

JavaScript – „przełącznik” trybu jasnego/ciemnego

Instrukcja krok po kroku – Adam Kwiatkowski

Na czym polega zadanie

Zadanie polega na przygotowaniu prostej strony internetowej, wykorzystującej JavaScript do zmiany motywu barwnego – przełączania się między trybem jasnym i ciemnym. Tryb jasny powinien mieć jasny kolor tła, a ciemne napisy. Tryb ciemny odwrotnie – ciemne tło, jasne napisy. Zadanie będzie też wykorzystywało reguły wyświetlania elastycznego „flex”, aby rozmieścić elementy. Kolejne kroki niezbędne do wykonania zadania:

- zbudować dokument HTML, w którym będzie „przycisk” przełączający oraz jakaś treść
- przygotować arkusz CSS, aby nadać stronie wygląd (w zewnętrznym pliku)
- napisać plik skryptowy, w którym będzie skrypt obsługujący zamianę trybu

1. Wskazówki co do Visual Studio Code

Przydatna wskazówka: Visual Studio Code pozwala na „wygenerowanie” początkowych linijek kodu związanych z dokumentem HTML – wystarczy utworzyć nowy plik i zapisać go jako plik .html, a następnie napisać w nim sam znak wykrzyknik (!) i nacisnąć Enter. Trzeba tylko pamiętać o zmianie zadeklarowanego języka i tytułu dokumentu, a także o podłączeniu arkusza stylów.

Druga przydatna wskazówka – w <head> dokumentu można napisać „link:css” i skorzystać z kolejnego skrótu, który generuje linijkę kodu odpowiedzialną za dodanie pliku CSS.

Trzecia przydatna wskazówka – możemy automatycznie utworzyć potrzebne pliki CSS i JS poprzez najpierw napisanie linijek, które mają podłączyć pliki o odpowiedniej nazwie (umieszczone w odpowiednich folderach), a następnie trzymając Ctrl kliknąć na nazwę pliku (którego jeszcze nie mamy) – Visual Studio Code poinformuje nas, że nie można otworzyć takiego pliku (bo go nie ma), ale zapyta, czy chcemy go utworzyć.

2. Dokument HTML

Dokument HTML ma składać się z dwóch bloków <div> umieszczonych w bloku <body>:

- pierwszy <div> będzie „pojemnikiem na przycisk” zmieniający styl strony. Powinien posiadać swoje ID (np. „buttonContainer”). Wewnątrz tego bloku należy umieścić drugi blok <div>, który będzie właściwym przyciskiem. Również powinien posiadać swoje ID (np. „switchButton”). W tym drugim bloku można umieścić jakąś treść (obrazek, ikonkę, tekst) który będzie wskazywać, w jakim trybie jesteśmy (np. jeżeli jesteśmy w trybie jasnym, to niech na przycisku wyświetla się ikonka „słońca”, a w trybie ciemnym „księżycyca”, albo można po prostu wpisać tam słowa „jasny”/”ciemny”). Do tego elementu należy także przypisać wyzwalacz „onclick”, który wykona funkcję zmieniającą tryb (np. onclick = „changeMode()”)
- kolejny blok <div>, (np. ID „contents”), w którym będzie dowolna zawartość tekstowa; można skorzystać ze skrótu „lorem” w Visual Studio, aby wygenerować taki losowy tekst.

3. Arkusz CSS

Stylistyka strony powinna wykorzystywać reguły wyświetlania elastycznego. Reguły jakie powinniśmy wykorzystać:

- body:
 - tło – ciemne
 - tekst – jasny
 - wyświetlanie (display) – elastyczne (flex)
 - kierunek układania się elementów elastycznych (flex-direction) – jeden pod drugim, w kolumnie
 - ułożenie elementów elastycznych w poziomie (justify-content) – wyśrodkowane
 - ułożenie elementów elastycznych w pionie (align-items) – wyśrodkowane
 - być może będzie trzeba „na sztywno” ustalić wysokość strony

