Question 1: What is Redux, and why is it used in React applications? Explain the core concepts of actions, reducers, and the store.

What is Redux?

Redux is a **state management library** for JavaScript applications, commonly used with React. It provides a centralized **global state** that makes state management more predictable and scalable, especially in large applications.

Why is Redux used in React?

- **Centralized State Management** – Stores the application state in a **single source of truth (store)**.
- **Predictability** – State changes occur only through **pure functions (reducers)**.
- **Easier Debugging** – With Redux DevTools, developers can **track state changes**.
- **Efficient Data Sharing** – Components can access state **without prop drilling**.

Core Concepts of Redux

1. **Store**
    - The **store** holds the entire application state.
    - It is created using `createStore()` (or `configureStore()` in Redux Toolkit).
    - Example:

```
import { createStore } from 'redux';
import rootReducer from './reducers';

const store = createStore(rootReducer);
```

Actions

- **Actions** are objects that describe what **changes should happen** in the store.
- They must have a **type** property and can carry extra **payload**.
- Example:

```
const increment = () => {
  return {
    type: 'INCREMENT'
  };
};
```

Reducers

- **Reducers** are **pure functions** that determine **how the state changes** in response to an action.
- They take **current state** and **action** as arguments and return **new state**.
- Example:

```
const counterReducer = (state = 0, action) => {
  switch (action.type) {
    case 'INCREMENT':
      return state + 1;
    case 'DECREMENT':
      return state - 1;
    default:
      return state;
  }
};
```

Dispatch

- The `dispatch()` method **sends actions to the reducer**.
- Example:

```
store.dispatch(increment());
```

How Redux Works in React

Install Redux & React-Redux

```
npm install redux react-redux
```

Create a store and provide it to the app

```
import { createStore } from 'redux';
import { Provider } from 'react-redux';
import counterReducer from './reducers/counterReducer';
import App from './App';

const store = createStore(counterReducer);

<Provider store={store}>
```

```
  <App />
</Provider>
```

Use `useSelector` and `useDispatch` in Components

```
import { useSelector, useDispatch } from 'react-redux';

function Counter() {
  const count = useSelector(state => state);
  const dispatch = useDispatch();

  return (
    <div>
      <h1>{count}</h1>
      <button onClick={() => dispatch({ type: 'INCREMENT'
})}>+</button>
      <button onClick={() => dispatch({ type: 'DECREMENT'
})}>-</button>
    </div>
  );

}
```