Question 1. What is Redux?

Answer.

Redux is a predictable state management library used to manage global state in React applications.

It follows:

- Single source of truth

- State is read-only

- Changes are made using pure functions (reducers)

Redux is useful when:

- Multiple components need same data

- App has complex state logic

- Debugging state changes is important

Question 2. Redux Core Concepts

Answer.

1. Store

The central place where the entire application state is stored.

2. Action

A plain JavaScript object describing what happened.

Example:

```
{
```

```
  type: "INCREMENT",

  payload: 5

}
```

3. Reducer

A pure function that takes:

- Current state
- Action

And returns:

- New state

4. Dispatch

  A function used to send actions to the reducer.

Question 3. Redux Flow

Answer.

  Component → dispatch(action) → Reducer → Store → Component re-renders

This one-way flow makes Redux predictable.

Question 4. Combining Multiple Reducers

Answer.

    Combining Multiple ReducersLarge applications use multiple reducers:

- authReducer
- counterReducer
- resumeReducer

Redux combines them into a single store.

Question 5. Middleware

Answer.

    Middleware allows:

- Async API calls
- Logging
- Error handling

Example: Redux Thunk

Question 6. Redux DevTools

Answer.

    Redux DevTools helps:

- View state changes
- Inspect dispatched actions

- Debug easily

- Time-travel debugging