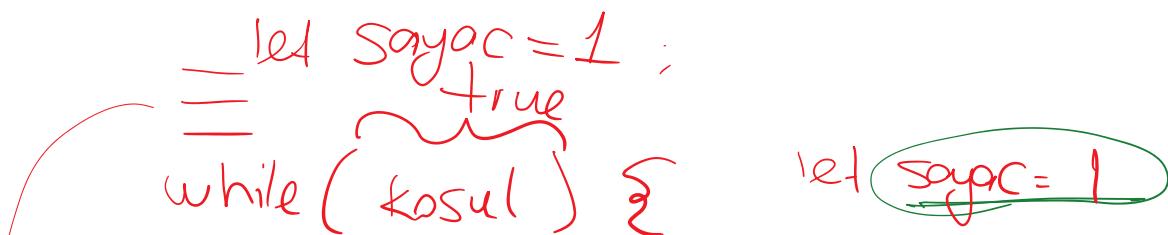
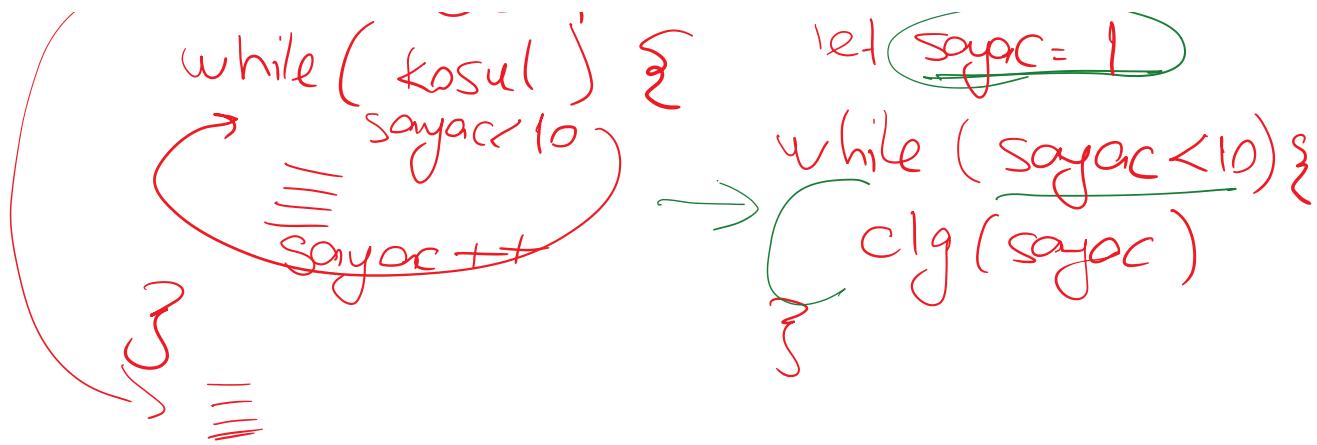


Carpim x 5





- 1.) hak = 5
- 2.) Rasgele sayı - (at (0-100) (rasgele)) 50
- 3.) Ibranik iden tahminini iste (-tahmin) → 25 7 ≠ 50
- 4.) hak = hak - 1 → 4 ≠ 2
- 5.) eger (-tahmin == rasgele)
 - Print (Tebrikler bi diniz) → bildiniz
 - degilse (-tahmin < rasgele)
 - Print (Arttir) → arttir
 - degilse
 - print (Azalt) → azalt
- 6.) eger hak > 0 goto 3
 - degilse print (Uzgunuz)

FUNCTIONS

20 Ağustos 2022 Cumartesi 10:22

~~hesapla(-)~~ funct ism:
const hesapla = function (n1, n2) {
 return n1 + n2
}
→ hesapla (3, 5)

const $\text{topla}(a \leftarrow (2, 3) \Rightarrow a + b)$

$\text{clg}(\text{topla}(2, 3)) \rightarrow 5$

const $\text{usAl} = (t, u) \Rightarrow t^{**} u$
 $\text{usAl}(2, 3) \quad 2^3 = 8$

const $\text{yazdir} = () \Rightarrow \text{clg}("Hi")$

const $\text{Sebm} = (x) \Rightarrow \text{clg}(x);$

const $\text{hesapla} = (x, y, islem) \Rightarrow \{$
 let sonuc
 $\text{if } (islem == '-' || '+') \}$
 $\text{else if } ($
 else
 $\text{return sonuc};$

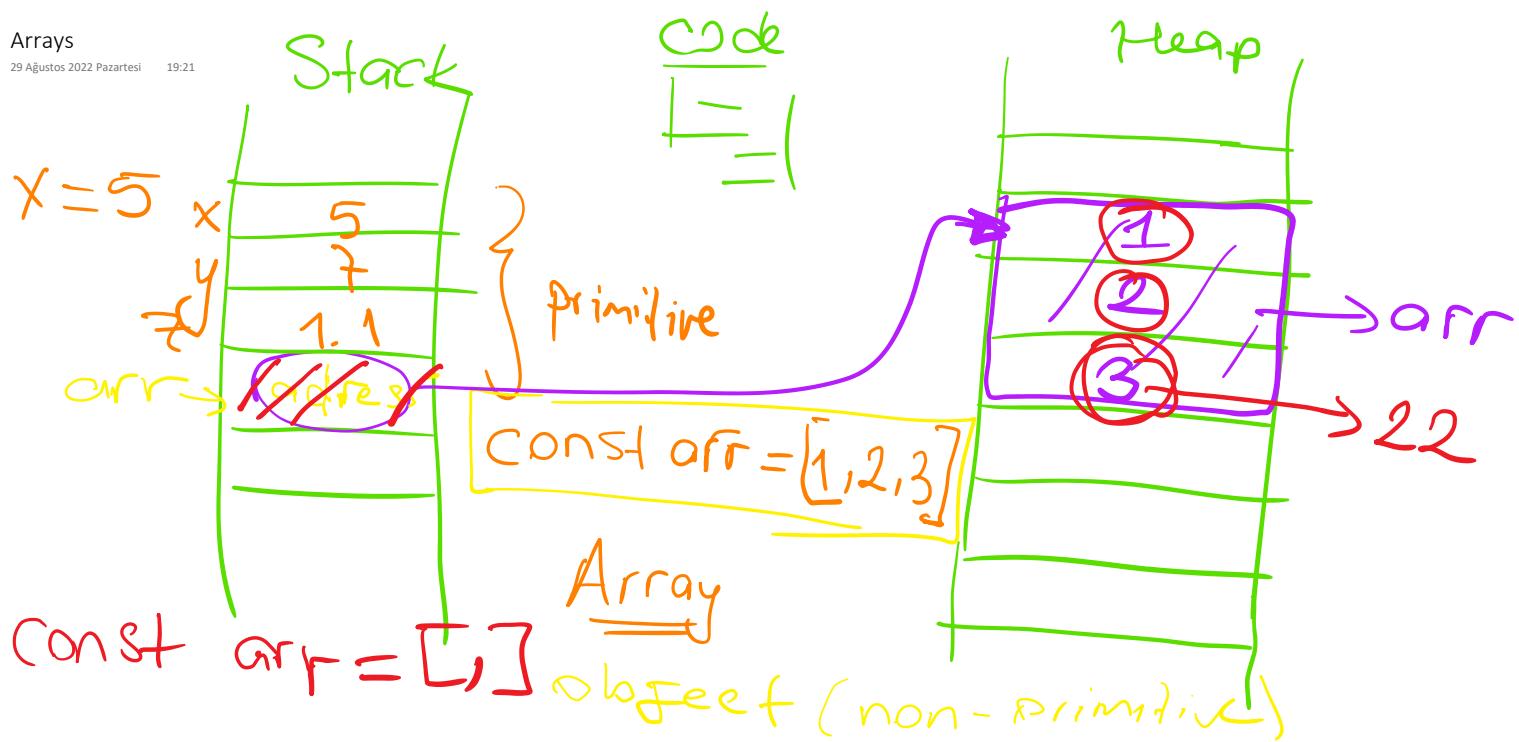
?

