**Visual Studio Code: Kujunda elektroonilise kella näide vastavalt maitsele või kindlale teemale, mahutades kella täisekraanile, et saaks kasutada lauakella või ekraanisäästja asemel. Selleks, et see sobiks paljudele ekraanidele, kasuta kujunduse loomisel protsendilisi väärtusi (nt width: 100%; ) või nt võimalda kella suurust kasutajal muuta.**

Selleks, et kujundada elektroonilist kella Visual Studio Code'is, tuleb kasutada veebirakenduste arendamiseks mõeldud tööriistu, näiteks HTML, CSS ja JavaScript. Siin on mõned sammud, mida võid järgida:

1. Loome HTML faili, kus kirjutame kella jaoks vajaliku struktuuri. Näiteks:

phpCopy code

<div class="clock"> <div class="hours"> </div> <div class="minutes"> </div> <div class="seconds"> </div> </div>

1. Kujundame kella CSS abil, kasutades protsendilisi väärtusi, et kohandada kella suurust vastavalt ekraani suurusele. Näiteks:

cssCopy code

.clock { width: 100%; height: 100%; display: flex; justify-content: center; align-items: center; font-size: 5em; }

1. Kirjutame JavaScripti koodi, mis võimaldab kella käivitada ja töötab värskendades kella iga sekundit vastavalt kasutaja ajatsooni seadistusele. Näiteks:

javascriptCopy code

function updateClock() { const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } setInterval(updateClock, 1000);

1. Lisame HTML-le JavaScripti faili ja alustame kella käivitamist. Näiteks:

phpCopy code

<script src="clock.js"></script> <script> updateClock(); </script>

Lõpuks saame salvestada failid ja avada HTML-faili Visual Studio Code'is, et näha, kuidas kella välja näeb. Kui soovite, võite lisada kasutajaliidese elemendid, mis võimaldavad kasutajal kella suurust muuta või valida erinevaid teemasid.

**Kas html kood on õige: <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours"></div> <div class = "minutes"></div> <div class = "seconds"></div> </div> <script> updateClock(); </script> </body> </html>**

Jah, see HTML-kood on õige. See loob veebilehe, millel on metaandmed, HTML-lehtede põhiline struktuur, lingid stiililehtede ja JavaScripti-failide jaoks ning veebilehe sisu. Veebilehel on ka üks **div**-element, mis sisaldab kella loomiseks kolme alam-elementi **hours**, **minutes** ja **seconds**. Viimane rida kutsub JavaScripti funktsiooni **updateClock()** välja, mis võimaldab kella pidevalt värskendada.

**Kas javascripti kood on õige: function updateClock(){ const currentTime = newDate(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } setInterval(updateClock, 1000);**

Peaaegu õige. Funktsiooni esimeses reas tuleks aga asendada **newDate()** **new Date()**-ga, et uus kuupäevaobjekt korrektselt luuakse.

Seega õige kood peaks olema:

javascriptCopy code

function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } setInterval(updateClock, 1000);

See funktsioon loob uue kuupäevaobjekti, võtab sellest hetkeaja tundi, minuteid ja sekundeid, lisab need sobivalt 0-ga, kui need on kahekohalised ja määrab päeva- või ööaja. Seejärel värskendab see kella HTML-i kaudu, muutes vastavate elementide sisu. Lõpuks käivitatakse funktsioon **updateClock** iga sekundi järel, kasutades funktsiooni **setInterval**.

**Visual Studio Code: Kujunda elektroonilise kella näide vastavalt maitsele või kindlale teemale, mahutades kella täisekraanile, et saaks kasutada lauakella või ekraanisäästja asemel. Selleks, et see sobiks paljudele ekraanidele, kasuta kujunduse loomisel protsendilisi väärtusi (nt width: 100%; ) või nt võimalda kella suurust kasutajal muuta. Kasuta eventListener-e'**

Siin on näide, kuidas elektroonilist kella saab kohandada vastavalt kasutaja eelistustele ja kasutada täisekraanil lauakellana või ekraanisäästjana. Kujunduse loomisel kasutatakse protsendilisi väärtusi ja kasutajal on võimalik kella suurust muuta nuppu klõpsates.

HTML:

phpCopy code

<!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="style.css"> <title>Clock</title> </head> <body> <div class="clock-container"> <div class="clock"> <div class="hours"></div> <div class="minutes"></div> <div class="seconds"></div> </div> <button id="resize-button">Resize Clock</button> </div> <script src="script.js"></script> </body> </html>

CSS:

cssCopy code

body { margin: 0; padding: 0; background-color: black; } .clock-container { display: flex; justify-content: center; align-items: center; height: 100vh; } .clock { display: flex; flex-direction: column; justify-content: center; align-items: center; width: 100%; height: 100%; font-size: 10vw; color: white; } button { position: absolute; top: 20px; right: 20px; padding: 10px 20px; font-size: 1.2rem; border: none; background-color: white; color: black; }

JavaScript:

javascriptCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); function updateClock() { const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } setInterval(updateClock, 1000); resizeButton.addEventListener('click', function() { clock.classList.toggle('large'); });

Kasutatakse kahte eventListenerit - **setInterval** funktsiooni, mis uuendab kella iga sekundi järel, ja nupuvajutuse eventListenerit, mis käivitab funktsiooni **toggle**, mis lisab või eemaldab klassi **large** kellale vastavalt kasutaja soovile. Klass **large** on määratud CSS-is ja see muudab kella suurust.

CSS-is kasutatakse protsendilisi väärtusi, et muuta kella vastavalt erinevatele ekra

**Kuidas saada tundide, sekundite ja minutite vahele koolon? function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours + ":"; document.querySelector(".minutes").innerHTML = ":" + currentMinutes; document.querySelector(".seconds").innerHTML = ":" + currentSeconds; } setInterval(updateClock, 1000);**

Saate lisada kooloni tundide ja minutite vahele, asendades järgmise rea:

javascriptCopy code

document.querySelector(".hours").innerHTML = currentHours + ":"; document.querySelector(".minutes").innerHTML = ":" + currentMinutes;

järgmise reaga:

javascriptCopy code

document.querySelector(".hours").innerHTML = currentHours + ":"; document.querySelector(".minutes").innerHTML = ":" + currentMinutes + ":";

See lisab kooloni minutite ja sekundite vahele. Lõplik kood koos kooloniga oleks järgmine:

javascriptCopy code

function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours + ":"; document.querySelector(".minutes").innerHTML = ":" + currentMinutes + ":"; document.querySelector(".seconds").innerHTML = ":" + currentSeconds; } setInterval(updateClock, 1000);

**Kuidas kuvada kuupäeva? function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours + ":"; document.querySelector(".minutes").innerHTML = ":" + currentMinutes + ":"; document.querySelector(".seconds").innerHTML = ":" + currentSeconds; } function upDateDate(){ d = new Date(); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } console.log(date, month, year); dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000);**

Teie praegune kood sisaldab juba funktsiooni **upDateDate**, mis võimaldab kuupäeva kuvada. Funktsioonis kasutatakse **new Date()** funktsiooni, et saada hetke kuupäev ja seejärel vormindatakse kuupäeva tervikuna stringiks, kasutades sobivaid kuupäevafunktsioone.

