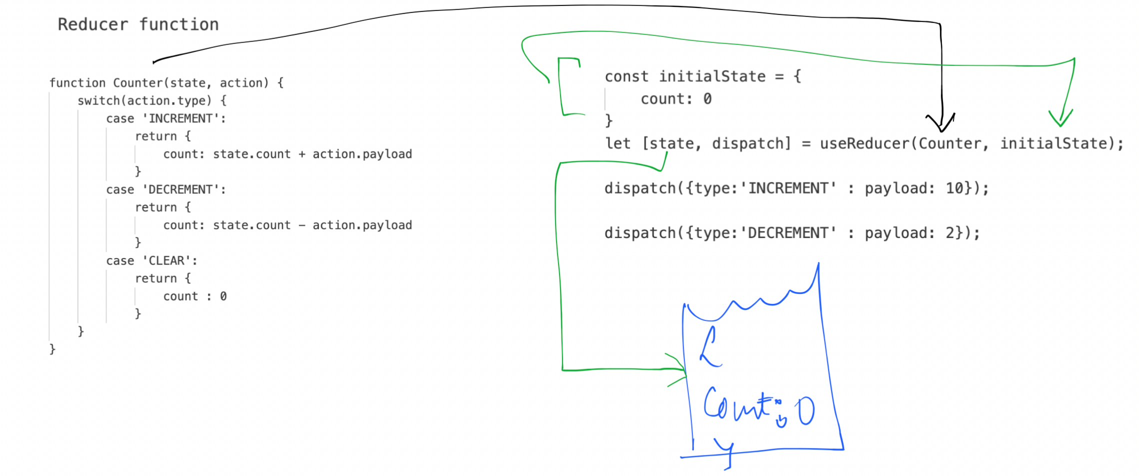
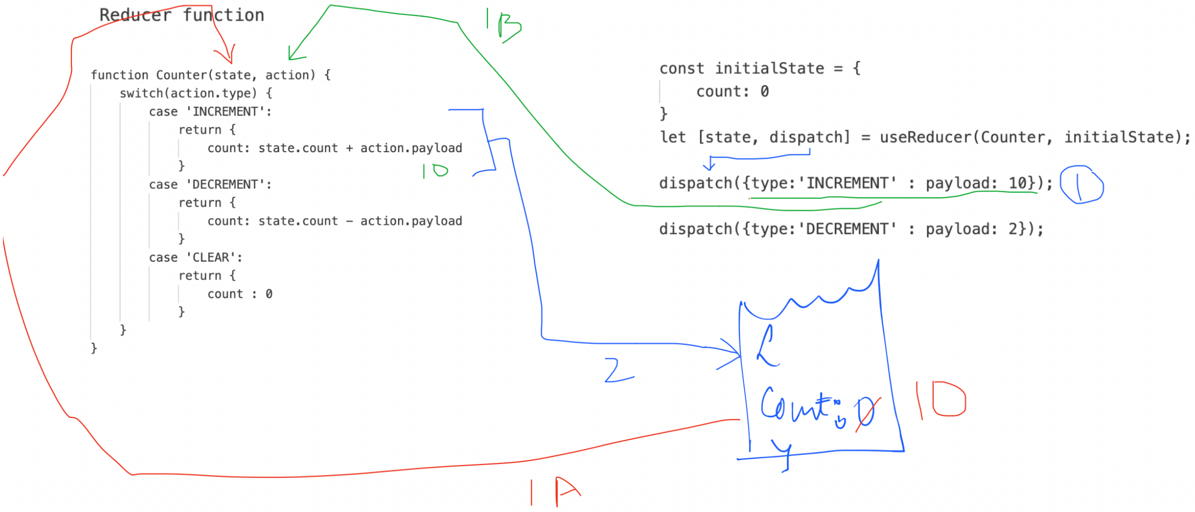


Reducer function

```
function Counter(state, action) {
    switch(action.type) {
        case 'INCREMENT':
            return {
                count: state.count + action.payload
            }
        case 'DECREMENT':
            return {
                count: state.count - action.payload
            }
        case 'CLEAR':
            return {
                count : 0
            }
    }
}
```

```
const initialState = {
    count: 0
}
let [state, dispatch] = useReducer(Counter, initialState);

dispatch({type:'INCREMENT' : payload: 10});

dispatch({type:'DECREMENT' : payload: 2});
```

