JavaScript

1 - Reading the following statements, please indicate which information will be printed on the screen.

The result of calculateBinomial(2, 3): 25
The result of calculateBinomial(4, 5):81
numberOfCalls:2
typeof squareOfA: undefined (because squareOfA is not defined in the global scope.)

React

1 - What are props in a component? Select all the options that apply

- ☑ They are the parameters of a component.
- □ It Is the state of the component.
- ☑ It is a mechanism to put data inside the component.

2 - What is a component? Mark only one option.

- □ An element.
- ☑ A function or class that returns an element.
- □ An element that returns a function or class.
- □ Other:

3 - Which of these component types are valid? Select all the options that apply.

- □ Complete Component.
- ☑ Class Component.
- ☑ Component Function.
- □ Parent Component.
- □ Child Component.

4 - What does the term JSX refer to?

- □ It is the mix between pure HTML and javascript , for this reason JSX is used to identify React syntax.
- ☑ Similar in appearance to XML and HTML, JSX is used to create elements using a familiar syntax.
- □ Term used by React that refers to the mixture of pure XML, HTML and pure JavaScript.
- □ Allows to represent the mixture of XML, which is similar to HTML, and JavaScript.

5 - How do I pass a value from a parent component to a child component? Select all the options that apply.

- ☑ Using props.
- □ Through the use of states.
- □ By only destructuring one or more objects.

- ☑ Referring the child component to the values of the parent component.
- □ Using values from the global application context usage.

6 - Created the following state of a variable:

```
   import { useState } from "react";
         const [value, setvalue] = useState("Value")
```

Which of these implementations change the state of a component? Check only one option.

- □ state.value = “my value”
- □ setValue(“my value”)
- ☑ setState({ value: “my value” })
- □ value = { value: “my value” }
- □ setValue({“value”: “my value”})

7 - How many and which arguments does useEffect Hook receive?

- □ An argument to provoke the effect.
- □ Three arguments, which are the dependencies, the states of the variables and the effect.
- ☑ Two arguments, which are the states of the variables and the dependencies.
- □ An argument that receives the state of the variables.
- □ Two arguments, which are the effect and the dependencies.

8 - How do you import Javascript modules installed with NPM or YARN? Check only one option.

- □ import name from 'node_modules/name';
- □ import name from '../../node_modules/name/name.js';
- □ import name from '../../node_modules/name/index.js';
- ☑ import name from 'name';
- □ Others:

9 - What functionality does the use of Context API provide?

- ☑ It is a mechanism that provides a way to pass data through the component tree without manually passing props at all levels.
- □ It is a functionality that facilitates the handling of data from parent components to child components.
- □ It is a React modality that replaces the use of props in its entirety to be able to consume variable values no matter where they are located, facilitating data management throughout the application.

10 - How do I import the following component, considering that it is inside a file in the same directory, called MyComponent.js?

```
export const MyComponent = ( ) =>{
      return(
        <h1> Hello World </h1>
```

```
    )
  }
```

Mark only one option.

- □ Import MyComponent from './MyComponent';
- □ import { MyComponent } from 'MyComponent';
- □ import MyComponent from 'MyComponent.js';
- ☑ import { MyComponent } from './MyComponent.js';
- □ Others:

11 - What is the difference between class components and functional components?

Class components come with a built-in lifecycle and can manage state, while functional components are simpler and use React Hooks for lifecycle-like functionality and local state management.

12 - How would you set the state of a component?

You can set a component's state using the useState() hook. Once you've used the useState() hook, you can access and update the state variables using the values returned by the hook.

Example:

```
import { useState } from "react";
const [count, setCount] = useState(0);
setCount(1);
```

13 - What happens when we change the state of a component?

When you change a component's state, it automatically updates the user interface to match the new state using a virtual DOM.

14 - Write a component that renders a list of elements, given a prop that contains an array of objects in the following form: { firstname: "demo", lastname: "demo", dni: 1234 }

(This prompt was generated with the assistance of AI.)

```
  import React from 'react';

  const ListComponent = ({ data }) => {
    return (
      <ul>
        {data.map((item, index) => (
          <li key={index}>
            {item.firstname} {item.lastname} (DNI: {item.dni})
          </li>
        ))}
      </ul>
    );
  };

  export default ListComponent;
```

15 - Write a component that receives a prop called "data", and renders an H1 with the content of data.

```
const DataDisplay = ({ data }) => {
  return (
    <h1>{data}</h1>
  );
};
```

16- What is the error in the following component?

```
export const Button = ({ text }) => {
  <div className="button">
    <h1 className="text-button">{text}</h1>
  </div>;
```

The error in the component is that it’s missing a return statement.

17 - Explain, in your own words, what this component does.

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

export const Componente = () => {
  const [state, setState] = useState(1);

  const click = (propertyValue: number) => {
    setState(propertyValue + 1);
  };
  return (
    <>
      <h1>{ state } </h1>
      <button onClick={() => click(state)}> Click here </button>
  </>
      );
};
```

When you click the "Click here" button, the number increases by 1.

18 - How would you make a Rest API call from JavaScript?

Using the Fetch API:

```
const fetchData = async () => {
  const response = await fetch('https://example.com/api/users');
  const data = await response.json();

  console.log(data);
};

fetchData();
```

19 - What is the use of the useEffect hook?

The `useEffect` hook is for handling side effects like making API requests or changing the DOM. It runs after a component renders. You can also use it to clean up when the component is removed.

20 - How would you develop a custom hook that exposes methods to save and modify a counter? The hook must have a parameter that allows the developer to specify how much the counter should add for each call to the setter.

(This prompt was generated with the assistance of AI.)

Here is a hook that exposes methods to save and modify a counter, with a parameter that allows the developer to specify how much the counter should add for each call to the setter.

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

export function useCounter(initialCount, step = 1) {
  const [count, setCount] = useState(initialCount);

  function increment() {
    setCount((prevCount) => prevCount + step);
  }

  function decrement() {
    setCount((prevCount) => prevCount - step);
  }

  function reset() {
    setCount(initialCount);
  }

  return {
    count,
    increment,
    decrement,
    reset,
  };
}
```