct way to listen for incoming requests on a specified port for the HTTP server in Node.js?  
  
\*

0/1

server.listen(3000);

server.listen(3000, function() {});

server.listen(function() {});

server.listen();

Correct answer

server.listen(3000, function() {});

How do you set the scrolling behavior of an <iframe> tag?  
\*

1/1

<iframe src="url" style="overflow:scroll;"></iframe>

<iframe src="url" navigation="scroll"></iframe>

<iframe src="url" move="auto"></iframe>

<iframe src="url" flow="scroll"></iframe>

Which of the following is a higher-order function in JavaScript?  
\*

1/1

A function that returns an array

A function that takes a function as an argument

A function that returns a string

A function that returns a number

What is the main difference between the PUT and PATCH methods in HTTP?  
\*

0/1

PUT can only be used with a request body, while PATCH can be used with or without a request body

PUT is idempotent, while PATCH is not

PUT is used for creation, while PATCH is used for updates

PUT replaces the entire resource, while PATCH modifies only a portion of it

Correct answer

PUT is idempotent, while PATCH is not

What will be the output ?

<div id="example">Hello World</div>  
  
<script>  
$(document).ready(function(){  
  $("#example").text(function(i, origText){  
    return "Old text: " + origText + " New text: Hello jQuery";  
  });  
});  
</script>  
\*

1/1

Error

Hello World

None of the above

Old text: Hello World New text: Hello jQuery

What is the output of the following code:  
  
let x = 10;  
let obj = { x: 20, foo: function() { console.log(this.x); } };  
obj.foo.call({ x: 30 });  
\*

1/1

undefined

30

20

10

Can multiple animation effects be combined in a single call to the animate() method in jQuery?  
\*

1/1

Yes, multiple effects can be combined

No, multiple effects require multiple calls to animate()

No, only one effect can be specified per call to animate()

Yes, but only if the effects are of the same type

What is the output of the following code:  
var person = {name:"John", age:30, city:"New York", hobbies:["reading", "traveling"]};  
var x;  
for (x in person) {  
  if (x === "hobbies") {  
    console.log(person[x]);  
    break;  
  }  
}  
\*

1/1

[reading, traveling]

None of the above

reading traveling

New York

What is the difference between .detach() and .remove() methods in jQuery?  
\*

1/1

.detach() keeps the data and events of the removed element, whereas .remove() does not.

Both methods are the same.

.detach() does not keep the data and events of the removed element, whereas .remove() does.

.detach() is used to remove elements from the DOM, whereas .remove() is used to remove data from elements.

How do you access the value of an input in ReactJS?  
\*

1/1

By using event.target.value

By using this.event.value

By using this.state.value

By using event.value

How can you prevent the default form submission behavior in ReactJS?  
\*

1/1

By using stopPropagation() on the form element.

By using preventDefault() on the form element.

By using preventDefault() on the submit button.

By using stopPropagation() on the submit button.

What is the CSS code to set the width of an element with class "box" to 50% and height to 300 pixels, and center it vertically and horizontally?  
\*

1/1

.box {width: 300px; height: 50%; margin: 0 auto;}

.box {width: 300px; height: 50%; text-align: center;}

.box {width: 50%; height: 300px; margin: auto; position: absolute; top: 0; left: 0; right: 0; bottom: 0;}

.box {width: 50%; height: 300px; align: center; valign: center;}

What is the default value of the "cellspacing" attribute in a HTML table?  
\*

0/1

0

1

2

3

Correct answer

0

What is the main difference between a function component and a class component in React?  
\*

1/1

Class components can handle lifecycle methods while function components cannot.

Class components must always render a component while function components have the option to return null.

Function components are stateless while class components can have state.

Function components can handle user-defined props while class components cannot.

In a React component, which of the following is the recommended way to update state based on the previous state?  
\*

1/1

this.setState({...});

this.setState(this.state = {...});

this.state = {...};

this.setState(previousState => {...});

What will be the output ?

<button id="myButton">Click Me</button>  
  
<script>  
$("#myButton").click(function(){  
  $("#myButton").text("Hello World");  
});  
</script>  
\*

1/1

None of the above

"Hello World"

Error

"Click Me"

Which of the following is not a method in the XMLHttpRequest object?\*

1/1

receive()

send()

setRequestHeader()

open()

How do you use the grid system in Bootstrap to create a layout with 2 equal-width columns?  
\*

1/1

By using the "row" and "col-6" classes

By using the "container" and "col-md-6" classes

By using the "container-fluid" and "col-lg-6" classes

By using the "row-fluid" and "span6" classes

In a Bootstrap grid, what is the difference between a container class and a row class?  
\*

0/1

The container class sets the overall number of columns in the grid while the row class sets the number of columns occupied by individual grid cells.

The container class sets the layout direction of the grid while the row class sets the alignment of individual grid cells.

The container class sets the overall height of the grid while the row class sets the height of individual grid cells.

The container class sets the overall width of the grid while the row class sets the width of individual grid cells.

Correct answer

The container class sets the overall width of the grid while the row class sets the width of individual grid cells.

What is the difference between using arrow functions and bind() in ReactJS event handlers?  
\*

1/1

Arrow functions are less efficient than bind() in terms of performance.

Arrow functions can access state directly, while bind() must be passed state as an argument.

There is no difference between arrow functions and bind() in ReactJS event handlers.

Arrow functions automatically bind the component instance, while bind() does not.

What is hoisting in JavaScript?  
\*

1/1

The process of moving declared variables and functions to the bottom of their scope

The process of moving declared variables and functions to the middle of their scope

The process of moving declared variables and functions to the top of their scope

The process of moving declared variables and functions to a separate scope

What are the three principles of Redux?  
\*

1/1

Event handling, styling, and server communication.

