Tee kodutoo1 projekti juurde leht „Seaded“.

Saadame veebilehe dark\_mode seadistuse brauserisse. Tegelikult mingit dark\_mode peale ei tule (kuigi hiljem võiks, praegu näeme kasutaja valikut tänu väljakuvatavale sõnale).

Nupuvajutusega pane käima funktsioon kus on sisse kirjutatud: **localStorage.setItem(„veebilehe\_kujundus“, „dark\_mode“);**

*Copy-paste siit ei õnnestu, kuna jutumärgid on teistsugused Visual Studio Code-s*

Vaata kas see ilmub, minnes veebilehel parem klõps -> inspect -> application -> local storage

Tee teine nupp, milles kutsu välja funktsioon, kuhu sisse kirjuta: **localStorage.setItem(„veebilehe\_kujundus“, „light\_mode“);**

Pane return() div sisse:

**{ localStorage.getItem(„veebilehe\_kujundus“)===“dark\_mode“ && <div>TUME LEHT</div> }**

**{ localStorage.getItem(„veebilehe\_kujundus“)===“light\_mode“ && <div>HELE LEHT </div> }**

Muuda kirjet, vajutades nuppu, mis kutsub funktsiooni välja. Tee **refresh**. Vaata kas sõna muutub. Muuda uuesti ja tee **refresh**. LISA ISESEISVALT KA KOLMAS (colored\_mode?) VÕIMALUS.

*MUUTUB AINULT REFRESHIDES, SEST VAID* ***useState*** *MUUDAB HTMLi*

Muuda nüüd **useState** abil nii, et ei pea refreshima:

Ülesse useState **import**

useState muutuv muutuja vasakule poole, nt „**kujundus**“ ja funktsioon millega muuta „muudaKujundus“. Pane useState algväärtuseks „light\_mode“.

Pane nuppude sisse nüüd kus on:’

localStorage.setItem(„veebilehe\_kujundus“, „dark\_mode“)

lisaks ka:

**muudaKujundus(„dark\_mode“);** ehk kutsu välja useState funktsioon

useState muudetav muutuja „kujundus“ pane nüüd localStorage.getItem asemele:

**{ kujundus === „dark\_mode“ && <div>TUME LEHT</div> }**

**{ kujundus === “light\_mode“ && <div>HELE LEHT </div> }**

Nüüd muutub silme ees, aga iga kord refreshides on HELE LEHT.

Et ka refreshides oleks õige kujundus (viimati localStorage.setItem abil saadetud), muuda useState algväärtuseks:

**const [kujundus, muudaKujundus] = useState(localStorage.getItem(„veebilehe\_kujundus“));**