3

... .

```
<div class="row">
  Header
  [3m 9 min]
  col-1 {
    width: 33.33%
  }
  col-2 {
    width: 33.33%
  }
  col-3 {
  }
  col-12 {
    width: 100%
  }
```

- container
- row
- header col-12
- row
- menu col-3

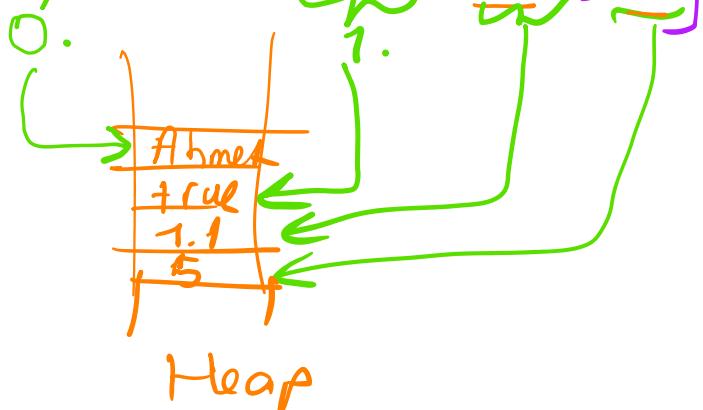


const myArr = ["Ahmet", true, 1.1, 5]

zero-index

$x = \text{myArr}[0]$

$\text{myArr}[3] \rightarrow 5$



$$3.333 \left(\frac{2}{2} \right)^2 \cdot \frac{1}{2} \cdot \frac{2^0}{1} \cdot \frac{-1}{2} \cdot \frac{2}{2} \cdot \frac{3}{2} \cdot \frac{-4}{1} \underline{\underline{0101}} \quad \underline{\underline{11-bit}}$$

$$\begin{array}{r} 3325 \\ 3341 \\ \hline 4+2 \\ 6 \end{array} \quad 0.5 + 0.125 = 0.625$$
$$1 \quad 0.25 + 0.2 = 0.4375$$

[^{array}]

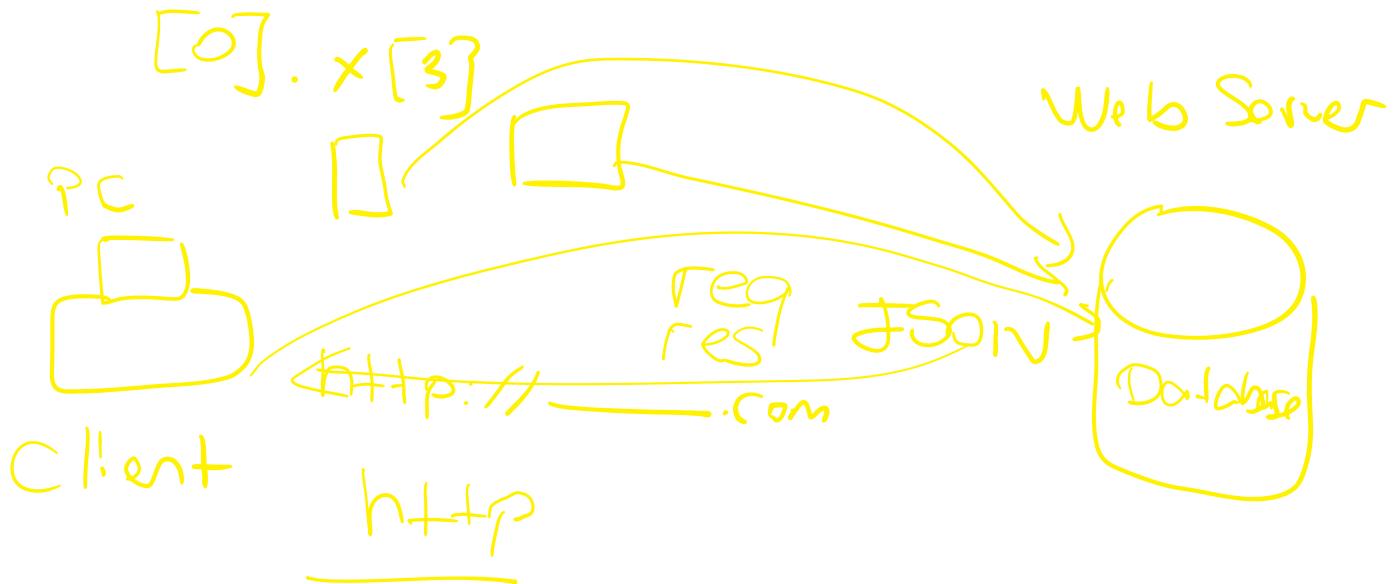
const Object = {
 key: value, unstructured
 ad: 'metmed', complex

