**Explain life cycle in class Component and function component with hooks**

**In React, a component is a reusable piece of code that represents a part of the user interface. Every React component has a life cycle, which refers to the various stages of a component's existence from creation to destruction. React class components and function components with hooks have different life cycle methods.**

**Class Component Life Cycle Methods:**

**a. Mounting: This stage is when the component is first created and mounted into the DOM. The following methods are called in the following order:**

**constructor(): This method is called first and is used to initialize state and bind methods.**

**static getDerivedStateFromProps(): This method is called after the constructor and before the render method. It is used to update the state based on the props.**

**render(): This method is called next and returns the JSX code that will be displayed in the DOM.**

**componentDidMount(): This method is called after the component is mounted into the DOM. It is used to perform any necessary setup, such as fetching data from an API or adding event listeners.**

**b. Updating: This stage is when the component is updated due to changes in props or state. The following methods are called in the following order:**

**static getDerivedStateFromProps(): This method is called first and is used to update the state based on the props.**

**shouldComponentUpdate(): This method is called next and is used to determine if the component should be updated or not. It returns a boolean value.**

**render(): This method is called next and returns the updated JSX code.**

**componentDidUpdate(): This method is called after the component is updated in the DOM. It is used to perform any necessary cleanup, such as removing event listeners.**

**c. Unmounting: This stage is when the component is removed from the DOM. The following method is called:**

**componentWillUnmount(): This method is called before the component is unmounted from the DOM. It is used to perform any necessary cleanup, such as removing event listeners or clearing intervals or timeouts.**

**Function Component with Hooks Life Cycle Methods:**

**Function components with hooks are a newer way of writing components in React. They have a different set of life cycle methods based on the hooks used.**

**a. Mounting: This stage is when the component is first created and mounted into the DOM. The following hooks are called in the following order:**

**useState(): This hook is used to initialize state.**

**useEffect(): This hook is called after the component is mounted into the DOM. It is used to perform any necessary setup, such as fetching data from an API or adding event listeners.**

**b. Updating: This stage is when the component is updated due to changes in props or state. The following hooks are called in the following order:**

**useState(): This hook is used to update state.**

**useEffect(): This hook is called after the component is updated in the DOM. It is used to perform any necessary cleanup, such as removing event listeners.**

**c. Unmounting: This stage is when the component is removed from the DOM. The following hook is called:**

**useEffect(): This hook is called before the component is unmounted from the DOM. It is used to perform any necessary cleanup, such as removing event listeners or clearing intervals or timeouts.**

**In summary, the life cycle of a React component is a set of methods that are called at different stages of the component's existence. Class components and function components with hooks have different life cycle methods, but they serve the same purpose of allowing developers to manage the behavior of their components at different stages.**