

```
const initialState = { value: 0 }

const counterSlice = createSlice({
  name: 'counter',
  initialState,
  reducers: {
    increment: (state) => state.value++,
    decrement: (state) => state.value--,
    incrementByAmount: (state, action: PayloadAction) => {
      state.value += action.payload
    },
  },
})

export const { increment, decrement, incrementByAmount } = counterSlice.actions
export default counterSlice.reducer
```

*Handwritten annotations:*

- reducer**: Points to the `reducers` object in the `createSlice` call.
- View**: Points to the `increment()` and `incrementByAmount(10)` calls.
- Action**: Points to the `incrementByAmount` reducer.

```
const initialState = { value: 0 }

const counterSlice = createSlice({
  name: 'counter',
  initialState,
  reducers: {
    increment: (state) => state.value++,
    decrement: (state) => state.value--,
    incrementByAmount: (state, action: PayloadAction) => {
      state.value += action.payload
    },
  },
})

export const { increment, decrement, incrementByAmount } = counterSlice.actions
export default counterSlice.reducer
```

*Handwritten annotations:*

- Clone**: Points to the `initialState` property.
- Store**: Points to the `reducers` object.
- State**: Points to the `incrementByAmount` reducer.

```
const initialState = { value: 0 }

const counterSlice = createSlice({
  name: 'counter',
  initialState,
  reducers: {
    increment: (state) => state.value++,
    decrement: (state) => state.value--,
    incrementByAmount: (state, action: PayloadAction) => {
      state.value += action.payload
    },
  },
})

export const { increment, decrement, incrementByAmount } = counterSlice.actions
export default counterSlice.reducer
```

*Handwritten annotations:*

- combineReducers**: Points to the `reducers` object.
- Root Reducers**: Points to the `reducers` object.
- Store**: Points to the `reducers` object.