Siiski tuleb teil HTML-is luua kaks uut elemendikonteinerit, üks kuupäeva jaoks ja teine päeva jaoks, et kuvada kuupäev ja päev vastavalt teie soovile. Näiteks võiksite lisada järgmise HTML-koodi:

phpCopy code

<div class="date-container"></div> <div class="day-container"></div>

Siis muutke funktsiooni **upDateDate** nii, et see kasutaks **document.querySelector()** meetodit uute elemendikonteinerite leidmiseks, ja seadistaks vastava sisu teksti:

vbnetCopy code

function upDateDate(){ const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"]; const months = ["jaanuar", "veebruar", "märts", "aprill", "mai", "juuni", "juuli", "august", "september", "oktoober", "november", "detsember"]; const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; }

Lõpuks kutsuge funktsioon **upDateDate()** välja koos funktsiooniga **updateClock()** järgmise koodiga:

scssCopy code

updateClock() upDateDate(); setInterval(updateClock, 1000);

**Kuidas saab CSS koodiga HTML lehe taustavärvi muuta?**

HTML lehe taustavärvi muutmiseks võite kasutada CSS-i **background-color** omadust. See võimaldab teil määrata mis tahes värviga taustavärvi. Siin on mõned näited:

Ühtse värviga taust:

cssCopy code

body { background-color: red; }

Gradient taustavärv:

cssCopy code

body { background: linear-gradient(to bottom right, #ff0066, #6600ff); }

Pildi taustavärv:

cssCopy code

body { background-image: url("pilt.jpg"); }

Lisaks saate kasutada ka teisi CSS-i omadusi, nagu **background-repeat**, **background-size**, **background-position** jne, et täpsemalt kohandada taustavärvi kuvamist vastavalt oma vajadustele.

**Kas CSS-i "background-color " peab välja tooma ka HTML-is või JavaScriptis?**

CSS-i **background-color** omadust ei ole vaja HTML-is ega JavaScriptis välja tuua. Seda saab kasutada otse CSS-i stiililehe või stiilibloki sees, et määrata HTML elemendi taustavärv.

Siin on näide, kuidas CSS-i **background-color** omadust saab kasutada HTML elemendi taustavärvi määramiseks:

htmlCopy code

<!DOCTYPE html> <html> <head> <title>Minu HTML lehekülg</title> <style> body { background-color: #f2f2f2; } </style> </head> <body> <h1>Tere tulemast minu lehele</h1> <p>Siin on minu lehe sisu</p> </body> </html>

Sel juhul on HTML lehe taustavärv määratud CSS-i stiililehe sees, kasutades **body** selektorit ja **background-color** omadust. HTML-is pole selleks vaja midagi eraldi kirjutada.

**Kuidas muuta teksti värvi: .clock{ width: 100%; height: 100%; display: flex; justify-content: space-around; align-items: center; font-size: 5em; background-color: lightgray; text-decoration-color: crimson; }**

Teksti värvi muutmiseks saate kasutada CSS-i omadust **color**. See omadus määrab HTML elemendi teksti värvi. Näiteks, kui soovite muuta **.clock** elemendi teksti värvi punaseks, võite selle CSS-i reegli lisada:

cssCopy code

.clock{ width: 100%; height: 100%; display: flex; justify-content: space-around; align-items: center; font-size: 5em; background-color: lightgray; color: crimson; }

Siin on lisatud uus rida **color: crimson;**, mis määrab teksti värvi punaseks. Saate kasutada mistahes CSS-i toetatud värviväärtust, nagu näiteks nime, HEX-koodi või RGB-väärtust, et määrata teksti värv.

**Stylesheet CSS ei toimi. Kas kõik õige?**

**HTML:**

**<!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html>**

**CSS:**

**.clock{ width: 100%; height: 100%; display: flex; justify-content: space-around; align-items: center; font-size: 10em; background-color: lightgray; color: crimson; }**

**JavaScript:**

**function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000);**

CSS-stiilid tunduvad olevat õigesti rakendatud. Kuid, kuna kella ja kuupäeva kuvamise elemendid on määratletud klassidega **.clock**, **.date-container** ja **.day-container**, siis peaks CSS-i reeglid olema rakendatud nende klasside järgi, mitte üldisele **body** elemendile.

Seetõttu peaks CSS-i reeglid olema lisatud eraldi faili (nagu antud juhul on **kodukell.css**) või **<style>** elemendi sisse (aga mitte **body** elemendi sisse). Näiteks, kui soovite kella taustavärvi muuta, peaks CSS välja nägema järgmiselt:

cssCopy code

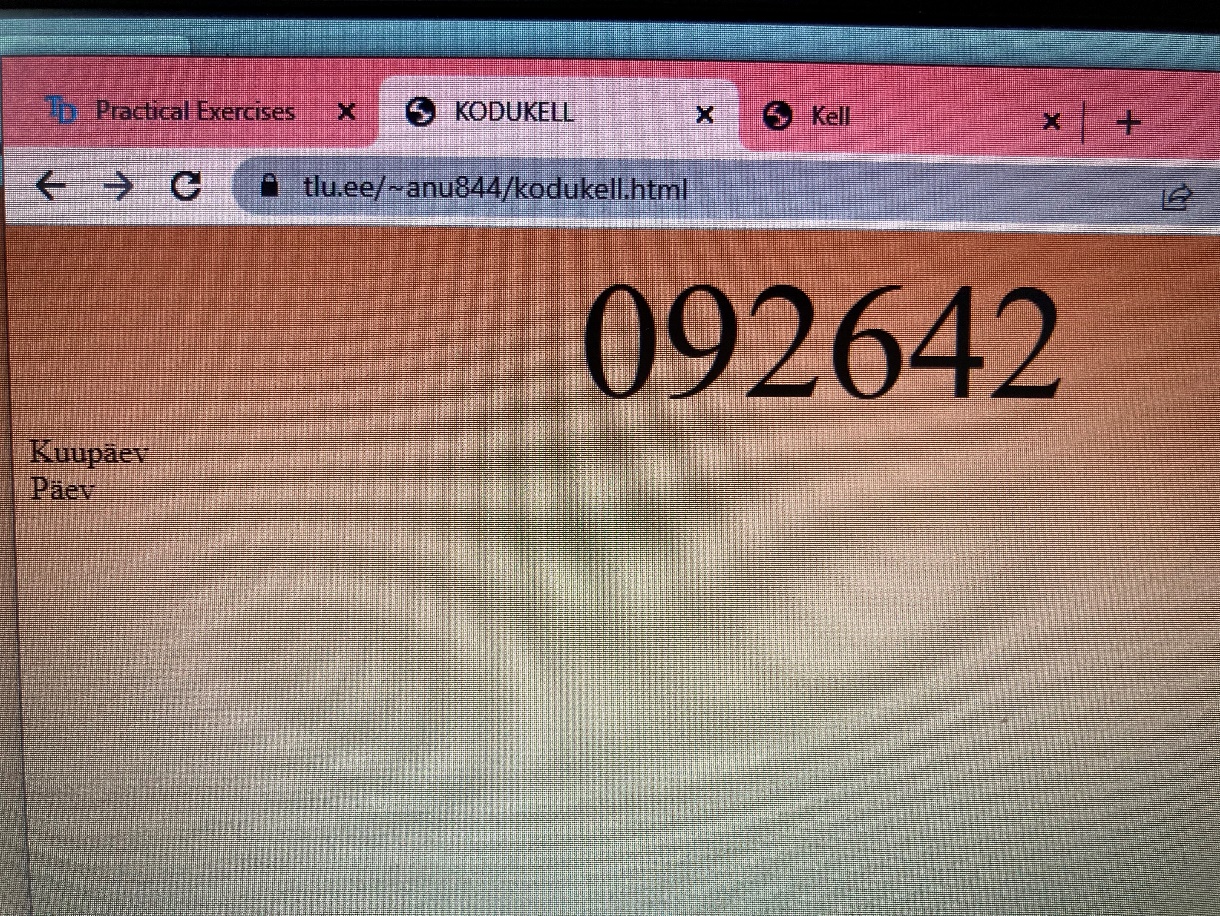
.clock { width: 100%; height: 100%; display: flex; justify-content: space-around; align-items: center; font-size: 10em; background-color: darksalmon; color: crimson; }

Sellisel juhul muudetakse kella taustavärvi "lightsalmon" asemel "darksalmon"-iks.