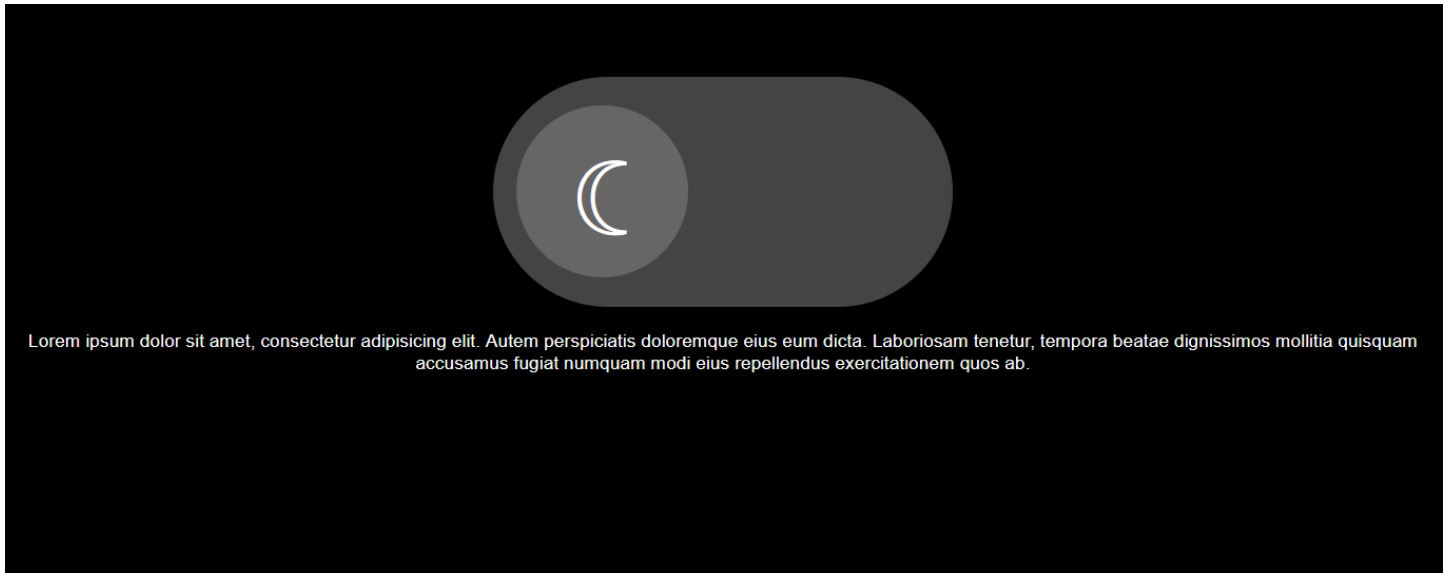
- pojemnik na przycisk:
 - tło – dowolne, ale inne od tła strony
 - wymiary – dowolne, np. 400 px szerokości, 200 px wysokości
 - wyświetlanie – elastyczne
 - kierunek układania się elementów elastycznych – jeden obok drugiego, w rzędzie
 - ułożenie elementów elastycznych w poziomie – na początku pojemnika (flex-start)
 - ułożenie elementów elastycznych w pionie – wyśrodkowane
 - warto też dodać np. zaokrąglone krawędzie (border-radius)
- przycisk:
 - tło – dowolne, ale inne od tła pojemnika na przycisk
 - wymiary – dowolne, ale takie, by mieściły się całkiem w pojemniku, np. 150 px szerokości i wysokości
 - wyświetlanie – elastyczne
 - ułożenie elementów elastycznych w poziomie - wyśrodkowane
 - ułożenie elementów elastycznych w pionie – wyśrodkowane
 - warto też dodać np. zaokrąglone krawędzie (border-radius)
 - jeżeli jest w środku tekst symbolizujący aktywny tryb, warto dostosować rozmiar czcionki (font-size)

