

CGI proovitöö

Maria Pibilota Murumaa
maria.pibilota@gmail.com
22.04.2020

Kasutusjuhend:

Programmi käivitamiseks, tuleb see alla laadida siit:

Avamiseks vajuta "kalkulaator.html" failile ning projekt avaneb brauseris. Kasutasin eelkõige Google Chrome, kuid rakendus töötab ka teiste veebilehitsejatega (nt: Firefox jpt).

Päeva pikkuse arvutamine:

1. Sisesta laiuskraad vahemikus -90 kuni 90
2. Sisesta pikkuskraad vahemikus -180 kuni 180
3. Vali meelepärane kuupäev
4. Vajuta nuppu "Arvuta"

või

1. Vali meelepärane kuupäev
2. Vali kaardil meelepärane asukoht -> Klicki valitud asukohale kaardil.

Ekraani paremal poolel ilmuvad päikesetõus, -loojang kui ka päeva pikkus UTC ajavööndis.

Päeva pikkuse muutumise kuvamine graafiliselt:

1. Vali päeva pikkuse arvutamise osas meelepärane asukoht kaardilt või sisesta pikkuse ja laiuskraad.
2. Vali perioodi alguskuupäev
3. Vali perioodi lõppkuupäev
4. Vajuta nuppu "Arvuta"

Ekraanile kuvatakse graafik, mis kirjeldab asukoha päeva pikkuse muutumist eelnevalt valitud kuupäeva lõikes

Lahendamiseks kulus umbes 19h.

Mis valmistas raskusi?

Hakkasin ise päikesetõusu ja -loojangu jaoks koodi kirjutama [valemite põhjal](#). Saadud tulemus ei olnud kuigi täpne. Pika pusimise peale otsustasin siiski kasutada juba valmis olevat lahendust, sest see andis täpsema tulemuse: <https://github.com/mourner/suncalc>. Käivitamise lihtsuse huvides, ei laadinud ma teeki alla, vaid lisasin selle JS faili (kärpides ridu) ning viitasin töö autorile.

Samuti oli keerukas valida kuidas tööd lahendada. Otsustasin lihtsa HTML, CSS ja JS kasuks, ning mitte kasutada React'i, sest nii on lihtsam tööd käivitada ja ei pea erinevaid mooduleid alla laadima.

Probleem, millest ei saanud lõpuks lahti, on seotud päeva pikkuste kuvamisega graafiliselt. Nimelt ei lase y telg kasutada kellaaja vormingut (nt 15:59 ja järgmine väärtus oleks 16:00). Ma oleksin ise üritanud määrata y-telje tüübiks h:mm vormingu, ent see ei õnnestunud. Siis tuli mõte, et seada sammu väärtused uueks nt 00 -> 15:00 ; 30 -> 15:30 ; 60 -> 16:00 ; 90 -> 16:30. Idee seisneb selles, et anname ette arvutuskäigu, kuidas y-telje samm arvestatakse.

Mis oli kerge?

Tore oli teha graafikut ja kaardiga seonduvat ülesannet. Samuti oli vahva kujundada tööd ning lisada paigutus jne.

Lahendamine

Kaart:

Kuna ma polnud varem väga kaartidega kokku puutunud, tuli kasu Leaflet API uurimisest. Sealt leidis ilusti kõik ülesse, mida vaja läheb või idee, kuidas teostada.

Loomes kaardi, mille asukoht on alguses default. Lisame ka markeri ning vihje, et saame asukohta muuta, kui klikime kaardile.

Kui vajutada nupule "Arvuta" määratakse sisendi järgi kaardil uus laius- ja pikkuskraad.

Muudame automaatselt ka *pop-up* teksti 'Asukoht' + sisseloetud koordinaadid.

Kui klikime kaardil, et muuta asukohta, siis kustutame markeri kihi ja lisame uue markeri klikitud kohta.

Arvutus:

Kui klikkida nupule "Arvuta", siis loome sisestatud väärtused sisse (laiuskraad, pikkuskraad, kuupäev). Kasutades SunCalc teeki, teeme vajalikud arvutused, et saaksime päikesetõusu ja -loojangu aja vastavalt asukohale.

Päeva pikkuse leidmiseks, eraldame päikesetõusu ja -loojangu ajast UTC tunni ja UTC minuti väärtuse. Kui me kasutaks lihtsalt `.getHours()` meetodi, siis antakse meile Eesti ajas kõik väärtused, mida me ei taha.

Lahutades loojangu ajast tõusu aja saame päeva pikkuse, mille kuvada ekraanil.

Graafik:

Loome sisestatud asukoha ja kuupäevade vahemiku sisse. Teeme vajalikud teisendused ja lisame iga päeva pikkuse ja kuupäeva eraldi järjendisse. Järjendite põhjal loome uue joondiagrammi, mille kuvame ekraanile.

Kasutatud teegid:

Päikesetõusu ja -loojangu leidmiseks: <https://github.com/mourner/suncalc>

Graafiku joonistamiseks: <https://www.chartjs.org/>

Fondid: <https://fonts.google.com/>

Kaart: <https://leafletjs.com/>

PS! Ühtegi teeki ei pea alla laadima.