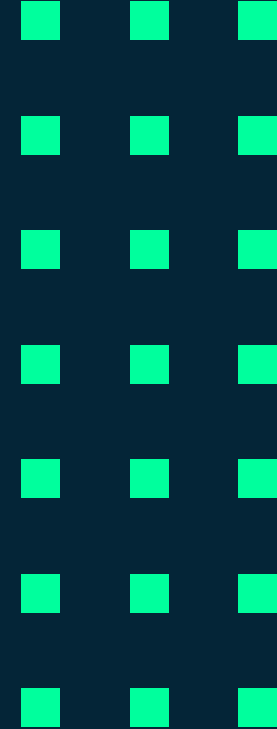# HOW CONTEXT WORK IN REACT.JS
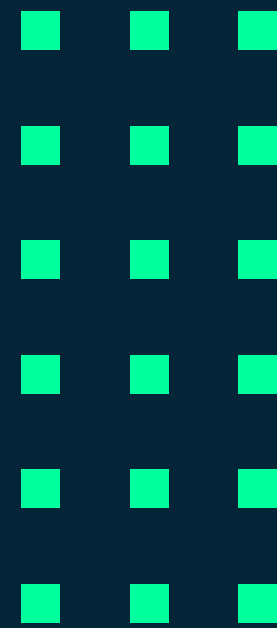
ASHWIN MALI

WHAT IS CONTEXT?

React Context provides a way to pass data through the component tree from parent to child components, without having to pass props down manually at each level.

Each component in Context is context-aware. Essentially, instead of passing props down through every single component on the tree, the components in need of a prop can simply ask for it, without needing intermediary helper components that only help relay the prop.

These Two Code Lines are Important to Create the Context

```javascript
const MyContext = createContext();
return <MyContext.Provider value={sharedData}>{children}</MyContext.Provider>;
```

ASHWIN MALI

useContext

useContext is used in React to consume values from a React context within a functional component. It provides a more concise and readable way to access context values compared to using the Consumer component

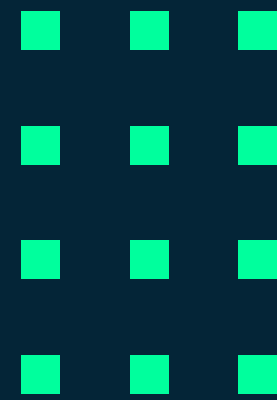Importing and Initializing of useContext Can be following :

```
1   import React, { createContext, useContext } from 'react';
2   const sharedData = useContext(MyContext);
```

Use Case

```jsx
import React, { createContext, useContext } from 'react';

const MyContext = createContext();

const MyProvider = ({ children }) => {
  const sharedData = 'This is shared data';

  return <MyContext.Provider value={sharedData}>{children}</MyContext.Provider>;
};

const ComponentA = () => {
  const sharedData = useContext(MyContext);

  return (
    <div>
      <p>Component A</p>
      <p>Shared Data: {sharedData}</p>
    </div>
  );
};

const App = () => {
  return (
    <MyProvider>
      <ComponentA />
    </MyProvider>
  );
};

export default App;
```

ASHWIN MALI

FOLLOW

ASHWIN MALI

Approach to BCA | HTML5 | CSS | JavaScript | React.js | Java | C++ 🚀

DON'T FORGET TO SAVE