Exam - Part 1: React

React: Question A (45 points)

1. How does React integrate into the MERN stack?
2. Describe the notion of component composition and its benefits in React application development.
3. Analyze the following example and provide an explanation of the purpose of the *useEffect* hook and its appropriate use cases.

```
function Timer() {
  const [count, setCount] = useState(0);

  useEffect(() => {
    let timer = setTimeout(() => {
      setCount((count) => count + 1);
    }, 1000);

    return () => clearTimeout(timer);
  }, []);

  return <h1>The component has rendered {count} times!</h1>;
}
```

4. Describe how this component applies conditional styling based on the *type* prop.

```
function Alert({ type }) {
    const styles = {
        success: { color: 'green' },
        error: { color: 'red' },
    };

    return <div style={styles[type]}>This is a {type} alert.</div>;
}
```

React: Question B (25 points)

Here's the logic for a custom hook designed for a generic input field.

1. What does this hook return?
2. Can we name this hook *myHook*? If not, justify your answer and provide an alternative name(s) for the hook.
3. What happens if, instead of writing **import { useState } from "react"**, we write **import useState from "react"**?

```
import { useState } from "react";

const myHook = (type) => {
  const [value, setValue] = useState("");
  const onChange = (event) => {
    setValue(event.target.value);
  };
  return { type, value, onChange };
};
```

React: Question C (80 points)

The **Signup** component below is for user registration. Refer to the code and:

1. Extract the logic for *signing up* into a custom hook.
2. How can you use the extracted hook in the *Signup* component?
3. Describe the mechanism for displaying an error message when the **error** state is **true**.
4. Modify the button within the **Signup** component so that it is disabled when **isLoading** state is **true**.
5. Explain the purpose of the following line: **body: JSON.stringify({ email, password })**.
6. Justify the difference between using **setIsLoading(true)** and **isLoading=true** when managing the loading state.
7. Discuss the significance of using **e.preventDefault()** in the *handleSubmit* function and what might happen if it were omitted.
8. Explain the difference between using **onChange={(e) => setEmail(e.target.value)}** and **onChange={setEmail(e.target.value)}** when handling input changes.
9. What is the difference between <form className= "signup" onSubmit={handleSubmit}> and <form class= "signup" onsubmit=handleSubmit()>.

```
import { useState } from "react"
import { useAuthContext } from './useAuthContext'
const Signup = () => {
  const [email, setEmail] = useState('')
  const [password, setPassword] = useState('')
  const [error, setError] = useState(null)
  const [isLoading, setIsLoading] = useState(false)
  const { dispatch } = useAuthContext()
```

```
    const signup = async (email, password) => {
      setIsLoading(true)
      setError(null)

      const response = await fetch('/api/user/signup', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify({ email, password })
      })
      const data = await response.json()

      if (!response.ok) {
        setIsLoading(false)
        setError(data.error)
      }
      if (response.ok) {
        localStorage.setItem('user', JSON.stringify(data))
        dispatch({type: 'LOGIN', payload: data})
        setIsLoading(false)
      }
    }

    const handleSubmit = async (e) => {
      e.preventDefault()
      await signup(email, password)
    }

    return (
      <form className="signup" onSubmit={handleSubmit}>
        <h3>Sign Up</h3>
        <label>Email address:</label>
        <input
          type="email"
          onChange={(e) => setEmail(e.target.value)}
          value={email}
        />
        <label>Password:</label>
        <input
          type="password"
          onChange={(e) => setPassword(e.target.value)}
          value={password}
        />
        <button>Sign up</button>
      </form>
    )
  }

  export default Signup
```