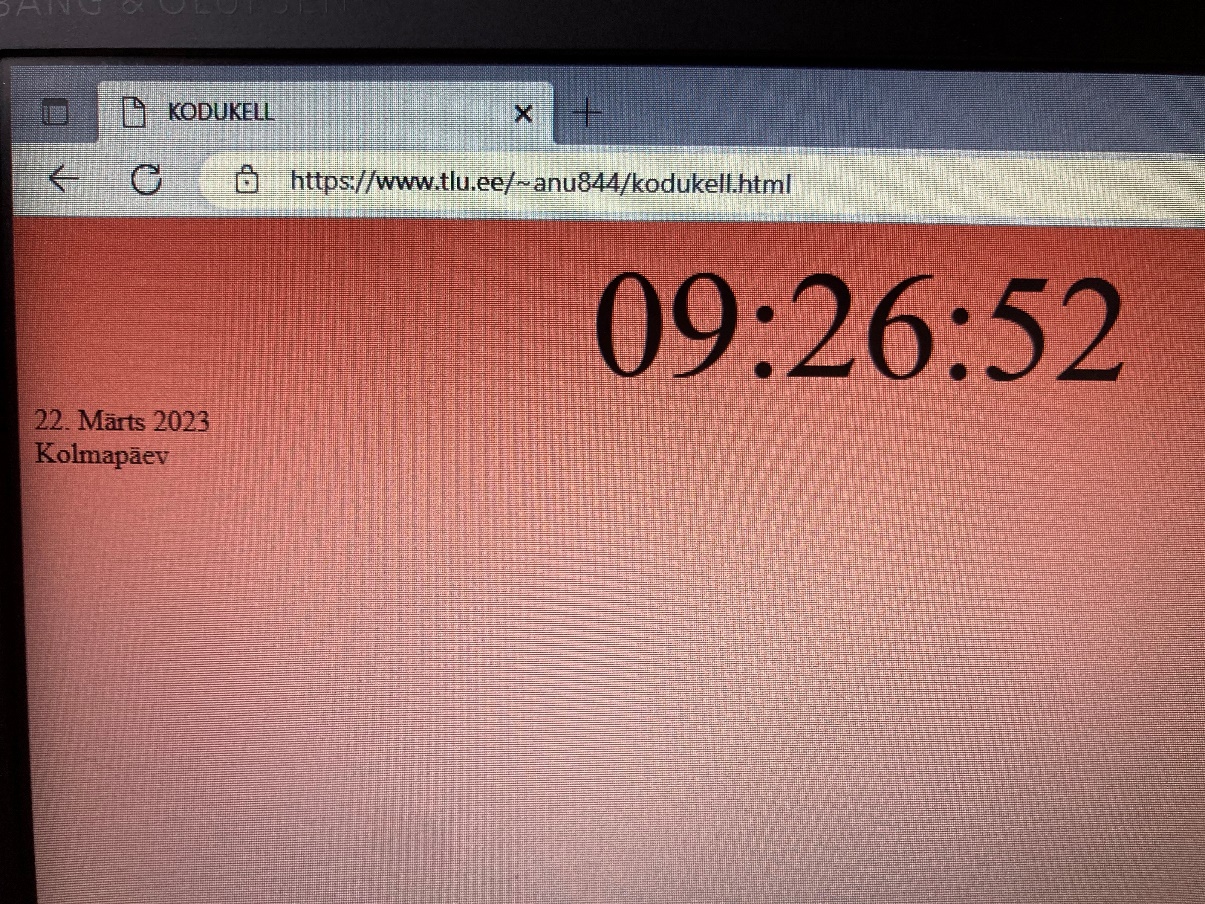
**SIIN SAIN ARU, ET MINU VEEBIBRAUSERID EI TEE KOOSTÖÖD EGA KUVA MULLE TEHTUD MUUDATUSI. EDASI VAATASIN OMA MUUDATUSI TELEFONIST, KUS KÕIK KOHE TOIMIS. NÜÜD AGA VÄIKE TAGASILÖÖK, KUIDAS LISADA EVENTLISTENERE, KUI AINULT TELEFONIS LEHT TÖÖTAB… TULEB NATUKE MÕELDA JA VÕIB-OLLA PAUS TEHA.**

LISAN KA PILDID HETKEL NÄHTAVATEST VARIANTIDEST:

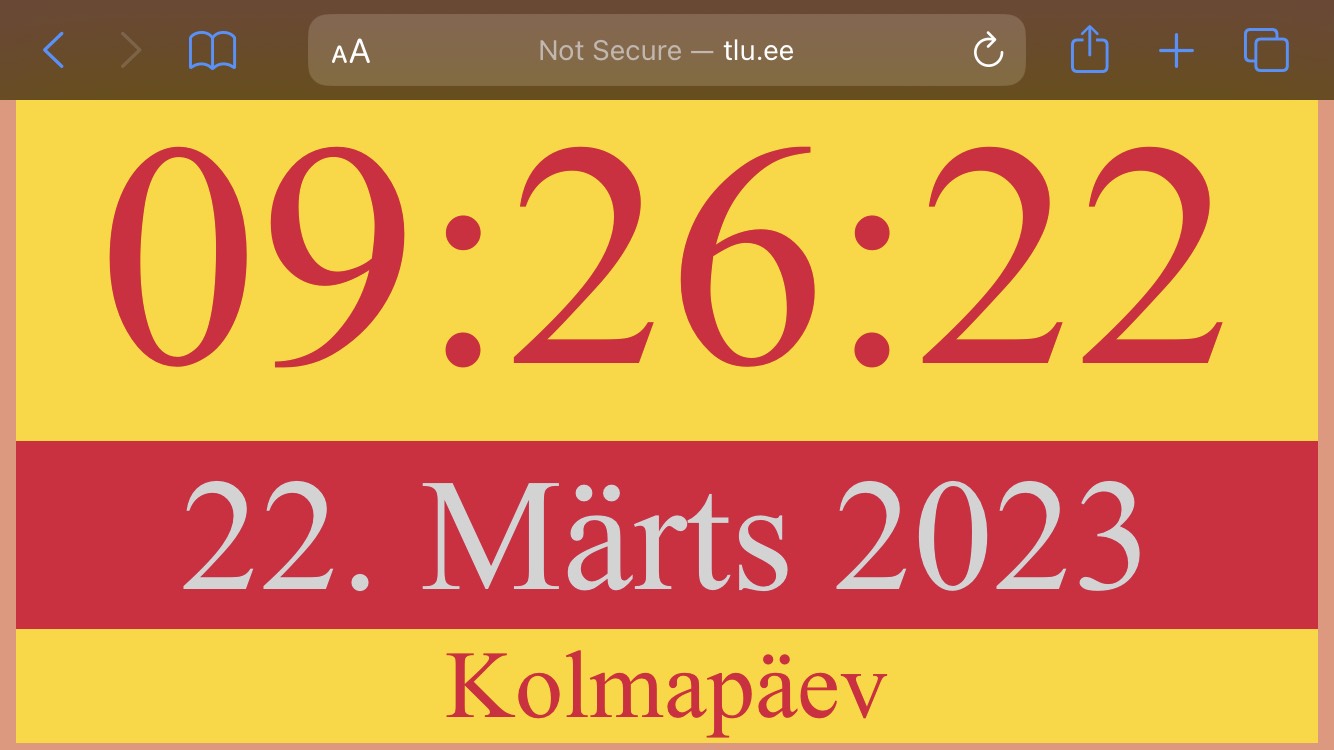
CHROME:



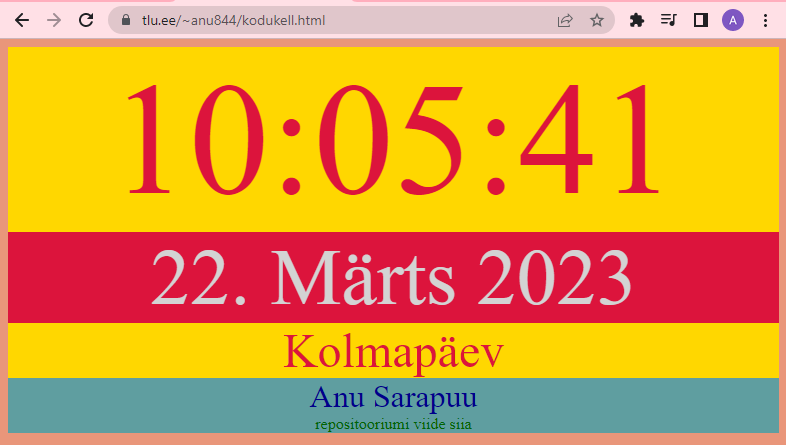
MICROSOFT EDGE:



IPHONE:



**UPDATE:** Chrome otsustas koostööd tegema hakata!



**EDASI UUED CHATGPT VESTLUSED:**

**"Button" ei muuda clock suurust veebilehel. Mis on valesti? HTML: <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Anu Sarapuu </div> <div class="repo"> repositooriumi viide siia </div> <button id="resize-button">Muuda kella suurust</button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html> CSS: .clock{ width: 100%; height: 100%; display: flex; justify-content: center; align-items: center; font-size: 1em; background-color: gold; color: crimson; } .date-container{ display: flex; justify-content: center; align-items: center; font-size: 5em; background-color: crimson; color: lightgrey; } .day-container{ display: flex; justify-content: center; align-items: center; font-size: 3em; background-color: gold; color: crimson; } .author{ display: flex; justify-content: center; align-items: center; font-size: 2em; background-color: cadetblue; color: darkblue; } .repo{ display: flex; justify-content: center; align-items: center; font-size: 1em; background-color: cadetblue; color: darkgreen; } button{ display: flex; justify-content: center; align-items: center; top: 20px; right: 20px; padding: 10px 20px; font-size: 1em; border: none; background-color: lightyellow; color: black; } JAVASCRIPT: const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('large'); });**

Based on the provided code, it seems that the issue is related to the CSS code for the clock element. You have defined the clock class with a height and width of 100%, which makes it take up the entire available space on the webpage. When you add the "large" class by clicking the "Muuda kella suurust" button, it does not change the size of the clock because the height and width are already set to 100%.

To fix this issue, you can add some new CSS rules to the ".large" class to change the size of the clock when the button is clicked. For example, you can change the width and height properties to a fixed size, like this:

cssCopy code

.large { width: 500px; height: 500px; }

Of course, you can adjust the size to whatever value you want. After making this change, the clock element should resize when the button is clicked.

**<button id="resize-button"> Muuda kella suurust väiksemaks/suuremaks </button> Kuidas saab "väiksemaks/suuremaks" veebilehel järgmisele reale?**

Selleks saab kasutada HTML-i **<br>** elemendi, mis lisab reavahetuse. Muutes nupu koodi järgnevalt:

bashCopy code

<button id="resize-button"> Muuda kella suurust<br>väiksemaks/suuremaks </button>

Sellisel juhul kuvatakse "väiksemaks/suuremaks" uuel real.

**Kuidas teha button, et clickiga 24h kell muuta 12h kella vastu? HTML: <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Autor: Anu Sarapuu </div> <div class="repo"> <a href="https://github.com/anu844/1-kodutoo.git"> REPOSITOORIUM</a> </div> <button id="resize-button"> Muuda kella suurust<br>(väiksemaks/suuremaks) </button> <button id="color-button"> Muuda kella värvi<br>(kollaseks) </button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html> CSS: .clock{ width: 100%; height: 100%; display: flex; justify-content: center; align-items: center; font-size: 10em; } .date-container{ display: flex; justify-content: center; align-items: center; font-size: 5em; background-color: crimson; color: lightgrey; } .day-container{ display: flex; justify-content: center; align-items: center; font-size: 3em; background-color: gold; color: crimson; } .author{ display: flex; justify-content: center; align-items: center; font-size: 2em; background-color: cadetblue; color: darkblue; } .repo{ display: flex; justify-content: center; align-items: center; font-size: 1em; background-color: lightblue; } button{ display: flex; justify-content: center; align-items: center; top: 20px; right: 20px; padding: 10px 20px; font-size: 1em; background-color: lightyellow; color: black; height: 60px; width: 280px; margin: 5px; font-size: 20px; border-radius: 30px; } .small{ font-size: 5em; } .yellow{ color: yellow; } Javascript: const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); });**

Saate lisada uue nupu, millel on click-kuulaja, mis vahetab kellaformaadi 24-tunniselt 12-tunniseks ja vastupidi. Selle saavutamiseks saate kasutada Date objekti ja selle getHours () meetodit, et saada praegune tund, seejärel kasutada if-else avaldusi, et kontrollida, kas tundide arv on suurem kui 12, ja vastavalt sellele määrake tundide arv ja päeva osa.

HTML-koodi saab muuta järgmiselt:

phpCopy code

<button id="format-button"> Muuda kella formaati<br>(24h / 12h) </button>

Ja JavaScript-koodi saab muuta järgmiselt:

javascriptCopy code

const formatButton = document.querySelector('#format-button'); formatButton.addEventListener('click', function(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".day-container").innerHTML = timeOfDay; });

Selle koodiga vahetate kellaformaadi nupuvajutusega. Kui tundide arv on suurem kui 12, lahutatakse sellest 12 ja määratakse päevaosa "PM". Kui tundide arv on 0, siis määratakse tundide arvuks 12. Kui tundide arv on väiksem kui 10, lisatakse selle ette null. Seejärel värskendatakse kuvatavat aega vastavalt uuele formaadile.