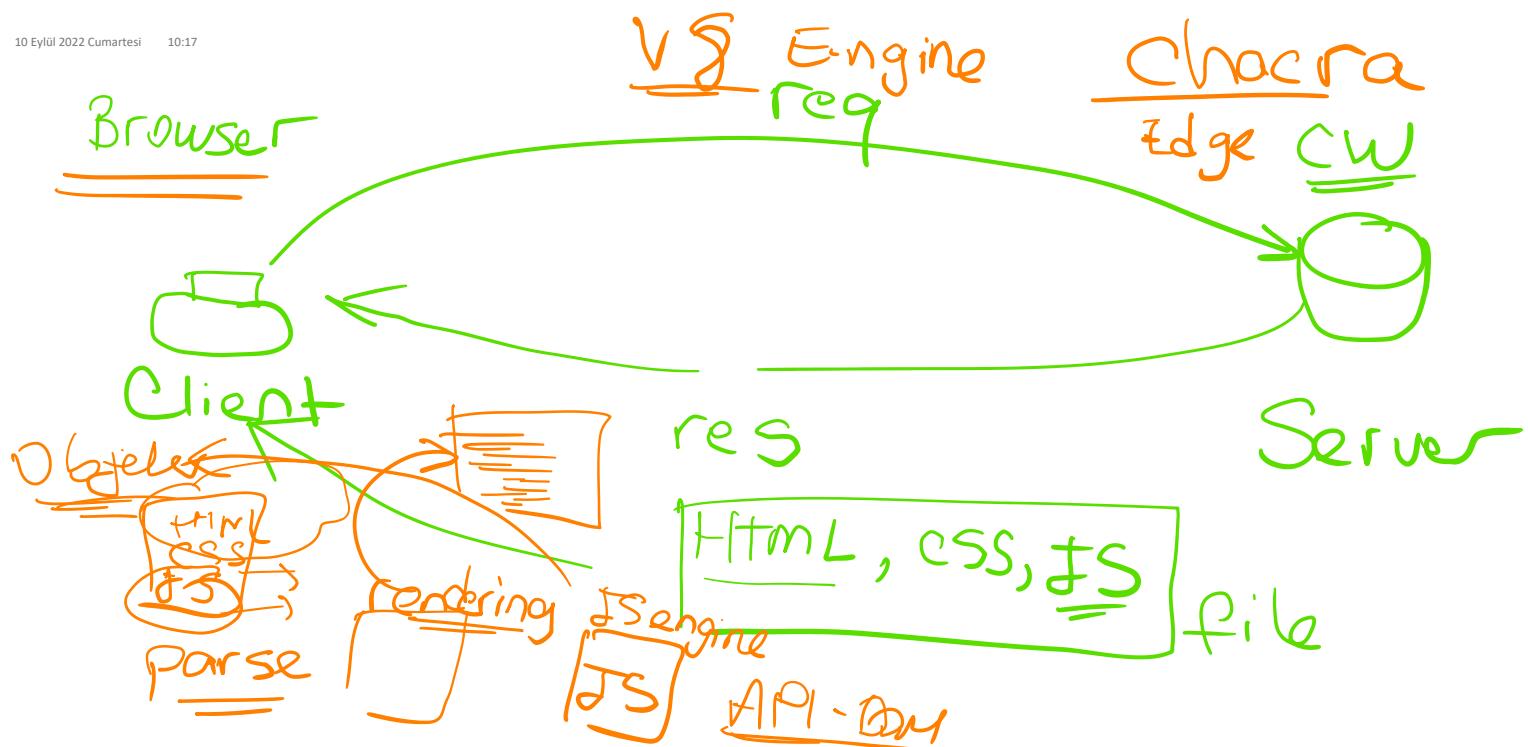
Soyad: 'yilmaz' { ?
 }
 [, ,] —
 function —

```
const x = {  
    key1: value1,  
    key2: value2,  
}  
→  
const y = {  
    name: 'Ahmet',  
    age: 22,  
}
```

JSON

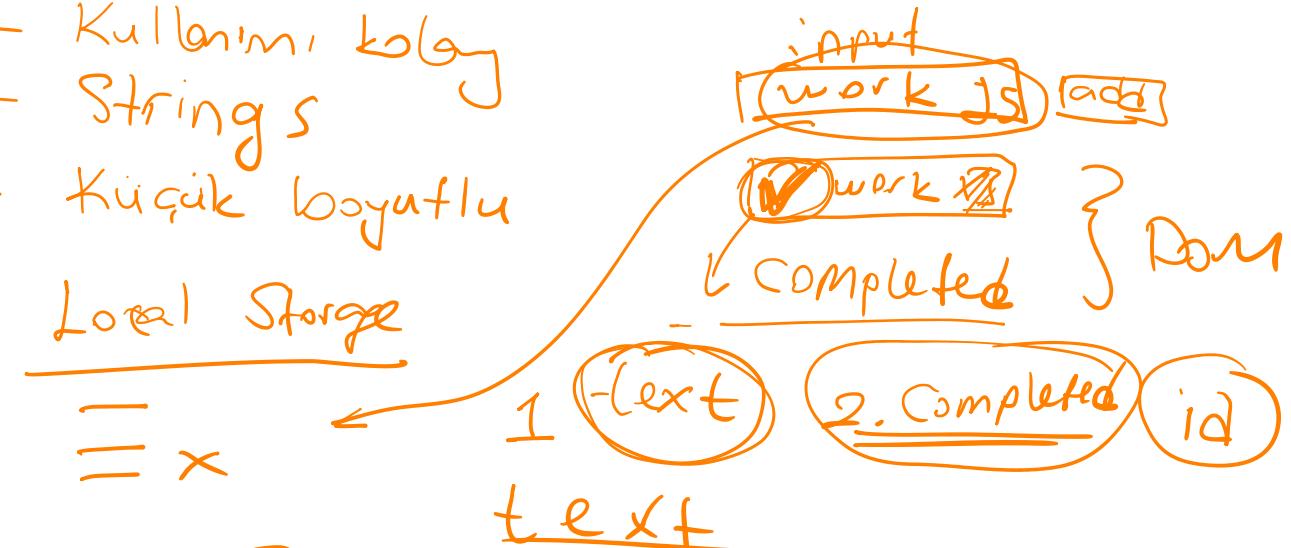
```
[{ } , { } , { } ]
```





Local Storage (String)

