Module quiz: Data and state

1. In React, data flows in one way: from a parent component to a child component. **1 / 1 ponto**

● True

○ False

Correto
That's correct. The above statement is true.

2. Why is one-way data flow important in React? **1 / 1 ponto**

● It ensures that the data is flowing from top to bottom in the component hierarchy.

○ It ensures that the data is flowing from bottom to top in the component hierarchy.

Correto
Correct. This ensures that the data is flowing from top to bottom in the component hierarchy.

3. True or false? State data is the data inside a component that a component can mutate. **1 / 1 ponto**

● True

○ False

Correto
Correct. State data is data inside a component that a component, which that component controls and can mutate.

4. What is prop drilling? **1 / 1 ponto**

● Prop drilling is a situation where you are passing data from a parent to a child component, then to a grandchild component, and so on, until it reaches a more distant component further down the component tree, where this data is required

○ Prop drilling is a situation where you are passing data from a child, to a parent component, then to a grandparent component, and so on, until it reaches a more distant component further up the component tree, where this data is required.

Correto
Correct. Prop drilling is a situation where you are passing data from a parent to a child component, then to a grandchild component, and so on, until it reaches a more distant component further down the component tree, where this data is required

5. The distinction between stateful and stateless components is that the latter doesn't have its own state. **1 / 1 ponto**

● True

○ False

Correto
That's correct! This statement is true.

6. Choose the correct statement. **1 / 1 ponto**

○ Remember that you should always change the values of props in children components; you should never work with them as they are. In other words, props are mutable.

● Remember that you should never change the values of props in children components; you should only work with them as they are. In other words, props are immutable.

Correto
That's correct! Props are immutable and thus you should not attempt to update them in children components.

7. Is this code valid? **1 / 1 ponto**

```
function App() {
    const handler = () => console.log('fourth example')
    return (
  return (
    <div>
      <button onClick = {handler} >
        Click Me!
      </button>
    </div>
  )
}
export default App
```

● Yes

○ No

✓ Correto
Correct! This code is an example of a valid onClick event handler.

8. Is this code valid? **1 / 1 ponto**

```
<button onClick={ () => console.log('clicked') }>
  Click me
</button>
```

● Yes

○ No

✓ Correto
Correct! This code is an example of a valid onClick event handler.

9. Select the correct statement: The useState hook... **1 / 1 ponto**

● ... lets you hook into React state and lifecycle features from a component.

○ ...is not part of React; you must import it from a third-party package.

○ ... has a convention that if the state variable is named, for example, *counter*, the function to update this counter variable should be named *counterFunction*.

○ ... should never be called at the top level of a React component.

✓ Correto
Correct.

10. The Context API allows you to: **1 / 1 ponto**

● Avoid having to pass state down through multiple levels of components.

○ Avoid having to use the return statement in a child component.

○ Avoid having to keep your components modular.

✓ Correto
Correct. Using Context API allows you to bypass having to pass state down through multiple levels of components.