

Ushin



React

المكتبة الجاوا سكريبت لبناء الواجهات

المكتبة الجاوا سكريبت لبناء الواجهات

Date

No

Arrow Function → No more issues with this keyword

```
function myfun() {  
  //  
}
```

Arrow

```
const myfun = () => {  
  //  
}
```

if there is return

```
const mult = (number) => {  
  return number * 2;  
}
```

```
const mult = number => number * 2;
```

React

- * React is javascript library for building user interfaces
- * React simplifies building complex interfaces.
- * is all about components

Component in React is just a javascript function

React Core Syntax and JSX → allows you to write HTML in your JS code

adding Events in React

Date / / No

`<button onClick = { console.log ('clicked!!!') } > title </button>`

↓
دو اکت کلیک ہوتا ہے اور اسے دکھاتا ہے

Example

```
const clickHandler = () => {  
  console.log ('clicked');  
}
```

`<button onClick = { clickHandler } > title </button>`

update () is

State

`import React, { useState } from 'react';`

`useState();`

Functions allows us to define values as state when it change it appear on the calling function.

React Hook

↓
calling `useState();` in the component function

`useState (props.title);`

Ex

1) `const [title, setTitle] = useState (props.title);`

2) `const clickHandler = () => {
 setTitle ('updated');
}`

3) use title in HTML → JSX

You can update state by `setInterval ()`
every minute for example





Date / / No

Two way binding → `setEnteredTitle()` `setTitle()`

Child to Parent

Bottom to up

Styled Component

بعض الميزات في دس
و بعض الميزات في دس
في دس

npm install --save styled-components

import styled from 'styled-components'

const Button = styled.button

رابط

after
before

`$: hover`
`$: active`

CSS Module StyleSheet

mdn

① import `styles` from './Button.css';

or classes

② Rename the CSS file → `button.module.css`

في بعض الحالات class لا يعمل في بعض المتصفحات

Errors

You should solve problems with your self, it will make you a better developer.

You need to stand and fix them in your own.

Break Point in Chrome

نقطة كسر

نقطة كسر



Date / / No



React Developer Tools extension

React Developer Tools extension components → tools → [77] installed

JSX limitations You can't return more than one "root" JSX element (You can't also store more than one root JSX element in a variable)

import { Fragment } from 'react';
`<Fragment> </Fragment>`
`<? </?>`

`<React.Fragment> </React.Fragment>`

Fragment allow you to write cleaner code without unnecessary HTML code

div is good but it isn't good practice and efficiency for Screen Reader

React Portals

[ref] → reference

* Side effect \rightarrow Render UI and React to user input
 \hookrightarrow e.g. like \rightarrow Store data in Browser, Send HTTP

`useEffect (() \Rightarrow { }, [dependencies]);`

Function should be executed
 After every component evaluation
 if the specified dependencies
 changed

if the dependencies change
 the function will only run

clean up function \rightarrow `return () \Rightarrow { };`

`useReducer()` For Management

if you have a lot of `useState()`
 \hookrightarrow replacement for `useState()`, if you need Powerful State
 management.

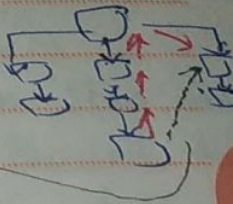
`const [state, dispatchFn] = useReducer (reducerFn, initialState, initFn);`

\downarrow
`(prevState, action) \Rightarrow newState`

useContext Hook

React context is not optimized for high frequency changes.
↳ shouldn't be used to replace all components communicating and props.

- * Context: lets us pass value deep into the component tree
- * So if you want to avoid passing some props through many levels, component composition is often a simpler solution than Context



How does React work?

- * React determine how the component tree currently looks like and what it should look like.
- * React Diff receives the differences and then manipulates the Real DOM

Component

- * re-evaluated whenever props, state, change
- * React executes component function

Real DOM

- * only updated in the place which need to change based on the difference react drive between previous state and current state
- * changed Rarely only when needed and that's important for performance.
- * Reach to Real DOM is very expensive

Prevent unnecessary Re-Evaluations With React.memo (hook) =

memo → tell React that for this component which is an argument ↓ React should look for the props of this component and check new value for the props and compare it with the previous one and only if it change, the component should be re-evaluated.