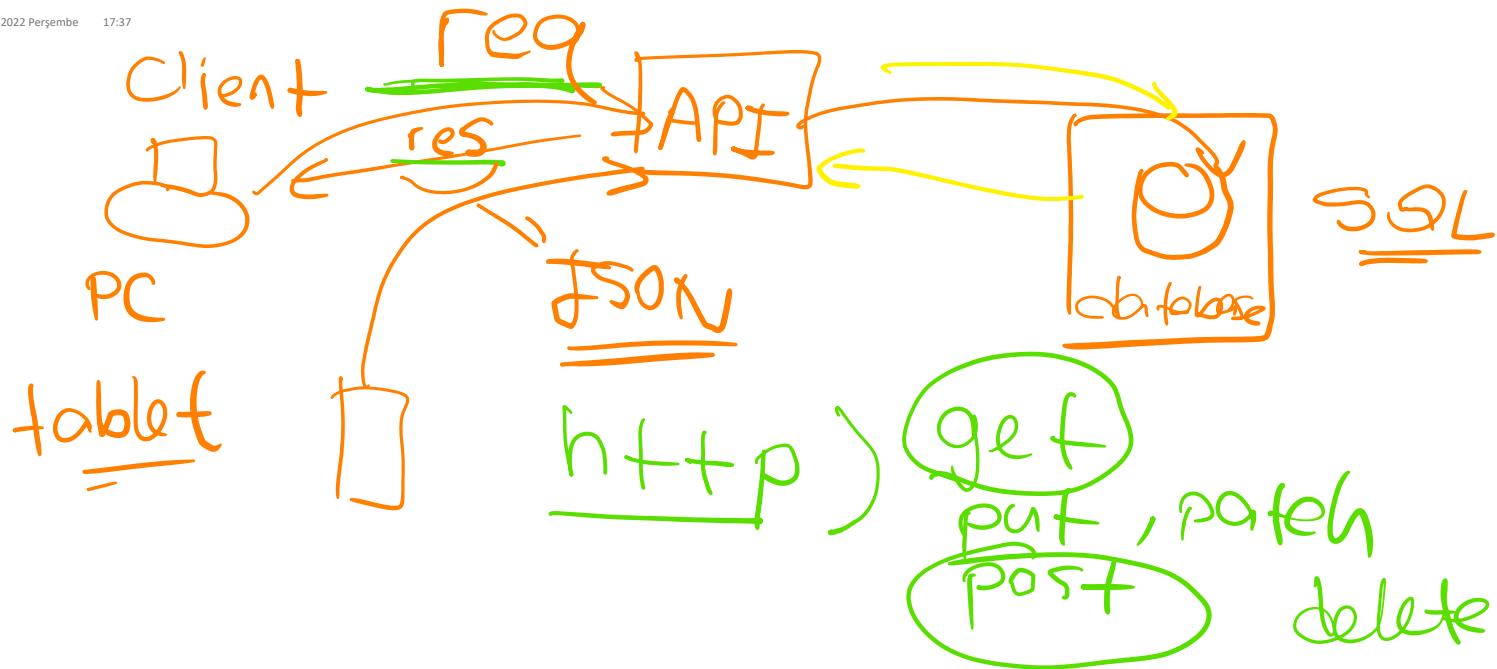
- Kullanımı kolay
- Strings
- Küçük boyutlu



`newTodo = {
 id : ? new Date().getTime(),
 text : input.value,
 completed : false,`

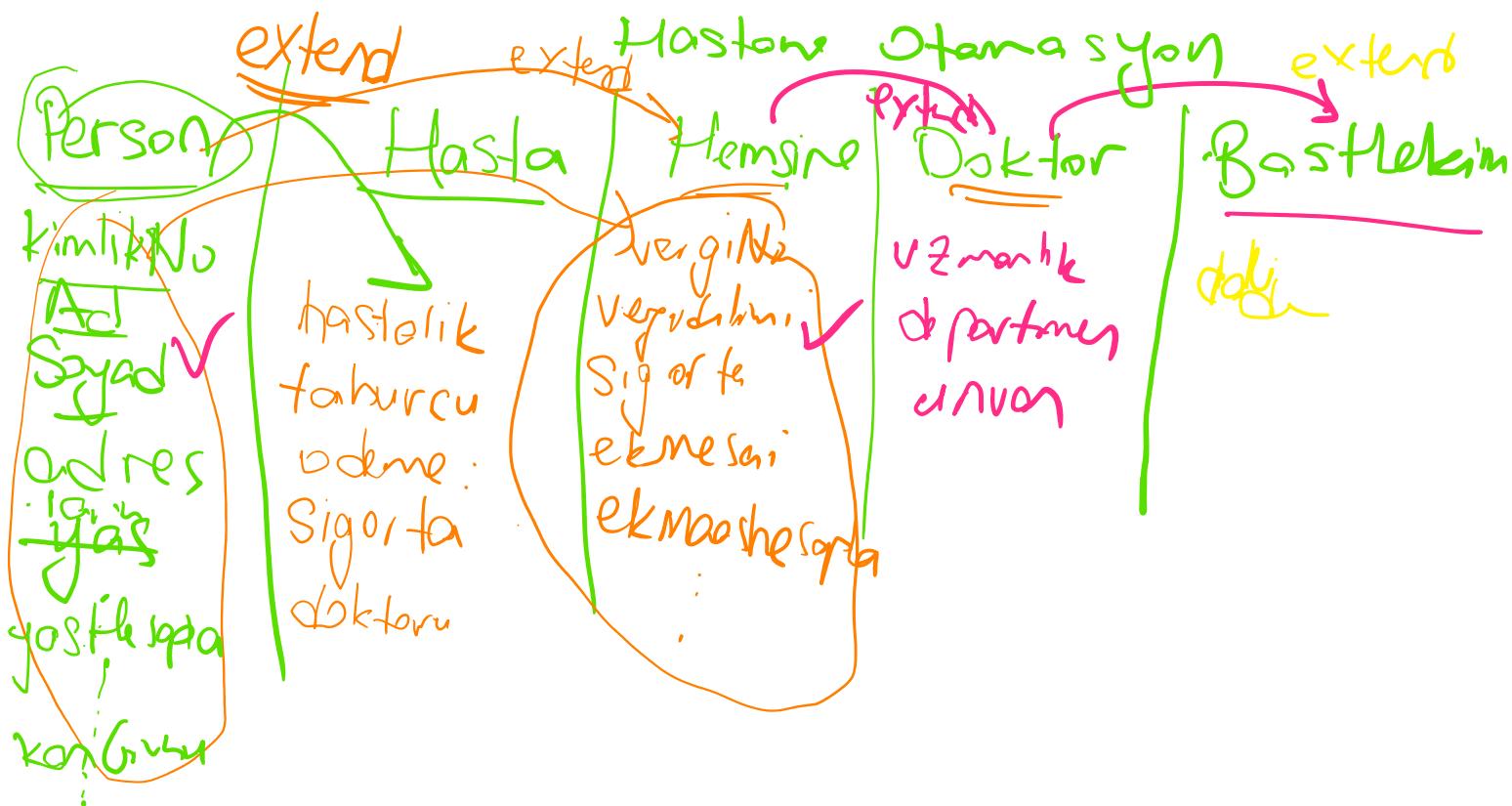
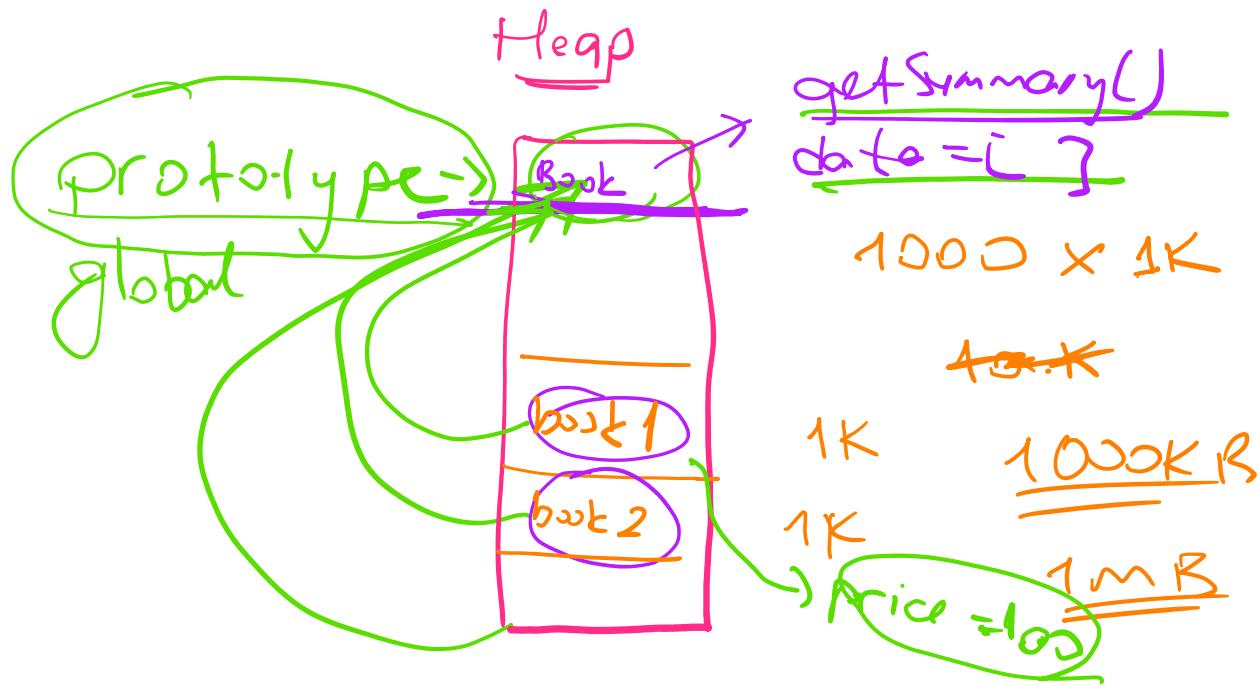
`let array = [3, 2, 1, 3, 2, 3] \Rightarrow JSON`

