***Tee uus projekt või soovi korral jätka mõnda vana.***

Eesmärk Kontakt.jsx lehel saavutada selline vaade:

Graphical user interface, text, application, chat or text message

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

**Kui uus projekt, siis pead kogu Routingu otsast tegema** ->

routingu installimise: „npm install react-router-dom“

URL / HTML seosed: <Routes> <Route path=““ element={} /> </Routes>

Tee projektile **menüü** ja 3 lehte eraldi failidena (pages kausta sisse):

Avaleht.jsx,

Meist.jsx,

Kontakt.jsx. Tulevikus võtame kasutusele ka Seaded.jsx, võid ka selle kohe ära teha. Lisa igaühele sattumiseks nupud.

Proovi peast teha, aga võid võtta ka aluseks näidetena mõni teine õpetus või copy-paste lõigud koos tehtud projektist – ka sellisel juhul läbimõtlemine kuidas-mida teeme on väga kasulik.

Võta aluseks kuidas neid lehti teha kas projekt nr 2 või nr 4 või nr 5 (neile on tehtud detailsed print-screenid).

**Läheme nüüd lehti täitma:**

Tee KOLM useState osa **Kontakt.jsx** sisse:

(useState-le import)

const [aadress, määraAadress] = useState("Tallinn");

const [telefon, määraTelefon] = useState("5512345");

const [email, määraEmail] = useState("bla@baa.com");

Kuva need kõik 3 HTML-s. Selleks tee return sisse:

<div>

<div>{ aadress }</div>

<div>{ telefon }</div>

<div>{ email }</div>

</div>

Kontrolli kas kõik 3 tulid niimoodi HTMLi nagu sa useState() sulgude sees igaühele kirjutasid -> esmakordselt lehele tulles algväärtused.

Tee järgmiseks nupp – „Muuda inglise keelseks“ ning kui sellele klikkida, siis kutsu välja funktsioon, mille sisus on **määraBla(uus\_väärtus\_talle);** ( määraAadress(„“) määraTelefon(„“) määraEmail(„“) )

Tee seda **kõigile** **kolmele** – aadressile, telefonile, e-mailile.

Kontrolli localhost-is, et nuppu vajutades kõik muutuvad selleks, milleks sa teda funktsiooni sees määrasid.

Tee järgmiseks dünaamiline väljakuvamine: { avaldis && <div>??</div> }. Selleks tee ka **neljas useState(),** mille nimeks pane ingliseKeelne (vasakpoolne kandiliste sulgude sees), loo funktsioon (parempoolne kandiliste sulgude sees) ja algväärtuseks „ei“: **useState(„ei“);** (jälgi jutumärke)

Nupu peale vajutades (kus sa kutsusid 3x määra() välja) muuda nüüd see ingliseKeelne samuti „jah“-ks läbi **määra\_() funktsiooni** .

Tee HTMLi dünaamiline väljakuvamine ehk **kui ingliseKeelne muutuja on „jah“-**ks määratud, siis kuva välja <div> sees „Leht on inglise keelne“. Seda lauset ei kuvata, kui ingliseKeelne on „ei“.

Kontroll kas muutuja võrdub või ei võrdu mingi väärtusega käib läbi kolme võrdusmärgi ( === )

Tee ÜKS useState osa **Meist.jsx** sisse:

(useState-le import, jälgi jutumärkide õigsust!)

const [message, määraMessage] = useState(„Vali mõni tegevus“);

Kuva see HTML-s. Selleks tee return sisse:

<div>

<div>{ message }</div>

</div>

Tee 3 nuppu:

<div>

<div>{ message }</div>

<button>Kandideeri tööle</button

<button>Vaata meie töötajaid</button

<button>Võta meie ühendust</button

</div>

Iga nupuga muuda sõnumit. Esimesele nupule klikkides võiks sõnum tulla:

„Selleks saada meile e-mail [jobs@html-css.com](mailto:jobs@html-css.com)“

Teisele nupule klikkides võiks sõnum tulla:

„Meil on 10 töötajat, kelle info ilmub veebilehele lähiajal“

Kolmandale nupule klikkides võiks sõnum tulla:

„Ühenduse võtmiseks mine lehele Kontakt“