**Kuidas väljastada nupud kõrvuti? <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Autor: Anu Sarapuu </div> <div class="repo"> <a href="https://github.com/anu844/1-kodutoo.git"> REPOSITOORIUM</a> </div> <button id="resize-button"> Muuda kella suurust<br>(väiksemaks/suuremaks) </button> <button id="color-button"> Muuda kella värvi<br>(kollaseks) </button> <button id="format-button"> Muuda kella formaati<br>(24h / 12h) </button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html>**

Saate väljastada nupud kõrvuti, kasutades CSS-i "display: inline-block" omadust. Selleks tuleb mõlemale nupule lisada CSS-klass, mis annab "display: inline-block" stiili. Näiteks võib muuta "resize-button" ja "color-button" nupu CSS-i järgmiselt:

cssCopy code

#resize-button, #color-button { display: inline-block; margin-right: 10px; }

See annab mõlemale nupule "display: inline-block" stiili, mis asetab need kõrvuti, ja "margin-right" omadus lisab nuppude vahele 10 pikslit paremalt.

Kui soovite kolme nuppu kõrvuti asetada, siis saate sama klassi kohaldada kõikidele kolmele nupule. Näiteks:

cssCopy code

.button { display: inline-block; margin-right: 10px; }

Seejärel lisage igale nupule "button" klass. Näiteks:

phpCopy code

<button id="resize-button" class="button"> Muuda kella suurust<br>(väiksemaks/suuremaks) </button> <button id="color-button" class="button"> Muuda kella värvi<br>(kollaseks) </button> <button id="format-button" class="button"> Muuda kella formaati<br>(24h / 12h) </button>

Sellisel juhul paigutatakse kõik kolm nuppu kõrvuti ja neil on 10 piksliline vahe üksteisest paremal.

**formatButton kestab ainule ühe sekundi. Miks? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; //document.querySelector(".day-container").innerHTML = timeOfDay; }); updateClock() upDateDate(); setInterval(updateClock, 1000);**

The **formatButton** seems to be slow because it's creating a new **Date** object every time it's clicked. Creating a new **Date** object is an expensive operation, and doing it repeatedly can cause a noticeable delay.

One way to fix this is to create the **Date** object only once, outside of the event listener, and then use that object inside the event listener. Here's an example:

javascriptCopy code

const currentTime = new Date(); formatButton.addEventListener('click', function(){ let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; });

By creating the **Date** object outside of the event listener, the expensive operation is only done once, and the button should respond more quickly.

**"formatButton" rakendub kui vajutada, kuid sekundi muutumisel käivitub uuesti "upDateClock" funktsioon. Kuidas jääb "formatButton" püsima senikauaks kuni vajutatakse uuesti "formatButton" nuppu? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); const currentTime = new Date(); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ //const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; //document.querySelector(".day-container").innerHTML = timeOfDay; });**

1 / 2

Probleem on selles, et koodis on kogu aeg kasutusel sama **currentTime** muutuja, mis võrdub alguses loodud uue **Date** objektiga. See tähendab, et funktsiooni **updateClock()** kutsutakse iga sekundi järel uuesti, kuid **formatButton** funktsioonis kasutatakse alati sama algse kellaaja väärtust.

Üks võimalus selle parandamiseks oleks kasutada eraldi muutujaid, mis salvestavad erinevate kellaegadega objekte. Näiteks võiksime luua uue muutuja **formattedTime**, mida ajakohastatakse iga kord, kui nuppu "formatButton" vajutatakse. Seejärel võib funktsioonis **updateClock()** kontrollida, kas **formattedTime** on defineeritud, ja kasutada seda kellaaja vorminguna, kui see on nii.

Allpool on näide selle kohta, kuidas seda teha:

javascriptCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let currentTime = new Date(); let formattedTime; function updateClock(){ let currentHours, currentMinutes, currentSeconds; if (formattedTime) { currentHours = formattedTime.getHours(); currentMinutes = formattedTime.getMinutes(); currentSeconds = formattedTime.getSeconds(); } else { currentHours = currentTime.getHours(); currentMinutes = currentTime.getMinutes(); currentSeconds = currentTime.getSeconds(); } currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ formattedTime = new Date(); let currentHours = formattedTime.getHours(); let currentMinutes = formattedTime.getMinutes(); let currentSeconds = formattedTime.getSeconds(); let timeOfDay

**"formatButton" rakendub kui vajutada, kuid sekundi muutumisel käivitub uuesti "upDateClock" funktsioon. Kuidas jääb "formatButton" püsima senikauaks kuni vajutatakse uuesti "formatButton" nuppu? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); const currentTime = new Date(); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ //const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; //document.querySelector(".day-container").innerHTML = timeOfDay; });**

Probleem seisneb selles, et "currentTime" muutujat kasutatakse mõlemas funktsioonis, "updateClock" ja "formatButton", kuid see on deklareeritud funktsiooni "upDateClock" siseselt. Seetõttu ei ole "formatButton" funktsioonil juurdepääsu ajale, mis on vajalik aja vormindamiseks.

Üks võimalus selle probleemi lahendamiseks on deklareerida "currentTime" muutuja väljaspool funktsioone, nii et see oleks kättesaadav kõigile funktsioonidele. Näiteks võib seda teha järgmiselt:

javascriptCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let currentTime = new Date(); function updateClock(){ currentTime = new Date(); // uuendame globaalset currentTime muutujat let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds

**formatButton.addEventListener väljastatakse ainult sekundiks. Miks? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); const currentTime = new Date(); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; });**

The reason why the formatButton.addEventListener is only displayed for a second is that the button is a type="submit" button. When you click on it, the page is automatically refreshed, causing the clock to reset to its initial value. To prevent the page from refreshing, you can add the event.preventDefault() method to the function that is called when the button is clicked. This will prevent the default behavior of the button and allow you to display the formatted time for as long as you like.

Here's an example of how you can modify the code to prevent the page from refreshing:

javascriptCopy code

formatButton.addEventListener('click', function(event){ event.preventDefault(); const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; });

By adding the **event.preventDefault()** method, the default behavior of the button is prevented, and the clock will display the formatted time until it is updated again.

**Still it formatButton lasts only for second**

The issue may be that the **formatButton** is a submit button inside a form element. By default, when you click on a submit button, the form will be submitted and the page will refresh, causing any changes made by the JavaScript to be reset.

To prevent the form from being submitted and the page from refreshing, you can add an event listener to the form and prevent the default behavior using **event.preventDefault()**:

javascriptCopy code

const form = document.querySelector('form'); form.addEventListener('submit', function(event) { event.preventDefault(); });

This will prevent the form from being submitted when the button is clicked, allowing your JavaScript to execute as expected.

**When click formatButton, it lasts only for a second. How to activate formatButton as long till formatButton is again clicked? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); const currentTime = new Date(); const form = document.querySelector('form'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; });**

To keep the formatButton activated until it is clicked again, you can use a boolean variable to toggle the state of the button. Here's an updated version of the code:

jsCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); const currentTime = new Date(); const form = document.querySelector('form'); let isFormatButtonActive = false; // add a variable to keep track of the state of the button function updateClock() { const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate() { const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if (date < 10) { date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function () { clock.classList.toggle('small'); }); colorButton.addEventListener('click', function () { clock.classList.toggle('yellow'); }); formatButton.addEventListener('click', function (event) { event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if (currentHours > 12) { formattedHours = currentHours - 12; timeOfDay = "PM"; } if (formattedHours === 0) { formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay

**Kas kood õige? formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); /\*const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds();\*/ let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; }});**

The code you provided has an issue. The **currentTime** variable is defined after its first use inside the if condition. Therefore, you need to move the **const currentTime = new Date();** line above the if condition. Here is the corrected code:

javascriptCopy code

let isFormatButtonActive = false; formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } });

In this updated code, I also added a **isFormatButtonActive** variable to keep track of the button's state.