`JSON.stringify('todos', array)`



async function () {
}

const getUsers = async () => {
 const res = await fetch (" - - ")
 const data = await res.json()
 return data
}



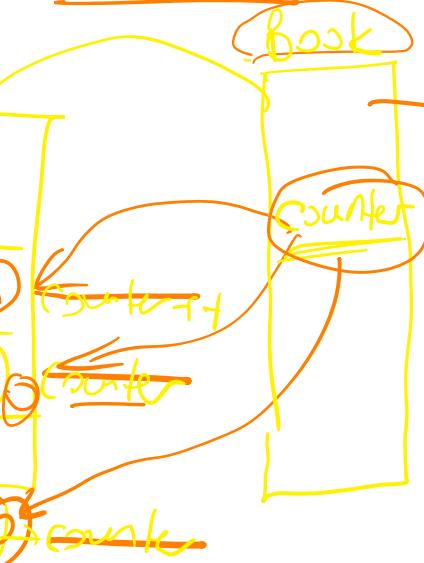
Book. counter++

Book. get(opt) = 500
1

price = 300 2

price = 200 3

Static counter = 0;



3 static
=
↓
class
deg iskelewi

<welcome/>

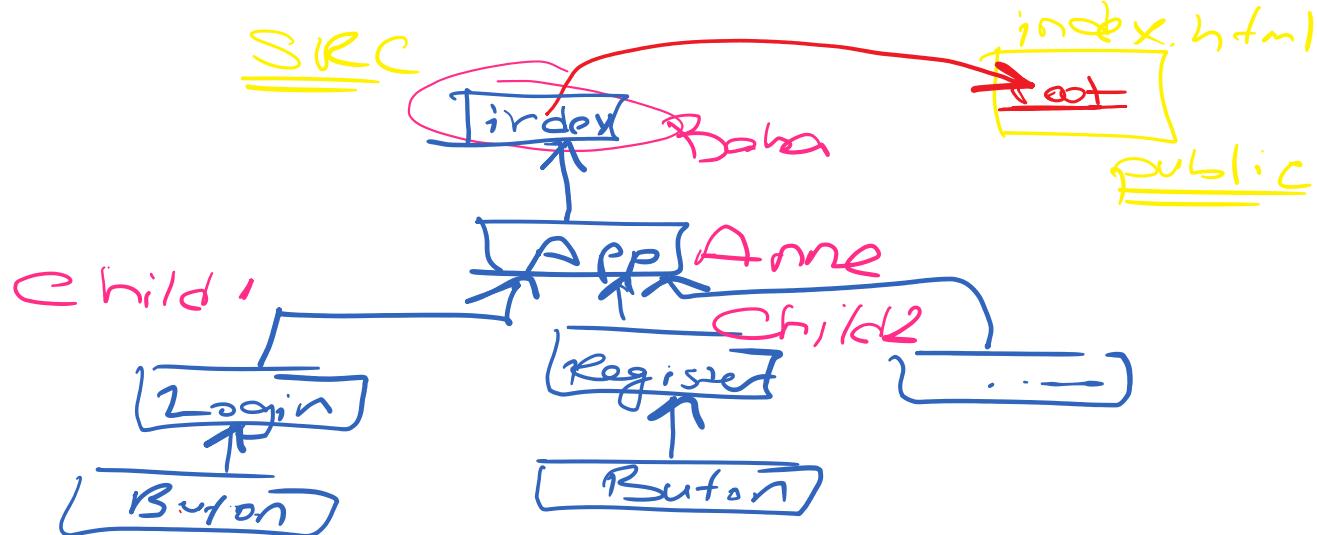
function welcome() {

 JS

 return <h1> - - - </h1>

 JSX

S



child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

[Register]
Button

Index Baba

index.html
root
public

child 1

[Login]
Button

App Anne

Child2

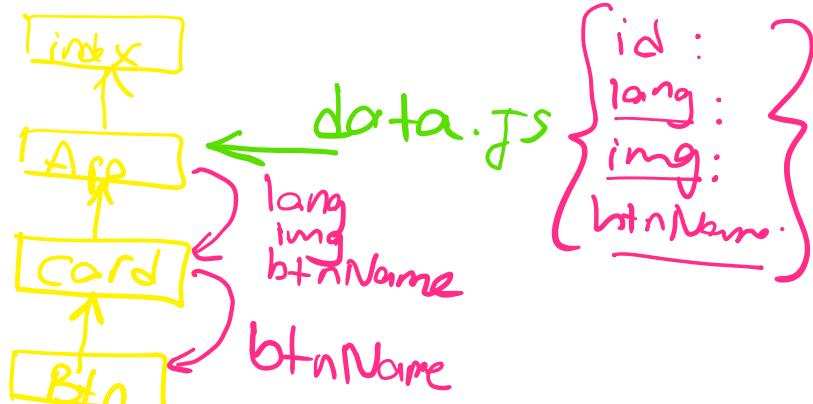
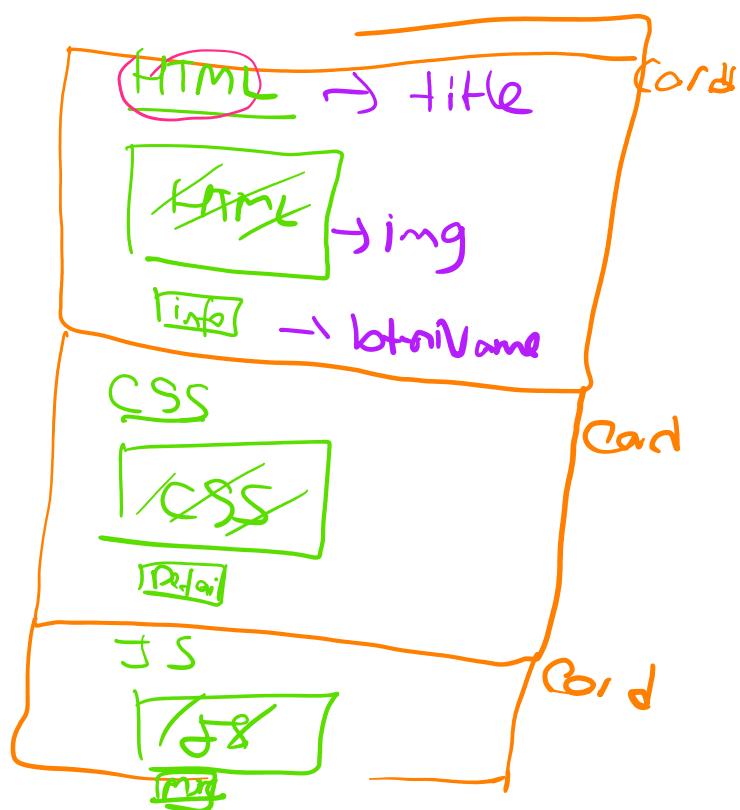
[Register]
Button

