### Koodin kommentointi/Readme

**Login -komponentin toiminta:**

React-komponentti nimeltään Login, joka on tarkoitettu käyttäjän kirjautumisnäkymän toteuttamiseen. Seuraavassa selostetaan koodin toiminta vaiheittain:

Tuonnit ja Tyylien Liittäminen

useNavigate tuodaan react-router-dom-kirjastosta, jota käytetään sivujen välillä navigointiin.

Login.css-tyylitiedosto liitetään mukaan, joka sisältää kirjautumislomakkeen tyylit.

Reactin useState-hook tuodaan, jonka avulla hallitaan komponentin tilaa.

useUser on konteksti-hook, joka mahdollistaa käyttäjätietojen käsittelyn (esim. kirjautuminen).

index.css-tyylitiedoston tuonti, joka sisältää yleiset tyylit.

Link tuodaan myös react-router-dom-kirjastosta, jota käytetään linkkien luomiseen.

Komponentin Rakenne

Login on funktionaalinen komponentti, joka sisältää kirjautumislomakkeen.

Kaksi tilamuuttujaa (username ja password) alustetaan tyhjiksi merkkijonoiksi useState-hookilla.

navigate-funktio alustetaan useNavigate-hookilla, jotta voidaan ohjata käyttäjä eri reitteihin.

validate on asynkroninen funktio, joka suoritetaan lomakkeen lähetystapahtumassa.

Jos käyttäjätunnus ja salasana eivät ole tyhjiä, yritetään kirjautua sisään login-funktiolla, joka saadaan useUser-hookista.

Kirjautumisen onnistuessa siirrytään etusivulle (navigate("/")).

Virhetilanteessa tulostetaan virhe konsoliin.

Lomakkeen Renderöinti

Komponentti palauttaa JSX-syntaksilla kirjautumislomakkeen, jossa on kaksi syöttökenttää: käyttäjätunnus ja salasana.

Syöttökentät päivittävät tilamuuttujia setUsername ja setPassword kun niiden arvo muuttuu.

Lomake sisältää kaksi painiketta: "Kirjaudu" lähetyspainikkeena ja "Poista tili" ilman toiminnallisuutta.

Link-komponentti tarjoaa reitin, josta pääsee luomaan uuden tilin (/CreateAcc).

Kokonaisuutena

Kun käyttäjä täyttää lomakkeen ja painaa "Kirjaudu"-painiketta, validate-funktio käynnistyy. Mikäli käyttäjätunnus ja salasana ovat kelvollisia, kirjautumisyritys tehdään. Onnistuneen kirjautumisen jälkeen käyttäjä ohjataan etusivulle. Epäonnistuneessa tapauksessa virhe näytetään konsolissa. Lisäksi käyttäjällä on mahdollisuus siirtyä tilinluontisivulle klikkaamalla "Luo tili" -linkkiä.

### tyylitys käyttäen sass kuvauskieltä

Jokaiselle sivulle ja komponentille on luotu SASS-tyylitiedosto, joka määrittelee tyylejä. Koodissa käytetään SASS:n syntaksia ja ominaisuuksia, kuten muuttujia ja sisäkkäisiä sääntöjä. Seuraavassa selostetaan koodin toiminta:

@import "../index.sass" tuo toisen SASS-tiedoston, joka todennäköisesti sisältää yleisiä muuttujia ja tyylejä, joita käytetään tässä tiedostossa.

.formcontent-luokka:

Asettaa elementin näyttämään flexboxina ja suuntaa sen lapset pystysuunnassa.

Antaa elementille padding-arvot ja leveyden sekä marginaalin.

.formlabel-luokka:

Määrittää fonttikoon, painon ja marginaalin alapuolelle.

#submit-id:

Asettaa marginaalin, taustavärin, tekstin värin, reunuksettomuuden, padding-arvot, fonttikoon, marginaalin yläpuolelle, pyöristetyt kulmat, kirjasinpainon, kirjainvälin ja fontin perheen.

Sisäkkäiset &:hover-säännöt määrittävät mitä tapahtuu kun hiiri on elementin päällä (muuttaa tausta- ja tekstinväriä).

.input-luokka:

Asettaa fonttikoon, padding-arvot, nollaa reunuksen, lisää varjon, pyöristää kulmat ja asettaa leveyden.

&:active, &:focus ja &:hover ovat pseudo-luokkia, jotka määrittävät elementin tyylin kun se on aktiivinen, fokusoitu tai hiiren yli liikutetaan. Ne muuttavat varjon ulkonäköä, poistavat reunuksen ja ääriviivan, ja &:hover lisää taustavärin.

.linksign-luokka:

Asettaa paddingin ja marginaalin.

Koodissa käytetty $blk, $bg, $loact ja $active ovat SASS-muuttujia, joiden arvot on määritelty tuodussa index.sass-tiedostossa. Nämä muuttujat viittaavat eri väriarvoihin, jotka voidaan muuttaa keskitetysti index.sass-tiedostossa.

Yhteenvetona, tämä SASS-koodi luo tietyn visuaalisen ilmeen kirjautumislomakkeelle ja sen elementeille, kuten syöttökentille ja linkeille, käyttäen SASS:n ominaisuuksia kuten muuttujia ja sisäkkäisiä sääntöjä.