**Kuidas teha button, et väljastada ainult Anu Sarapuu? <div class="author"> Autor: Anu Sarapuu </div>**

Saate luua nupu ja funktsiooni, mis muudab elemendi teksti "Autor: Anu Sarapuu"-ks. Siin on üks võimalus, kuidas seda teha:

HTML:

phpCopy code

<button id="author-button">Näita ainult Anu Sarapuu</button> <div class="author"> Autor: Anu Sarapuu </div>

JavaScript:

javascriptCopy code

const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Autor: Anu Sarapuu'; });

Kui kasutaja klõpsab nuppu, siis **click** sündmuse käivitamisel muudetakse **.author** elemendi tekst sisuks "Autor: Anu Sarapuu". Kui soovite, et nupp muudaks teksti tagasi algseks, võite lisada teise sündmuse käitleja, mis muudab teksti uuesti "Autor: [algne autor]".

**how to make button for getting random background color by click. <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Autor: Anu Sarapuu </div> <div class="repo"> <a href="https://github.com/anu844/1-kodutoo.git"> REPOSITOORIUM</a> </div> <button id="resize-button">Muuda kella suurust<br>(väiksemaks/suuremaks)</button> <button id="color-button">Muuda kella värvi<br>(kollaseks)</button> <button id="format-button">Muuda kella formaati<br>üheks sekundiks(12h)</button> <button id="author-button">Eemalda nime eest "Autor:"</button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html>**

You can use JavaScript to generate a random background color and apply it to the body of the page when the button is clicked. Here's an example:

1. Add a new button element with an id of "color-button" to your HTML:

htmlCopy code

<button id="color-button">Muuda taustavärvi</button>

1. Add the following code to your JavaScript file:

jsCopy code

const colorButton = document.getElementById('color-button'); colorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; });

This code adds a click event listener to the colorButton element. When the button is clicked, it generates a random hexadecimal color code and sets the background color of the body element to that color. The **Math.random()** function returns a random number between 0 and 1, which is then multiplied by the maximum value of a 24-bit hexadecimal color code (16777215) to generate a random number between 0 and 16777215. The **toString(16)** method converts the number to a hexadecimal string, which is then concatenated with a "#" character to form a valid color code.

**Kuidas teha nupp, mis kuvaks kuupäeva numbritega? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } }); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Anu Sarapuu'; }); backColorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; });**

Selleks, et kuvada kuupäeva numbritega, tuleks koodi lisada uus funktsioon, mis kuvaks kuupäeva. Seejärel tuleks lisada setInterval meetod, et funktsioon töötaks automaatselt ja uuendaks kuupäeva. Siin on näide funktsioonist, mis kuvab kuupäeva numbritega:

javascriptCopy code

function updateDateNumeric(){ const d = new Date(); const dateContainer = document.querySelector(".date-container-numeric"); let date = d.getDate(); let month = d.getMonth()+1; let year = d.getFullYear(); if(date < 10){ date = "0" + date; } if(month < 10){ month = "0" + month; } dateContainer.innerHTML = date + "." + month + "." + year; } updateDateNumeric(); setInterval(updateDateNumeric, 86400000); // 24 hours

Siin kasutame uut HTML elemendi klassi **.date-container-numeric**, et kuvada kuupäev numbritega. Lisaks kasutame setInterval meetodit, et uuendada kuupäeva iga 24 tunni tagant (86400000 millisekundit).

Samuti peate kopeerima ka CSS-i osa, et kuvada uut HTML elementi **.date-container-numeric**. Selleks tuleks lisada järgmine kood:

cssCopy code

.date-container-numeric { font-size: 40px; font-weight: bold; margin-top: 20px; }

Nii näeks HTML osa välja, kui kasutate kahte erinevat HTML elementi **.date-container** ja **.date-container-numeric**:

htmlCopy code

<div class="clock"> <div class="hours"></div> <div class="minutes"></div> <div class="seconds"></div> <div class="date"> <div class="date-container"></div> <div class="date-container-numeric"></div> </div> </div>

Kui olete kõik muudatused teinud, siis peaks kuvama nii kellaaja kui ka kuupäeva numbritega.

**Kuidas teha dateButton, et klikkides kuvaks html kuupäeva numbrites? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); const dateButton = document.getElementById('#date-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } }); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Anu Sarapuu'; }); backColorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; }); dateButton.addEventListener('click', function(){ dateContainer.innerHTML = date + ". " + month + "." + year; });**

Tundub, et olete juba loonud funktsiooni "upDateDate()", mis saab kuupäeva jaoks vajalikud andmed ning uuendab need vastavatele HTML-elementidele. Seega saate lihtsalt lisada sündmuse kuulaja "dateButton" elemendile ja kutsuda funktsiooni "upDateDate()" selle sündmuse käivitamisel. Muutke ka "dateButton" elemendi ID väärtus "#" asemel lihtsalt "date-button".

Siin on värskendatud kood koos "dateButton" elemendi ja sündmuse kuulajaga:

javascriptCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); const dateButton = document.getElementById('date-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; //const timeOfDay = (currentHours < 12) ? "AM" : "PM"; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes();

**vastus jäi pooleli**

Vabandust, ma ei saanud aru, millist vastust te täpselt ootate. Kas saaksite palun täpsustada, millega vajate abi?

**dateButtoniga palun**

Saate teha dateButtoni, mis kuvab kuupäeva numbrites järgmiselt:

1. Esiteks, looge HTML-s uus nupuelement, millele antakse ID ja klassinimi:

htmlCopy code

<button id="date-button" class="date-button">Date</button>

1. Seejärel peate skriptifailis leidma nupuelemendi ja lisama sellele klõpsamise sündmuse:

javascriptCopy code

const dateButton = document.getElementById('date-button'); const dateContainer = document.querySelector('.date-container'); dateButton.addEventListener('click', function(){ const currentDate = new Date(); const year = currentDate.getFullYear(); const month = currentDate.getMonth() + 1; const date = currentDate.getDate(); dateContainer.textContent = year + "-" + month + "-" + date; });

1. Lisaks peate HTML-le lisama ka kuupäeva kuvamise elemendi, mille sisuks on kuupäeva konteineri klass:

htmlCopy code

<div class="date-container"></div>

Nüüd peaks kuupäev kuvatama HTML-is nupule klõpsates. Palun veenduge, et olete lisanud ka CSS-i kujundused vastavalt oma vajadustele.

**Datebutton ei tööta. Miks=? Kas kõik õige? const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); const dateButton = document.getElementById('date-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dateContainer = document.querySelector(".date-container"); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } }); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Anu Sarapuu'; }); backColorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; }); dateButton.addEventListener('click', function(){ const currentDate = new Date(); const year = currentDate.getFullYear(); const month = currentDate.getMonth() + 1; const date = currentDate.getDate(); dateContainer.innerHTML = date + ". " + month + "." + year; });**

Tundub, et Datebuttoniga seotud funktsioonis **upDateDate()** puudub definitsioon dateContainer muutujale, mida kasutatakse uue kuupäeva kuvamiseks. Seetõttu ei saa uut kuupäeva õigesti kuvada.