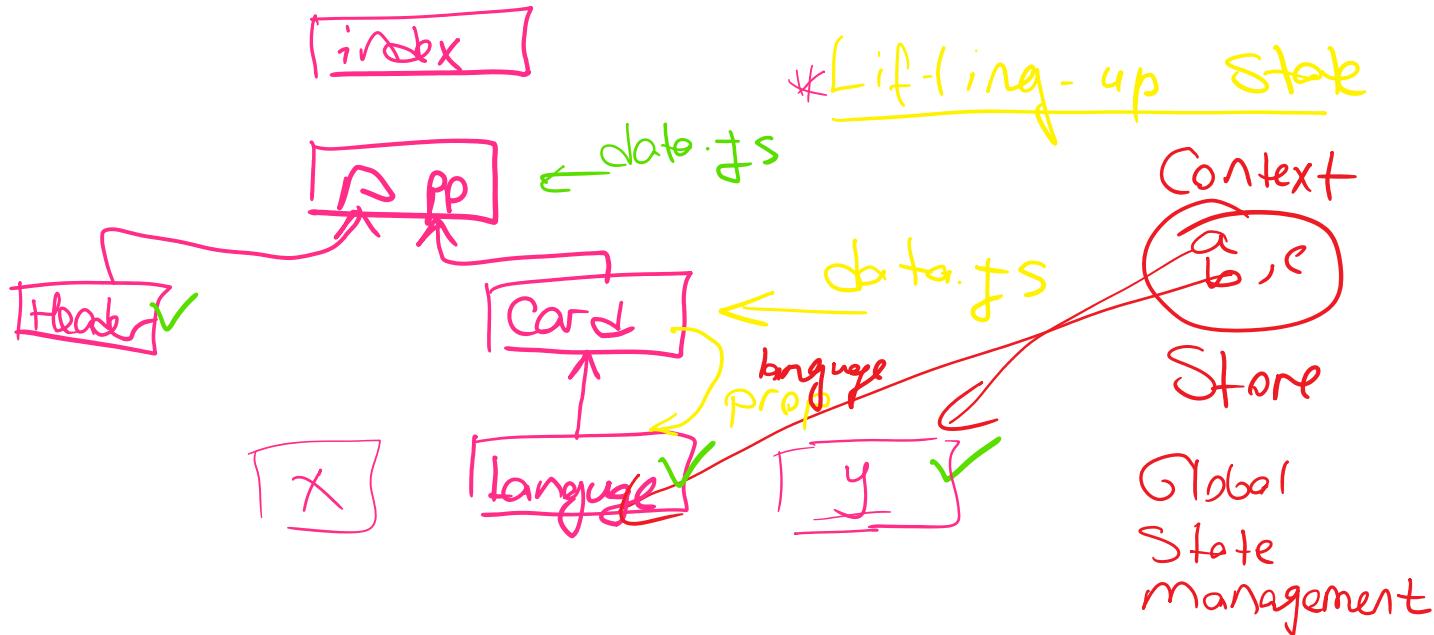
- Module CSS
 ✓ - Sass

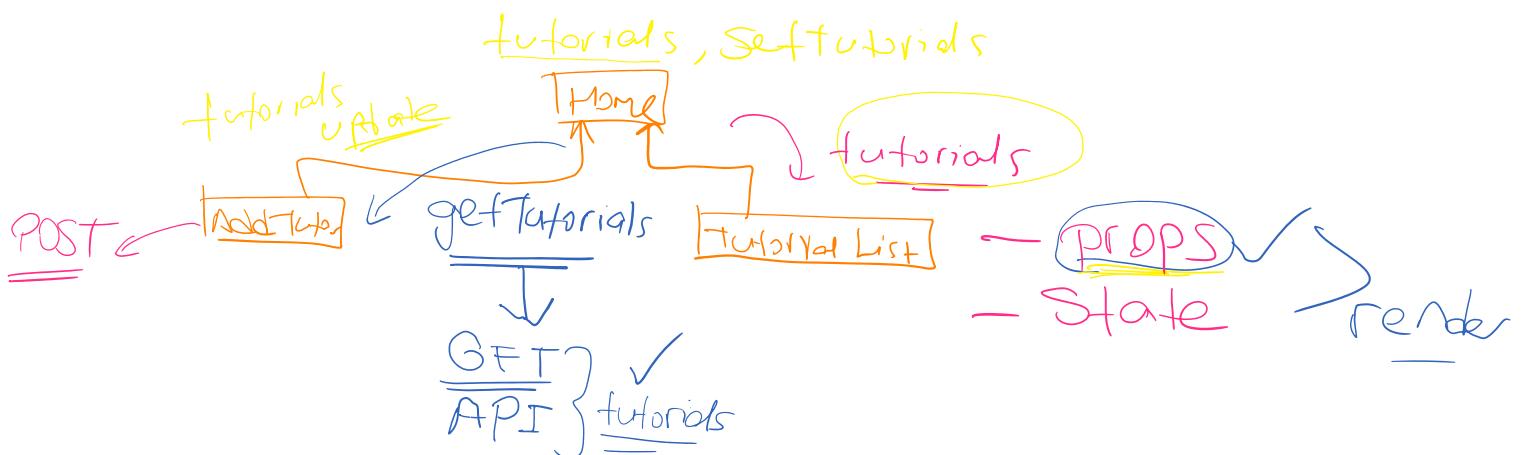
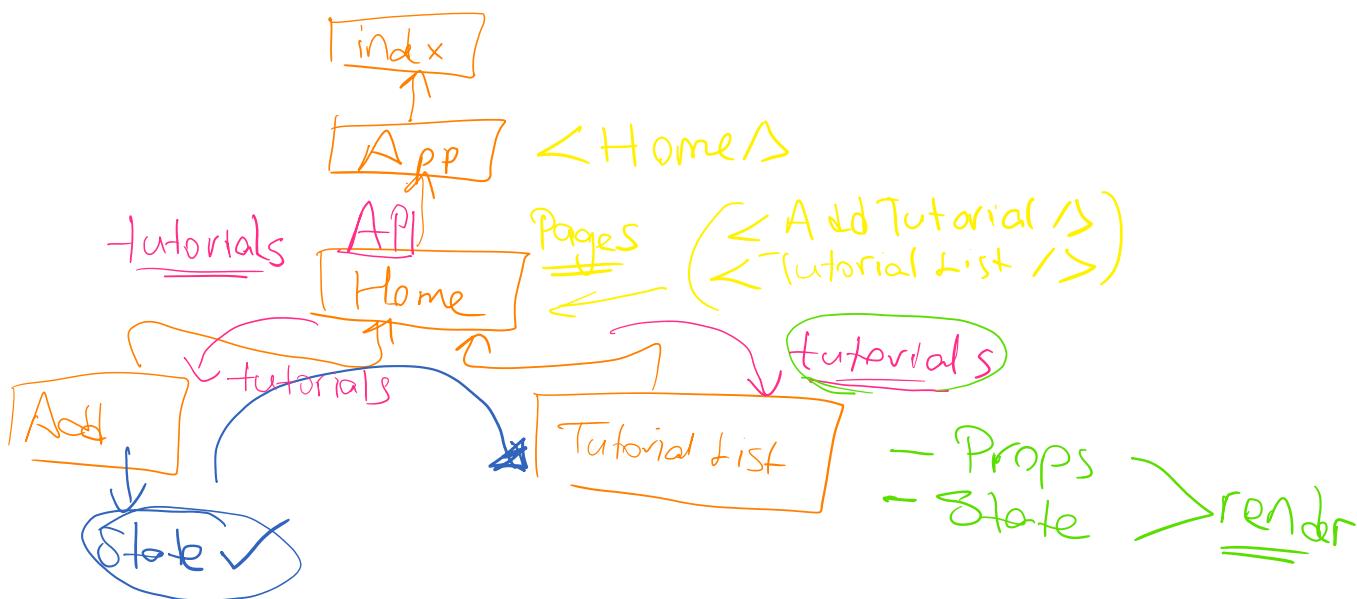
