

## Search questions from Quizizz Library

Enter topic name

Search questions

☐

1. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

## What is jQuery?

## Answer choices

- |  |                                |
|--|--------------------------------|
| A JavaScript library for HTML document manipulation and event handling | A backend programming language |
| A CSS framework for styling web pages                                  | A replacement for JavaScript   |

## Answer explanation

jQuery is a **lightweight JavaScript library** that simplifies **HTML DOM manipulation, event handling, animations, and cross-browser compatibility**. It is **not** a backend language, a CSS framework, or a replacement for JavaScript but rather a tool that enhances JavaScript functionality.

☐

2. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

Which symbol does jQuery use as a shortcut for the jQuery function?

## Answer choices

- |   |    |
|---|----|
| % | \$ |
| ? | @  |

## Answer explanation

In jQuery, the **\$ symbol** is used as an alias for the jQuery function. This allows selecting elements and applying jQuery methods in a concise way.

☐

3. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

Which jQuery function is used to prevent code from running, before the document is finished loading?

## Answer choices

- |                      |                        |
|----------------------|------------------------|
| \$(body).onload()    | \$(document).load()    |
| \$(document).ready() | \$(document).loading() |

## Answer explanation

The `$(document).ready()` function ensures that the jQuery code runs **only after the DOM (HTML elements) is fully loaded**, preventing errors caused by trying to manipulate elements before they exist.

☐

4. Open Ended

30 seconds

1 point

AI

☐

☐

Edit

☐

Fill in the missing jQuery code to hide only the paragraph that is clicked:

```
$(document).ready(function(){
});
```

Evaluate responses using AI: **OFF**

#### Answer explanation

```
$(document).ready(function(){
$("p").click(function(){
$(this).hide();
});
});
```

☐

5. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

Which jQuery statement returns a node or nodes with class **q13**?

#### Answer choices

`$("q13")`

`$(".q13")`

`$("#q13")`

`$("@q13")`

☐

6. Open Ended

30 seconds

1 point

AI

☐

☐

Edit

☐

Fill in the missing jQuery code to hide all paragraph elements when the mouse enters any of them:

```
$(document).ready(function(){
    $("p").mouseenter(
        // add code here
    )
});
```

Evaluate responses using AI: **OFF**

#### Answer explanation

```
$(document).ready(function(){
$("p").mouseenter(function(){
$("p").hide();
});
});
```

☐

7. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

What is React?

Answer choices

A backend framework for Node.js

A database management system

A CSS framework for styling web pages

A JavaScript library for building **interactive and reusable UI components** in web applications

Answer explanation

React is a **JavaScript library** developed by **Meta (formerly Facebook)** for **building user interfaces**.

It follows a **component-based architecture**, allowing developers to create **reusable UI components**.

React uses the **Virtual DOM** to efficiently update web applications without reloading the page.

It is commonly used in **single-page applications (SPAs)**.

☐

8. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

Which command is used to create a new React app?

Answer choices

react new my-app

npm create-react-app my-app

npm create-react-app my-app

react-init my-app

☐

9. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

What is JSX (JavaScript XML) in React?

Answer choices

A CSS preprocessor

A database management system

A state management library

A syntax extension that allows writing HTML in JavaScript

Answer explanation

**JSX (JavaScript XML)** is a **syntax extension** for JavaScript used in React.

It allows developers to write **HTML-like code** inside JavaScript files.

JSX makes it easier to create **React components** by defining the UI structure in a readable format.

☐

10. Multiple Choice

30 seconds

1 point

AI

☐

☐

Edit

☐

How do you update the state in a React component using useState?

Answer choices

setValue(newValue);

this.state.value = newValue;

state.value = newValue;

updateState(newValue);

Answer explanation

In React, the **useState hook** provides a function to update the state. If value is a state variable, it is updated like this:

```
const [value, setValue] = useState(0); // Initialize state
```

```
setValue(10); // Correct way to update state
```

Calling setValue(newValue) **triggers a re-render**, ensuring the UI updates with the new state.

11. Multiple Choice 30 seconds 1 point

AI

Edit

What is the primary purpose of React Router?