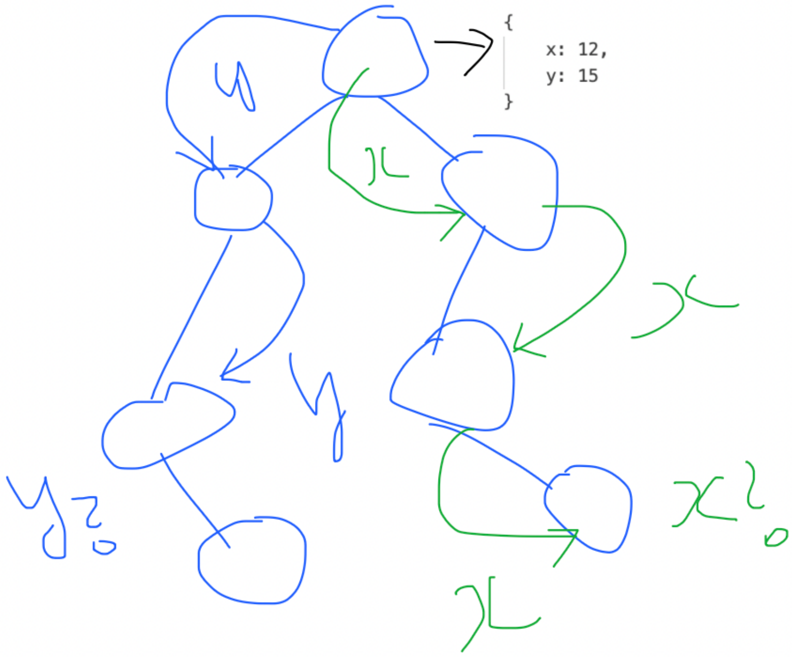


Reducer function

```
function Counter(state, action) {
    switch(action.type) {
        case 'INCREMENT':
            return {
                count: state.count + action.payload
            }
        case 'DECREMENT':
            return {
                count: state.count - action.payload
            }
        case 'CLEAR':
            return {
                count : 0
            }
    }
}
```

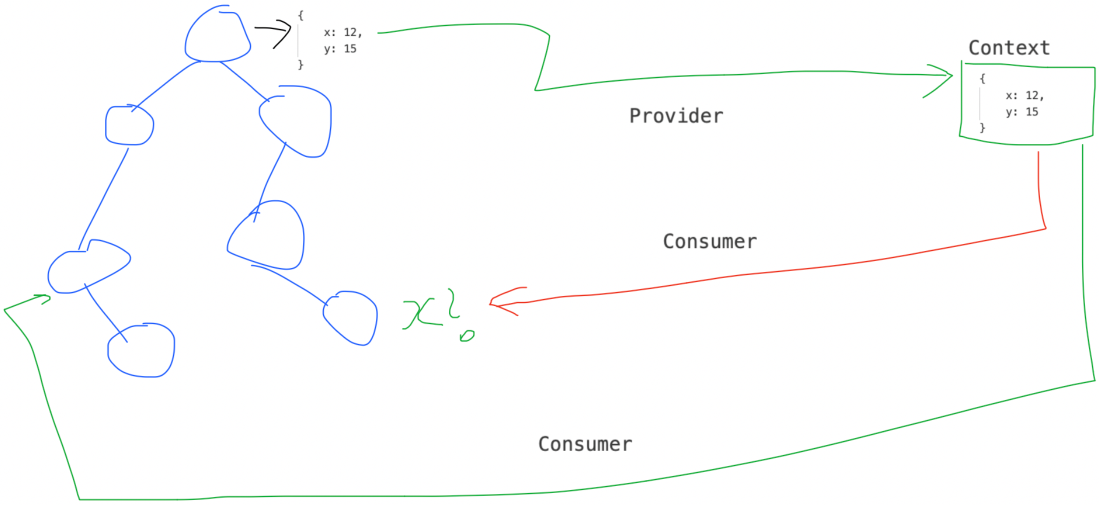
```
const initialState = {
    count: 0
}
let [state, dispatch] = useReducer(Counter, initialState);

dispatch({type:'INCREMENT' : payload: 10});

dispatch({type:'DECREMENT' : payload: 2});
```

```
<div>
Count: {state.count}
</div>
```



```
{
    x: 12,
    y: 15
}
```

props drill



```
{
    x: 12,
    y: 15
}
```

Context

```
{
    x: 12,
    y: 15
}
```

Provider

Consumer

Consumer