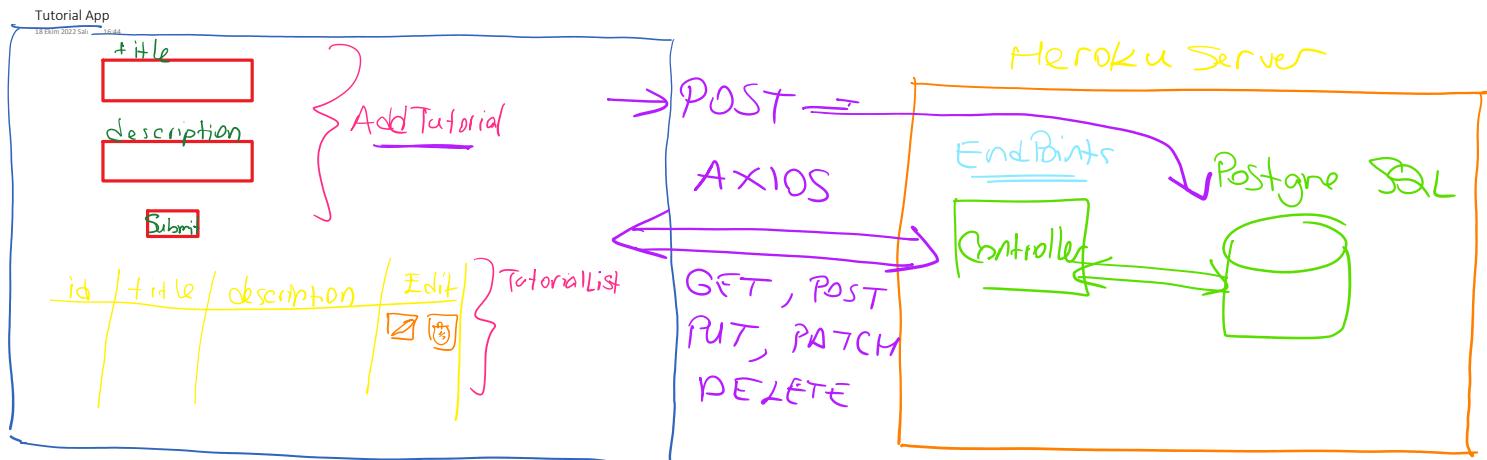
Basic Styling

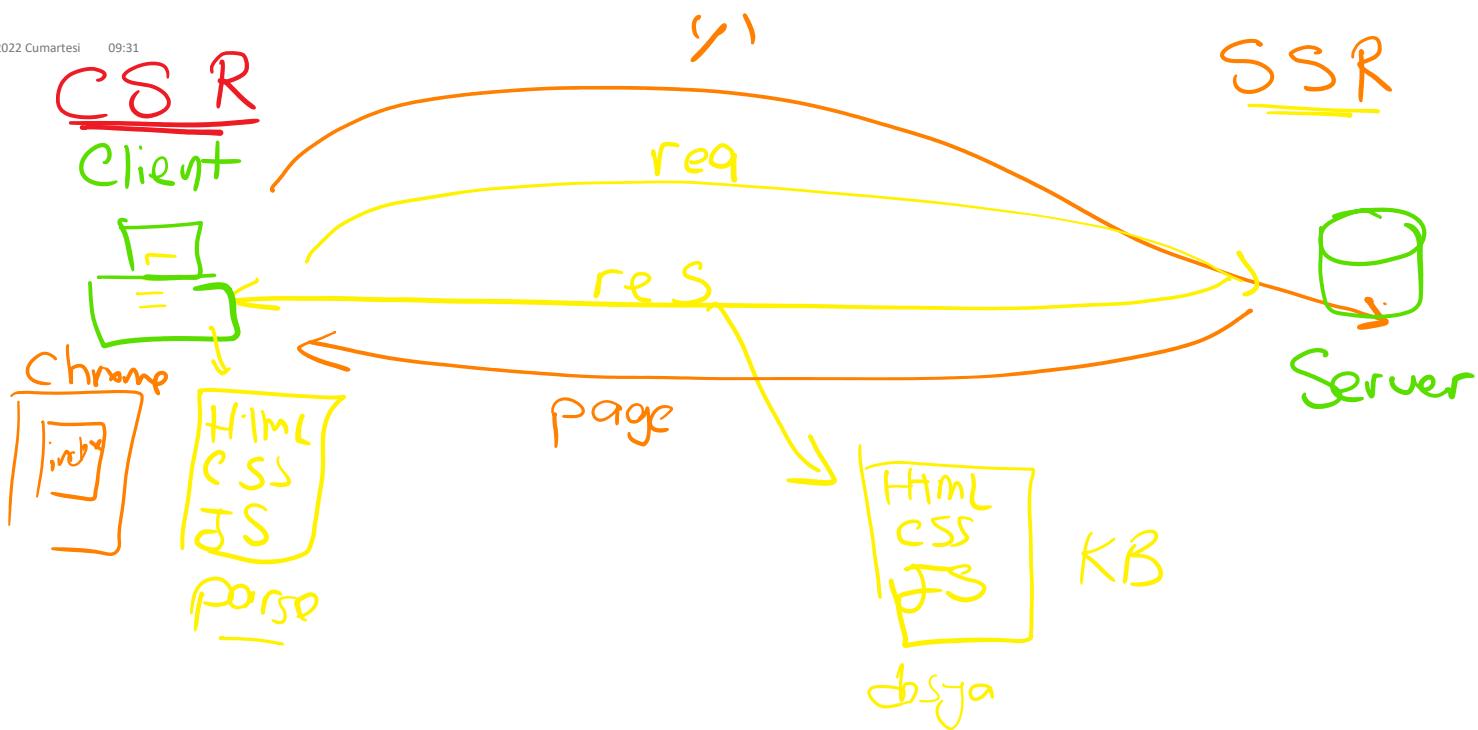
- ✓ Bootstrap - React Bootstrap
- ✓ Styled Components
- ✓ Material UI
- 3'rd