Answer choices

To manage global state in a React application

To optimize React performance

To handle navigation and routing in a React application

To fetch data from APIs

Answer explanation

**React Router** is a **client-side routing library** that enables **navigation between different views (pages)** in a React app **without reloading the page**. It allows developers to define **routes (<Route> components)** and manage **URL-based rendering**.

12. Multiple Choice 30 seconds 1 point

AI

Edit

Which component is responsible for defining a route, and which component is responsible for wrapping all routes in React Router?

Answer choices

<Route> and <Router>

<NavLink> and <Switch>

<Route> and <Routes>

<Link> and <Routes>

Answer explanation

**<Route>** is used to **define a route** in React Router, specifying which component should render based on the URL.

**<Routes>** is used to **wrap all <Route> components**, ensuring that only the first matching route is rendered.

13. Multiple Choice 30 seconds 1 point

AI

Edit

Which component is used to create navigational links in React Router?

Answer choices

<a>

<Navigate>

<Route>

<Link>

14. Multiple Choice 30 seconds 1 point

AI

Edit

What is the purpose of the useState hook in React?

Answer choices

To fetch data from an API

To store a local state value and provide a function to update it

To define routing paths in a React app

To apply styles dynamically

## Answer explanation

The useState hook is used in **functional components** to manage **local state**.

It returns **two values**:

**1. The current state value**

**2. A function to update the state**

When the state is updated, React **re-renders** the component to reflect the changes.

☐

15. Multiple Choice

30 seconds

1 point

AI

☐☐

Edit

☐

Which is the correct way to attach an event handler to a button in React?

```
function handleClick() {  
  alert("Button clicked!");  
}
```

## Answer choices

<button onClick="handleClick()">Click Me</button>

<button onClick={handleClick}>Click Me</button>

<button onclick="handleClick()">Click Me</button>

<button Click={handleClick}>Click Me</button>

## Answer explanation

In **React**, event handlers **must be written in camelCase**, so onClick is used instead of onclick.

The event handler **should be passed as a function reference** (onClick={handleClick}), not as a string.

☐

16. Multiple Choice

30 seconds

1 point

AI

☐☐

Edit

☐

What are props in React?

## Answer choices

A way to store component state

A function that updates the state

A built-in React method for handling events

Short for "properties," used to pass data between components

## Answer explanation

**Props** (short for "**properties**") allow **data to be passed** from a **parent component** to a **child component** in React.

Props are **read-only**, meaning a component **cannot modify its own props**.

They make components **dynamic and reusable**.

☐

17. Multiple Choice

30 seconds

1 point

AI

☐☐

Edit

☐

In the following code, how are posts and setPosts being used?

```
import { useState } from "react";  
import { BrowserRouter as Router, Route, Routes } from "react-router-dom";  
import Info from "./Info";
```

```
function App() {
  const [posts, setPosts] = useState([]);
  return (
    <BrowserRouter>
    <Routes>
    <Route path="/info" element={<Info posts={posts} setPosts={setPosts} />} />
    </Routes>
    </BrowserRouter>
  );
}
```

**Answer choices**

They are variables passed through URL parameters

They are event handlers for form submission

They are local variables being passed as props, allowing the Info component to create and manage its own independent state separate from the parent component.

They are state variables being passed as props, allowing the Info component to access and update the state maintained in the parent component.

**Answer explanation**

Since setPosts is passed down, the child component (Info) can **modify the state** that belongs to the parent component (App).



18. Open Ended

3 minutes

1 point

AI



Edit



React Programming Challenge: User Profile App

**Problem Statement:**

You are building a simple **User Profile App** using React. The app should:

- (1) **Display the user's name and age.**
- (2) **Allow the user to update their name** using an input field.
- (3) **Use React state (useState)** to manage the name.
- (4) **Pass props correctly** between components.

