Beginner's Guide to React useRef

What is useRef?

useRef is a React Hook that allows you to create a reference to a DOM element or store mutable values that do not trigger a re-render when updated.

Syntax:

const myRef = useRef(initialValue);

`myRef.current` will hold the value or the DOM node.

Why do we use useRef?

- To directly interact with DOM elements (e.g., focusing input fields)

- To persist values across renders without causing re-renders

- To store things like timer IDs, previous values, or any mutable value

Where do we use useRef in real time?

- Auto-focus input fields

- Scroll to a particular section on click

- Play/pause audio or video

- Store timer IDs or previous state values

- Measure dimensions of DOM elements

Example 1: Auto-focus input

```
import React, { useRef, useEffect } from 'react';

export default function App() {
  const inputRef = useRef(null);

  useEffect(() => {
```

```
    inputRef.current.focus();

  }, []);



  return <input ref={inputRef} placeholder='Type here...' />;

}
```

Example 2: Scroll to section

```
import React, { useRef } from 'react';



export default function App() {
  const sectionRef = useRef(null);



  const scrollToSection = () => {
    sectionRef.current.scrollIntoView({ behavior: 'smooth' });
  };



  return (
    <>
      <button onClick={scrollToSection}>Scroll to Section</button>
      <div style={{ height: '800px' }}></div>
      <div ref={sectionRef}>Hello Section </div>
    </>
  );
}
```

Example 3: Store timer ID

```
import React, { useRef } from 'react';



export default function App() {
```

Beginner's Guide to React useRef

```
const timerRef = useRef(null);

const startTimer = () => {
  timerRef.current = setInterval(() => {
    console.log('Tick');
  }, 1000);
};

const stopTimer = () => {
  clearInterval(timerRef.current);
};

return (
  <>
    <button onClick={startTimer}>Start</button>
    <button onClick={stopTimer}>Stop</button>
  </>
);
}
```

Interview Answer Tip

useRef is a hook in React that lets us reference DOM elements directly or store values that persist across renders without triggering a re-render. Its commonly used for focusing inputs, scrolling to elements, or keeping values like timers. For example, we can auto-focus an input using ref.current.focus() in useEffect.