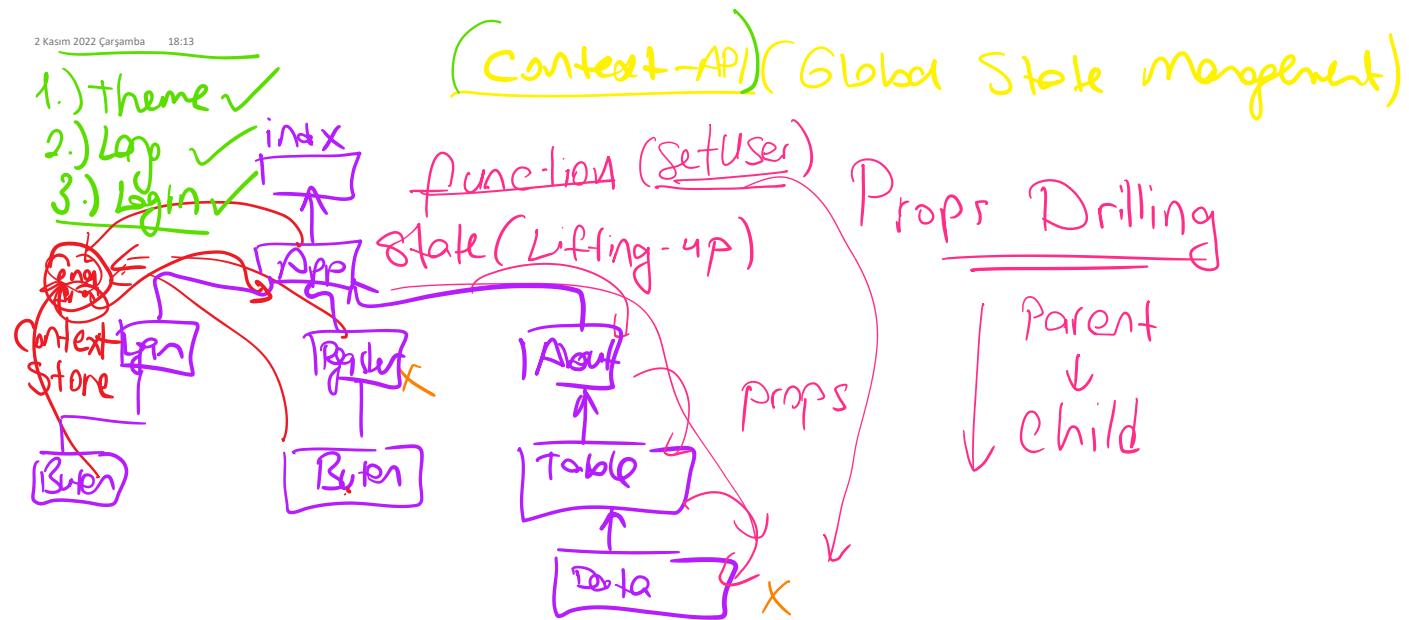
Ant design
tailwind









Creating

1.) `export const LoginContext = createContext();`

Providing

2.) `<LoginContext.Provider value={{ user, setUser }}>`
`<App /> { Child }`

`</LoginContext.Provider>`

Consuming

3.) `const { user, setUser } = useContext(LoginContext)`

State . Counter

State . Count . Counter

State . foods . foodsList

<Model>

open →

setOpen →

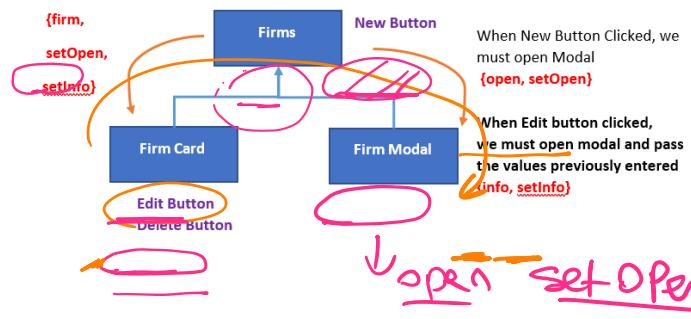
<Model>

1

(-1)

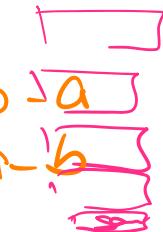
1

1 -1 → 1



ascii
sort()

number
Sort(a, b) ⇒ a -> b



`[["r", "z"].Sort()`

$$\text{Sort}(a, b) \Rightarrow \begin{cases} 5 - 4 = 1 & a - b = 1 \\ \text{ascii} & \\ (+), (-) & \\ & 0 \end{cases}$$

`[{"-1", "-3", "2"}]`

item item
a b

$$\text{Sort} \leftarrow \begin{cases} \frac{a > b}{1} & 1 \\ \frac{a < b}{-1} & -1 \\ \frac{a == b}{0} & 0 \end{cases}$$

`.Sort(a, b) => (`

`[{"id": -1, "stock": -3}, {"y": y+2}], [{"y": y+2}],`

`products.Sort(a, b) =>`

`a[arg] - b[arg]`



$$\frac{a+b}{y=y+2}$$