Single source of truth, immutability, and pure functions.

State management, styling, and server communication.

Immutability, state management, and server communication.

Which of the following is true about Props, State, and Context in React components?  
\*

1/1

Props are used to pass data from parent to child component, State is used to pass data through the component tree without having to pass props down manually at every level, and Context is used to store component-specific data.

Props are used to pass data from parent to child component, State is used to store component-specific data, and Context is used to pass data through the component tree without having to pass props down manually at every level.

Props are used to store component-specific data, State is used to pass data from parent to child component, and Context is used to pass data through the component tree without having to pass props down manually at every level.

Props are used to pass data through the component tree without having to pass props down manually at every level, State is used to store component-specific data, and Context is used to pass data from parent to child component.

What is the correct syntax to specify multiple CSS classes to be applied to an HTML element?  
\*

1/1

<element class="class1 class2">

<element class="class1, class2">

<element class1="class1" class2="class2">

<element class="class1 + class2">

What is the output of the following code:  
 let x = 10;  
let obj = { x: 20, foo: function() { console.log(this.x); } };  
let bar = obj.foo;  
let baz = { x: 30 };  
baz.bar = bar;  
baz.bar();  
\*

0/1

10

undefined

20

30

Correct answer

30

What is the output of the following code:  
  
var person = {name:"John", age:30, city:"New York", hobbies:["reading", "traveling"]};  
var str = JSON.stringify(person, ["hobbies"]);  
console.log(str);  
\*

1/1

{"name":"John","age":30,"city":"New York","hobbies":["reading","traveling"]}

{"hobbies":["reading","traveling"]}

None of the above

{"name":"John"}

How can you define a route in an ExpressJS application?  
\*

1/1

Using the app.get() method.

Using the [router.post](https://www.google.com/url?q=http://router.post&sa=D&source=editors&ust=1689874671598083&usg=AOvVaw0hGmMWu1nKT8XcfGeSdaWg)() method.

Using the server.put() method.

Using the req.delete() method.

What HTTP methods are considered to be safe methods?  
\*

1/1

POST and PUT

OPTIONS and TRACE

GET and HEAD

DELETE and PATCH

What HTTP status code indicates that the server is unable to fulfill the request because it is dependent on another resource that has failed?  
\*

0/1

502 Bad Gateway

503 Service Unavailable

510 Not Extended

504 Gateway Timeout

Correct answer

502 Bad Gateway

What is the correct way to create an HTTP server in Node.js using the HTTP module?  
\*

1/1

var server = http.createServer(function (request, response) {});

var server = http.Server();

var server = new http.Server();

http.createServer();

What is the correct way to pass arguments to a ReactJS event handler?  
\*

1/1

onClick={this.handleClick.bind(this, arg1, arg2)}

onClick={this.handleClick(this, arg1, arg2)}

onClick={this.handleClick(arg1, arg2)}

onClick={() => this.handleClick(arg1, arg2)}

What will be the output of the following code:  
var str1 = "Hello";  
var str2 = " World";  
var res = "";  
for (var i = 0; i < str1.length; i++) {  
res += str1[i];  
}  
for (var i = 0; i < str2.length; i++) {  
res += str2[i];  
}  
console.log(res);  
\*

1/1

Hello World

Error

ellH dlroW

World Hello

What is the CSS code to set the background image of an element with id "header" to "header-bg.jpg", repeat it both vertically and horizontally, and set the background color to lightgray as a fallback?  
\*

1/1

#header {background-image: url("header-bg.jpg") repeat both; color: lightgray;}

#header {background: url("header-bg.jpg") repeat; background-color: lightgray;}

#header {bg-image: url("header-bg.jpg") repeat-x repeat-y; bg-color: lightgray;}

#header {background-image: url("header-bg.jpg"); background-repeat: repeat-x repeat-y; fallback-color: lightgray;}

How do you create a controlled component in ReactJS to handle form inputs?  
\*

1/1

By using the onClick event to update the state with the input value

By using the onChange event to update the state with the input value

By using the onSubmit event to update the state with the input value

By using the onInput event to update the state with the input value

What is the correct way to write to a file asynchronously in Node.js?  
\*

0/1

fs.writeFile('file.txt', data, sync = true)

fs.writeFileSync('file.txt', data)

fs.writeSync('file.txt', data)

fs.write('file.txt', data, async = true)

Correct answer

fs.writeFileSync('file.txt', data)

What will be logged to the console in the following code?  
  
class MyParentComponent extends React.Component {  
constructor(props) {  
  super(props);  
  this.state = { message: 'Hello World' };  
}  
  
handleClick() {  
  this.setState({ message: 'Goodbye World' });  
}  
  
render() {  
  return (  
    <MyChildComponent message={this.state.message} onClick={() => this.handleClick()} />  
  )  
}  
}  
  
function MyChildComponent (props) {  
console.log(props);  
return (  
  <div>  
    <div>{props.message}</div>  
    <button onClick={props.onClick}>Change Message</button>  
  </div>  
)  
}  
\*

1/1

{ message: 'Goodbye World', onClick: [Function] }

{ onClick: [Function] }

{ message: 'Hello World' }

{ message: 'Hello World', onClick: [Function] }

What are the different ready states of a request in AJAX?  
\*

1/1

No ready state.

1

3

5

What is the output of the following code ?  
  
File: employee.js  
exports.name = 'John Doe';  
exports.age = 30;  
exports.designation = 'Manager';  
exports.getDetails = function() {  
return Name: ${this.name}, Age: ${this.age}, Designation: ${this.designation};  
};  
  
Code:  
var employee = require('./employee');  
console.log(employee.getDetails.bind({name: 'Jane Doe', age: 28})().designation);  
\*

0/1

Undefined

Jane Doe

Error

Manager

Correct answer

Undefined