Starter Code (Complete the missing parts)

1. **App.js** (Manages State and Composition)

```
import { useState } from "react";
import ProfileHeader from "./ProfileHeader";
import ProfileEditor from "./ProfileEditor";
function App() {
  // TODO: Initialize state for 'name' with "John Doe"
  /* COMPLETE THIS LINE */
  return (
    <div style={{ textAlign: "center", padding: "20px" }}>
      /* Pass 'name' as a prop to ProfileHeader */
      <ProfileHeader name={/* COMPLETE THIS PROP */} age={25} />
      /* Pass 'name' and 'setName' to ProfileEditor */
      <ProfileEditor name={/* COMPLETE THIS PROP */} setName={/* COMPLETE THIS PROP */} />
    </div>
  );
}
```

export default App;

2. **ProfileHeader.js** (Displays User's Name and Age)

```
function ProfileHeader({ /* COMPLETE THIS LINE */ }) {
  return (
    <div>
      <h1>Welcome, {name}!</h1>
      <p>Age: {age}</p>
    </div>
  );
}
```

export default ProfileHeader;

3. **ProfileEditor.js** (Allows Name Update via Input Field)

```
function ProfileEditor({ /* COMPLETE THIS PROP */ }) {
  return (
    <div>
      <h3>Edit Your Name</h3>
      <input
        type="text"
        value={name}
        onChange={(e) => /* COMPLETE THIS EVENT HANDLER */}
        placeholder="Enter new name"
      />
    </div>
  );
}
```

export default ProfileEditor;

Tasks to Complete:

- (1) **Initialize name state** in App.js using useState.
- (2) **Pass props correctly** to ProfileHeader and ProfileEditor.
- (3) **Handle state updates** inside the onChange event in ProfileEditor.

Evaluate responses using AI: **OFF**

#### Answer explanation

1. App.js (Manages State and Composition)

```
import { useState } from "react";
import ProfileHeader from "./ProfileHeader";
import ProfileEditor from "./ProfileEditor";

function App() {
  // Initialize state for 'name' with "John Doe"
  const [name, setName] = useState("John Doe");
  return (
    <div style={{ textAlign: "center", padding: "20px" }}>
      { /* Pass 'name' and 'age' as props to ProfileHeader */ }
      <ProfileHeader name={name} age={25} />
      { /* Pass 'name' and 'setName' to ProfileEditor */ }
      <ProfileEditor name={name} setName={setName} />
    </div>
  );
}
```

export default App;

2. ProfileHeader.js (Displays User's Name and Age)

```
function ProfileHeader({ name, age }) {
  return (
    <div>
      <h1>Welcome, {name}!</h1>
      <p>Age: {age}</p>
    </div>
  );
}

export default ProfileHeader;
```

3. ProfileEditor.js (Allows Name Update via Input Field)

```
function ProfileEditor({ name, setName }) {
  return (
    <div>
      <h3>Edit Your Name</h3>
      /* Input updates the 'name' state in App.js */
      <input
        type="text"
        value={name}
        onChange={(e) => setName(e.target.value)}
        placeholder="Enter new name"
      />
    </div>
  );
}

export default ProfileEditor;
```

19. Multiple Choice 30 seconds 1 point AI Edit

What should replace the missing code in the handleChange function to correctly update the state with the input value?

```
import { useState } from "react";
function InputComponent() {
  const [text, setText] = useState("");
  function handleChange(event) {
    setText(_____); // Captures input value
  }
  return (
    <div>
      <input type="text" value={text} onChange={handleChange} />
      <p>You typed: {text}</p>
    </div>
  );
}

export default InputComponent;
```

Answer choices

event.value

event.inputValue

event.currentValue

event.target.value



**Answer explanation**

In React, `event.target.value` **retrieves the latest value entered in an input field.**

It is commonly used inside `onChange` event handlers to update the component's state dynamically.

☐

20. Fill in the Blank

20 seconds

1 point

AI

☐☐

Edit

☐

Fill in the missing code to correctly define a **404 page route** in `App.js`:

```
<Route path="____" element={<h1>404 - Page Not Found</h1>} />
```

**Answer**

\*

**Answer explanation**

The `"**"` wildcard matches **any URL that does not match other defined routes**, ensuring that an **unknown URL** displays a **404 page**.

```
<Route path="*" element={<h1>404 - Page Not Found</h1>} />
```

☐

21. Multiple Choice

30 seconds

1 point

AI

☐☐

Edit

☐

What is the primary purpose of the `useNavigate()` hook in React Router?

**Answer choices**

To create a new route in your application

To navigate programmatically to a different route in your application

To fetch data from a REST API

To protect a route from unauthorized access