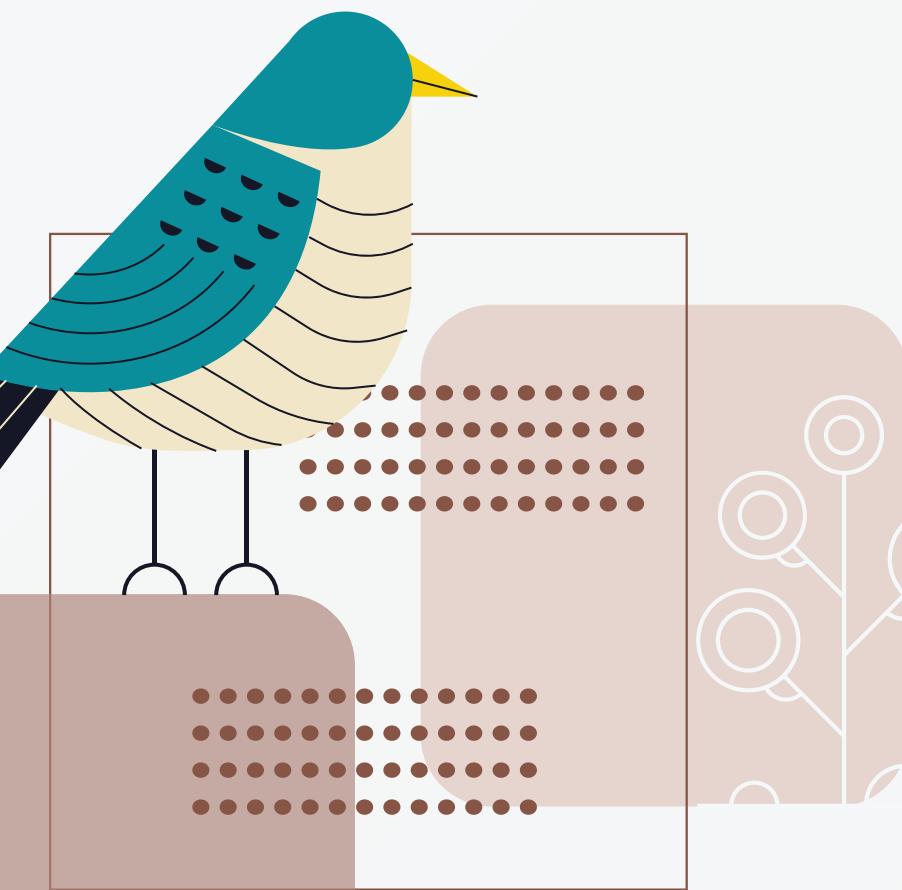


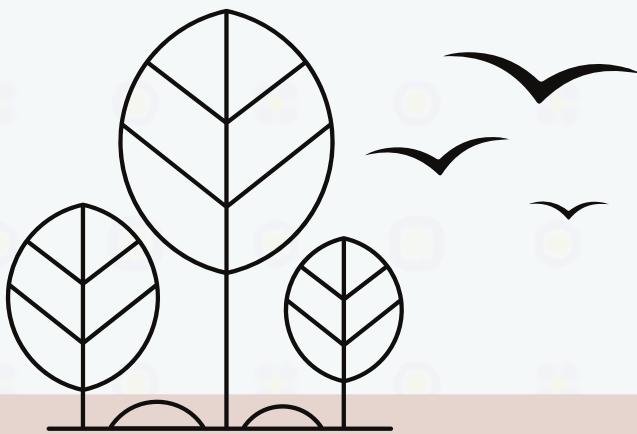
REDUX TOOLKIT

Presented by: hema vardhan



CREATION FLOW OF REDUX

1. import redux: const redux = require('redux')
2. initialize create function: const createStore = redux.createStore
3. create a State object for your application : const initialState = {
stateVariable1,
stateVariable2,...}
4. create reducer function as mentioned in part 1
5. create store using create function : const store = createStore(reducer)
6. to update states dispatch action using action creator:store.dispatch(actionCreator)
7. to get the state use: store.getState()

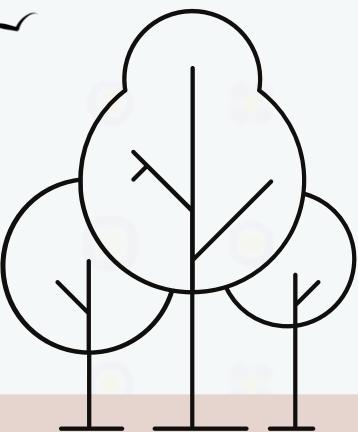


BIND ACTIONS

not useful now-a-days

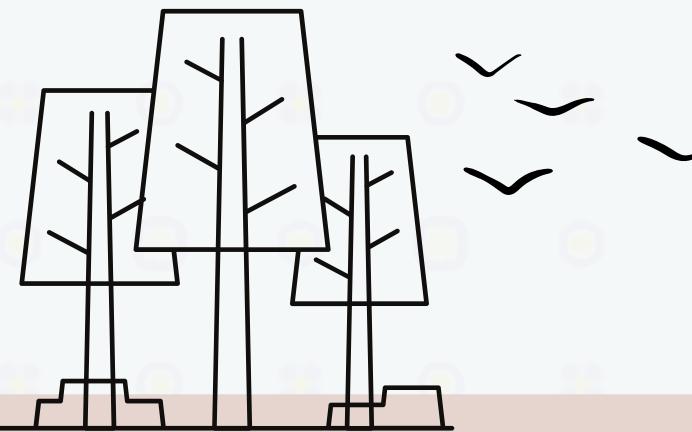
helper function in redux:

1. initialize function : const bindActionCreators = redux.bindActionCreators
2. instead of sending multiple dispatch functions we can just for multiple actions we can use: const actions = bindActionCreators({action1,action2,...},functionToBind)
eg: const actions = bindActionCreators({orderCAke,restockCake},store.dispatch)
3. now to dispatch instead of using store.dispatch(ctionCreator) use:
actions.orderCake()
actions.orderCake()
actions.restockCake()



ICECREAM & CAKE

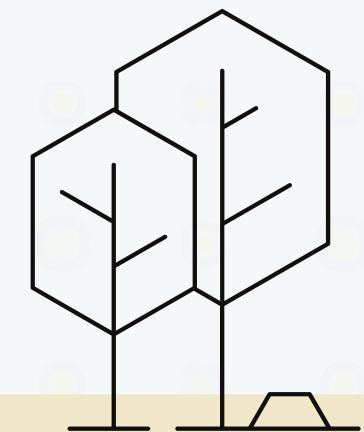
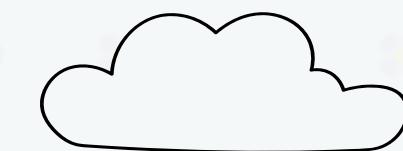
problem Statement: in a shop there are cakes and icecreams:
iceCreams are kept in freezer and cakes in shelf
we have shopkeeper to manage both but to make more scalable we use 2 shopkeepers
one shopkeeper to sell icecreams and other cakes
shopKeeper 1 handles CAKE_ORDER from customer
shopKeeper 2 handles ICECREAM_ORDER from customer
now state has to track two variables
1.cakes in shelf 2.icecreams in freezer
diffrent shopkeepers for diffrent variables help in breaking down problem



DISCUSSION

possibilities are:

1. create one state where all states lie and one reducer where all cases of actions are mentioned
2. create diffrent states to store diff states seperately and create reducers separately
3. create one state and diffrent reducers
best way is to seperate all states and reducer but createStore only takes one argument so to create multiple reducers and one store we use “combineReducers”



COMBINE REDUCERS

1. initialize function: `const combineReducers = redux.combineReducers`

2. before creating store we call this function:

```
const rootReducer = combineReducers({
```

```
    cake: cakeReducer,
```

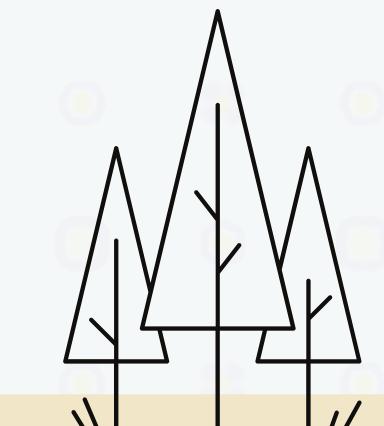
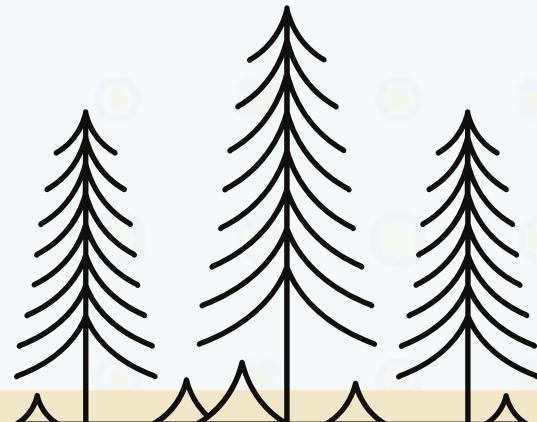
```
    iceCream: iceCreamReducer
```

}by doing this the store stores the states as values with given key names in
rootReducers

this function accepts an object as arg

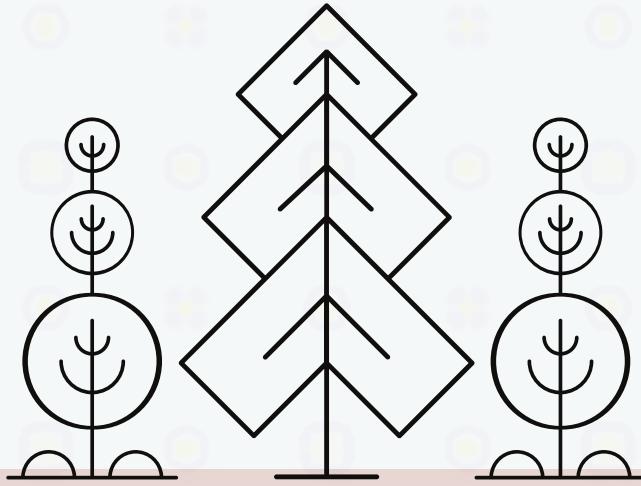
we give a unique key and value as reducers

finally give `const store = createStore(rootReducer)`





THANK YOU



HemaVardhan|BROCHILL |2024 may