Lisage funktsiooni **upDateDate()** esimesse ritta **const dateContainer = document.querySelector(".date-container");**, et see definitsioon oleks funktsiooni ulatuses kättesaadav. See peaks lahendama probleemi.

**DateButton ei tööta. Miks? Kas kõik õige? HTML: <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Autor: Anu Sarapuu </div> <div class="repo"> <a href="https://github.com/anu844/1-kodutoo.git"> REPOSITOORIUM</a> </div> <button id="resize-button">Muuda kella suurust<br>(väiksemaks/suuremaks)</button> <button id="color-button">Muuda kella värvi<br>(mustaks)</button> <button id="format-button">Muuda kella formaati<br>üheks sekundiks(12h)</button> <button id="author-button">Eemalda nime eest "Autor:"</button> <button id="backColor-button">Muuda taustavärvi<br>(juhuslik värv)</button> <button id="date-button" class="date-button">Kuva kuupäev numbritega</button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html> Javascript: const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); const dateButton = document.getElementById('date-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const dateContainer = document.querySelector(".date-container"); const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } }); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Anu Sarapuu'; }); backColorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; }); dateButton.addEventListener('click', function(){ const currentDate = new Date(); const year = currentDate.getFullYear(); const month = currentDate.getMonth() + 1; const date = currentDate.getDate(); dateContainer.innerHTML = date + ". " + month + "." + year; }); CSS: .clock{ width: 100%; height: 100%; display: flex; justify-content: center; align-items: center; font-size: 10em; color: crimson; background-color: gold; } .date-container{ display: flex; justify-content: center; align-items: center; font-size: 5em; background-color: crimson; color: lightgrey; } .day-container{ display: flex; justify-content: center; align-items: center; font-size: 3em; background-color: gold; color: crimson; } .author{ display: flex; justify-content: center; align-items: center; font-size: 2em; background-color: cadetblue; color: darkblue; } .repo{ display: flex; justify-content: center; align-items: center; font-size: 1em; background-color: lightblue; } button{ display: inline-block; justify-content: center; align-items: center; top: 20px; right: 20px; padding: 10px 20px; font-size: 1em; background-color: rgb(252, 252, 156); color: black; height: 70px; width: 280px; margin: 5px; font-size: 20px; border-radius: 30px; } .small{ font-size: 5em; } .black{ color: black; }**

Probleem seisneb selles, et JavaScript-failis pole kirjutatud funktsiooni, mis võimaldaks kuupäeva numbritega kuvamist nupule klõpsates. Lisage järgmine funktsioon:

javascriptCopy code

dateButton.addEventListener('click', function(){ const dateContainer = document.querySelector(".date-container"); const d = new Date(); let date = d.getDate(); let month = d.getMonth() + 1; let year = d.getFullYear(); dateContainer.innerHTML = date + "." + month + "." + year; });

See funktsioon lisatakse HTML-failis nupu elemendile sündmusekuulaja, nii et kui nuppu klõpsatakse, käivitatakse funktsioon, mis kuvab kuupäeva numbritega. Samuti peate muutma HTML-i järgmiselt:

htmlCopy code

<button id="date-button" class="date-button">Kuva kuupäev numbritega</button>

Märkus. Pidage meeles, et kuupäeva kuu tagastab **getMonth** meetod arvuna 0–11, mitte 1–12. Seetõttu lisame 1 kuu väärtusele, et saada korrektne kuu number.

**Kuidas teha, et kui dateButtonid teist korda vajutatakse, kuvatakse kuupäev nii nagu enne dateButtoni vajutamist? HTML: <!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <link rel="stylesheet" href="kodukell.css"> <script src="kodukell.js" defer></script> <title>KODUKELL</title> </head> <body> <div class="clock"> <div class = "hours">Tunnid</div> <div class = "minutes">Minutid</div> <div class = "seconds">Sekundid</div> </div> <div class="date-container"> Kuupäev </div> <div class="day-container"> Päev </div> <div class="author"> Autor: Anu Sarapuu </div> <div class="repo"> <a href="https://github.com/anu844/1-kodutoo.git"> REPOSITOORIUM</a> </div> <button id="resize-button">Muuda kella suurust<br>(väiksemaks/suuremaks)</button> <button id="color-button">Muuda kella värvi<br>(mustaks)</button> <button id="format-button">Muuda kella formaati<br>üheks sekundiks(12h)</button> <button id="author-button">Eemalda nime eest "Autor:"</button> <button id="backColor-button">Muuda taustavärvi<br>(juhuslik värv)</button> <button id="date-button" class="date-button">Kuva kuupäev numbritega</button> <script> updateClock(); </script> <style> body{ background-color: darksalmon; } </style> </body> </html> Javascript: const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#format-button'); let isFormatButtonActive = false; const authorButton = document.querySelector('#author-button'); const authorDiv = document.querySelector('.author'); const backColorButton = document.getElementById('backColor-button'); const dateButton = document.getElementById('date-button'); function updateClock(){ const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); currentHours = (currentHours < 10 ? "0" : "") + currentHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds; document.querySelector(".hours").innerHTML = currentHours ; document.querySelector(".minutes").innerHTML = currentMinutes; document.querySelector(".seconds").innerHTML = currentSeconds; } function upDateDate(){ const dateContainer = document.querySelector(".date-container"); const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; dateContainer.innerHTML = date + ". " + months[month] + " " + year; } updateClock() upDateDate(); setInterval(updateClock, 1000); resizeButton.addEventListener('click', function(){ clock.classList.toggle('small'); }); colorButton.addEventListener('click', function(){ clock.classList.toggle('black'); }); formatButton.addEventListener('click', function(event){ event.preventDefault(); if (isFormatButtonActive) { // if button is already active, deactivate it const currentTime = new Date(); document.querySelector(".hours").innerHTML = currentTime.getHours() + ":"; document.querySelector(".minutes").innerHTML = currentTime.getMinutes() + ":"; document.querySelector(".seconds").innerHTML = currentTime.getSeconds(); isFormatButtonActive = false; } else { // if button is not active, activate it const currentTime = new Date(); let currentHours = currentTime.getHours(); let currentMinutes = currentTime.getMinutes(); let currentSeconds = currentTime.getSeconds(); let timeOfDay = "AM"; let formattedHours = currentHours; if(currentHours > 12){ formattedHours = currentHours - 12; timeOfDay = "PM"; } if(formattedHours === 0){ formattedHours = 12; } formattedHours = (formattedHours < 10 ? "0" : "") + formattedHours + ":"; currentMinutes = (currentMinutes < 10 ? "0" : "") + currentMinutes + ":"; currentSeconds = (currentSeconds < 10 ? "0" : "") + currentSeconds + " " + timeOfDay; document.querySelector(".hours").innerHTML = formattedHours; document.querySelector(".seconds").innerHTML = currentSeconds; document.querySelector(".minutes").innerHTML = currentMinutes; isFormatButtonActive = true; } }); authorButton.addEventListener('click', function() { authorDiv.textContent = 'Anu Sarapuu'; }); backColorButton.addEventListener('click', function() { const randomColor = Math.floor(Math.random()\*16777215).toString(16); document.body.style.backgroundColor = "#" + randomColor; }); dateButton.addEventListener('click', function(){ const dateContainer = document.querySelector(".date-container"); const d = new Date(); let date = d.getDate(); let month = d.getMonth() + 1; let year = d.getFullYear(); dateContainer.innerHTML = date + "." + month + "." + year; }); CSS: .clock{ width: 100%; height: 100%; display: flex; justify-content: center; align-items: center; font-size: 10em; color: crimson; background-color: gold; } .date-container{ display: flex; justify-content: center; align-items: center; font-size: 5em; background-color: crimson; color: lightgrey; } .day-container{ display: flex; justify-content: center; align-items: center; font-size: 3em; background-color: gold; color: crimson; } .author{ display: flex; justify-content: center; align-items: center; font-size: 2em; background-color: cadetblue; color: darkblue; } .repo{ display: flex; justify-content: center; align-items: center; font-size: 1em; background-color: lightblue; } button{ display: inline-block; justify-content: center; align-items: center; top: 20px; right: 20px; padding: 10px 20px; font-size: 1em; background-color: rgb(252, 252, 156); color: black; height: 70px; width: 280px; margin: 5px; font-size: 20px; border-radius: 30px; } .small{ font-size: 5em; } .black{ color: black; }**