4. Skrypt JS

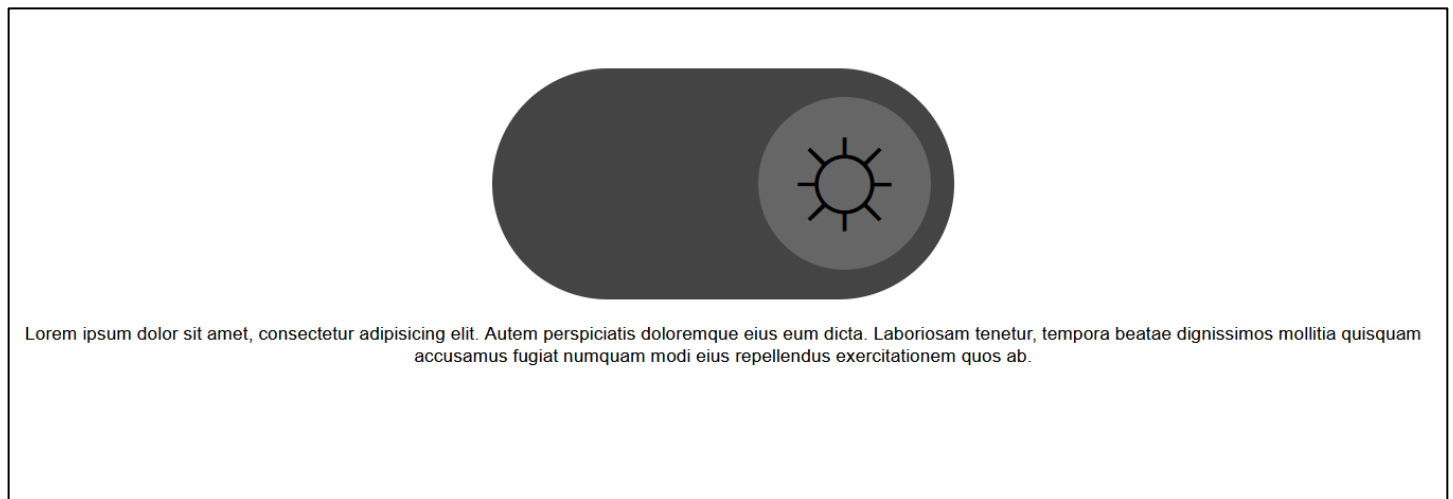
Skrypt JS ma składać się ze zmiennej globalnej (czyli umieszczonej poza funkcją), przechowującej obecnie aktywny tryb oraz funkcji zmieniającej tryb. Kolejne kroki, które ma wykonać skrypt:

- utworzenie zmiennej globalnej (np. let mode), której przypisana zostanie wartość obecnie aktywnego trybu (jeżeli zaczęto od trybu ciemnego, to wartość zmiennej mode można ustawić na „dark”)
- zbudowanie funkcji (np. changeMode), która będzie aktywowana z każdym naciśnięciem przycisku na stronie:
 - funkcja ma rozpocząć się od warunku (if), który sprawdzi, jaki jest obecnie aktywny tryb. Jeżeli obecnie aktywny jest tryb „dark” (if (mode == „dark”)), to ma wykonać się część kodu, która zamieni styl poszczególnych elementów na tryb „light”:
 - zamiana tła strony na jasny – można wykorzystać jedną z kilku metod, np.:
 - document.getElementsByTagName(„body”)[0].style.backgroundColor = „wybrany kolor” – ta metoda wyszukuje wszystkie elementy <body> i wybiera pierwszy z nich (ten o indeksie 0)
 - document.querySelector(„body”).style.backgroundColor = „wybrany kolor” – ta metoda wybiera pierwszy z brzegu element, który spełnia warunki zapytania w nawiasach
 - document.getElementById(„jakieś ID jakie nadano elementowi <body>”).style.backgroundColor = „wybrany kolor” – ta metoda wymaga nadania elementowi <body> jakiegoś ID, którym można się posłużyć do znalezienia elementu
 - zamiana koloru tekstu strony na ciemny – można wykorzystać powyższe metody, tylko zmieniać atrybut „color”, a nie „backgroundColor”
 - zamiana pozycji przycisku z początku na koniec pojemnika – przycisk ma zachować się jak przełącznik, ma przeskoczyć z miejsca na miejsce po zmianie stylu. Należy zmienić atrybut pojemnika na przycisk, który odpowiada za ułożenie elementów w nim się znajdujących w poziomie na „na końcu pojemnika” (np. document.getElementById(„buttonContainer”).style.justifyContent = „flex-end”).
 - jeżeli umieściliśmy jakiś tekst w środku przycisku, należy go zaktualizować (np. document.getElementById(„switchButton”).innerText = „jasny”)
 - zamiana wartości zmiennej globalnej mode na „light”, aby zasignalizować, że wykonaliśmy zmiany i obecnie aktywny tryb to tryb jasny
 - w przeciwnym wypadku, jeżeli obecnie aktywny jest tryb „light”, to ma wykonać się część kodu, która zamieni styl poszczególnych elementów znów na „dark” – należy wykonać wszystkie kroki wymienione wyżej, ale z odwrotnym rezultatem:
 - tło strony powinno znów być ciemne
 - tekst powinien znów być jasny
 - przycisk powinien wrócić na początek swojego pojemnika (flex-start)
 - tekst w przycisku powinien znów wskazywać aktywny aktualnie tryb („ciemny”)

- zmienna globalna przechowująca wartość aktywnego trybu powinna zostać zmieniona znów na „dark”



Rysunek 1. Strona w trybie "ciemnym", przed zamianą



Rysunek 2. Strona w trybie "jasnym", po zamianie