Prevent Function Re-Creation With useCallback (hook);

useCallback (c) → hook that allows us to store function across component execution, to tell React that i want to save function and this function shouldn't be created with every execution.

useMemo()

Hook

allows you to store any kind of data

import Component from React
Date / / No

Functional Component

```
function Product (props) {  
  // ...  
}
```

Class-based Component

```
class Product extends Component {  
  render() {  
    // ...  
  }  
}
```

Class-based Components Can't use React Hooks

ComponentDidMount() → called once component mounted - use Effect (·, [·])
 ~ ~ update() → ~ ~ ~ updated - ~ ~ (·, [value])
 ~ ~ WillUnmount() → ~ ~ right before it is unmounted
 ~ ~ use Effect (·) → { return () → {·} }, [·]

try and catch you can use them in JavaScript
 but now we are at React → **JSX**

Error Boundary → you should make them in class not function

```
import { Component } from 'react';  
class ErrorBoundary extends Component {  
  constructor() { super(); this.state = { hasError: false }; }  
  componentDidCatch (error) { console.log(error); this.setState({ hasError: true }); }  
  render() { if (this.state.hasError) {  
    return <P> Something went wrong </P>  
  }  
  return this.props.children;  
}
```

<ErrorBoundary>

</ErrorBoundary>

Call to import HTML to where
 +
 +

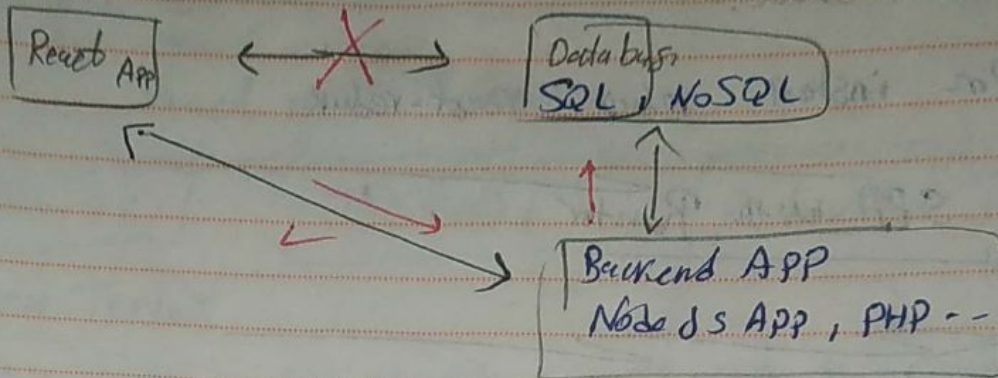
Section 14

Sending HTTP Request

Date: / / No



Browser Side APPS don't directly talk to databases



API

firebase ~~behaves~~ ^{allows} us to use database if ~~we~~ ^{you} backend can use it without writing any code.

Section 15

Custom Hooks

Section 16

Forms & user Input

Section 18

انہ (فہامات) تہمہ و لہ آتہ و لہ بالذہ و لہ بالہ
قالہ اب خلیفہ لہ العزیز لہ العزیز

Date: / / No: 77

Redux: a state management system for cross-component or app wide state.

npm install redux react-redux

Section 20) SPA with Reactor

Section 21

Section 22

Authentication is needed if content should be protected not accessible by everyone.

Section 23

Next JS → Frame another React

*** It is React Framework for Production ***
Full Stack Framework for React JS

① cmd \xrightarrow{cd} npx create-next-app

↳ name

→ open vsc and open the folder

→ npm install

npm run dev

next js

import Link from "next/link"; → `<Link href = "/ /">`
to → React

Section 24

Animating React

Date / / No

Libraries → react transition group → react library
→ react motion
→ react move
→ react route transition

Section 25

Testing

Manual Testing

write code → preview → test in browser
↓
hard to test all possible combinations

Automated Testing

code that tests your code
↓
very technical but allows
you to test all building blocks
at once

unit Test

Integration Test

End to End Test (etc)

the most common

test individual blocks

Test combination multiple
of blocks

Test complete scenarios

make new file → name, test.js

↓ inside the file

import name of file to test from ;

test ('description of test', () => {

· render (<name of file to test />);

expect (), to BeInTheDocument();

}

in console use

npm test

→ run the test

✓ Jest → Delightful JS testing

✓ React testing library

← React hooks testing library

Session 27

Type Script + React

Date...../...../..... No.....

typescript is superset to javascript

number
string
boolean

Variable

add static typing to js

JS is dynamically typed

npm install typescript
npx tsc

```
let age: number;  
let isMar: boolean boolean;  
let hobbies: string[];  
hobbies: ['sport', ...];
```

Type Aliases

```
type Person = {  
  name: string;  
  age: number;  
};
```

```
let Person: Person;  
let People: Person[];
```