Selleks, et teha nii, et kui dateButtonid teist korda vajutatakse, kuvatakse kuupäev nii nagu enne dateButtoni vajutamist, saab teha järgmise muudatuse JavaScripti koodi:

1. Muuda funktsiooni **upDateDate()** nii, et see salvestaks kuupäeva vormingu esialgse oleku. See tähendab, et kuupäeva kuvamine numbritega oleks vaikimisi olek ja kuupäeva kuvamine sõnadega oleks teine olek.

javascriptCopy code

let isDateWordFormat = false; // lisatud uus muutuja, mis näitab, kas kuupäev kuvatakse numbritega või sõnadega let defaultDateFormat = ""; // lisatud uus muutuja, kus salvestatakse kuupäeva algne vorming function upDateDate(){ const dateContainer = document.querySelector(".date-container"); const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; if (defaultDateFormat === "") { // kui algne vorming pole salvestatud, salvesta see defaultDateFormat = date + ". " + months[month] + " " + year; } dateContainer.innerHTML = isDateWordFormat ? date + ". " + months[month] + " " + year : defaultDateFormat; // kontrollitakse, kas kuupäev kuvatakse numbritega või sõnadega ja vastavalt sellele kuvatakse kuupäev }

1. Muuda dateButtoni kuulaja nii, et see vahetaks kuupäeva vormingu esialgse ja teise oleku vahel. Selleks lisatakse funktsioonile **dateButton.addEventListener()** tingimus, mis kontrollib, kas kuupäev on juba sõnades, ja vastavalt sellele muudetakse **isDateWordFormat** väärtust. Samuti kutsutakse üles **upDateDate()** funktsioon, et kuvada õige kuupäeva vorming.

javascriptCopy code

dateButton.addEventListener('click', function(){ isDateWordFormat = !isDateWordFormat; // muudetakse kuupäeva vormingu olekut upDateDate(); // kuvatakse õige kuupäeva vorming });

Kokkuvõttes näeb kood välja selline:

javascriptCopy code

const clock = document.querySelector('.clock'); const resizeButton = document.querySelector('#resize-button'); const colorButton = document.querySelector('#color-button'); const formatButton = document.querySelector('#

**"Kokkuvõttes näeb kood välja selline: " jäi kood poolikuks**

Vabandust, kuid ilma täpsema kontekstita ei ole mul võimalik aidata teid selle koodiga. Kas saaksite anda rohkem teavet selle kohta, millele see kood peaks viitama või millises programmeerimiskeeles see kirjutatud on? Siis saaksin teid paremini juhendada.

**Milline näeb kood kokkuvõttes välja?**

Vabandust, kuid mul pole teavet ega konteksti, mille kohta te küsite. Palun andke mulle rohkem üksikasju, mida soovite teada koodi kohta.

**Selleks, et teha nii, et kui dateButtonid teist korda vajutatakse, kuvatakse kuupäev nii nagu enne dateButtoni vajutamist, saab teha järgmise muudatuse JavaScripti koodi: Muuda funktsiooni upDateDate() nii, et see salvestaks kuupäeva vormingu esialgse oleku. See tähendab, et kuupäeva kuvamine numbritega oleks vaikimisi olek ja kuupäeva kuvamine sõnadega oleks teine olek. javascript Copy code let isDateWordFormat = false; // lisatud uus muutuja, mis näitab, kas kuupäev kuvatakse numbritega või sõnadega let defaultDateFormat = ""; // lisatud uus muutuja, kus salvestatakse kuupäeva algne vorming function upDateDate(){ const dateContainer = document.querySelector(".date-container"); const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; if (defaultDateFormat === "") { // kui algne vorming pole salvestatud, salvesta see defaultDateFormat = date + ". " + months[month] + " " + year; } dateContainer.innerHTML = isDateWordFormat ? date + ". " + months[month] + " " + year : defaultDateFormat; // kontrollitakse, kas kuupäev kuvatakse numbritega või sõnadega ja vastavalt sellele kuvatakse kuupäev } Muuda dateButtoni kuulaja nii, et see vahetaks kuupäeva vormingu esialgse ja teise oleku vahel. Selleks lisatakse funktsioonile dateButton.addEventListener() tingimus, mis kontrollib, kas kuupäev on juba sõnades, ja vastavalt sellele muudetakse isDateWordFormat väärtust. Samuti kutsutakse üles upDateDate() funktsioon, et kuvada õige kuupäeva vorming. javascript Copy code dateButton.addEventListener('click', function(){ isDateWordFormat = !isDateWordFormat; // muudetakse kuupäeva vormingu olekut upDateDate(); // kuvatakse õige kuupäeva vorming }); Kokkuvõttes näeb kood välja selline: Milline näeb kokkuvõttes kood välja?**

Siin on lõplik JavaScripti kood koos muudatustega, et saavutada soovitud käitumine:

javascriptCopy code

let isDateWordFormat = false; let defaultDateFormat = ""; function upDateDate(){ const dateContainer = document.querySelector(".date-container"); const months = ["Jaanuar", "Veebruar", "Märts", "Aprill", "Mai", "Juuni", "Juuli"] const days = ["Pühapäev", "Esmaspäev", "Teisipäev", "Kolmapäev", "Neljapäev", "Reede", "Laupäev"] const d = new Date(); const dayContainer = document.querySelector(".day-container"); let date = d.getDate(); let month = d.getMonth(); let year = d.getFullYear(); let day = d.getDay(); if(date < 10){ date = "0" + date; } dayContainer.innerHTML = days[day]; if (defaultDateFormat === "") { defaultDateFormat = date + ". " + months[month] + " " + year; } dateContainer.innerHTML = isDateWordFormat ? date + ". " + months[month] + " " + year : defaultDateFormat; } const dateButton = document.querySelector(".date-button"); dateButton.addEventListener('click', function(){ isDateWordFormat = !isDateWordFormat; upDateDate(); });

See kood sisaldab kahte muudatust, et saavutada soovitud tulemus. Esimene muudatus on uue muutuja **defaultDateFormat** lisamine, mis salvestab kuupäeva algse vormingu. Teine muudatus on **dateButton** kuulaja muutmine, et see vahetaks kuupäeva vormingu esialgse ja teise oleku vahel ning kutsuks üles **upDateDate()** funktsiooni õige kuupäeva kuvamiseks.