

## Zadanie – utworzenie szablonu kalkulatora

1. Celem ćwiczenia jest zaprojektowanie podstawowego wyglądu kalkulatora funkcyjnego działającego w przeglądarce WWW. Ćwiczenie do wykonania w parach (para = dwójkami, bez 3 i 4 osoby); dopuszczona jest praca w pojedynkę.

2. Technologie/języki opisowe/programowania niezbędne do wykonania zadania:

- HTML – hipertekstowy język znaczników
- CSS – kaskadowy arkusz stylów
- dowolna przeglądarka internetowa (polecana jest Mozilla Firefox lub Chromium/Opera)
- dowolny edytor plików tekstowych (polecany Notepad++, Kate bądź podobne)
- repozytorium git (niewymagane lecz nie wykorzystanie repozytorium powoduje obniżenie oceny o jeden stopień!)

3. Tworzony szablon jest jedną z części większego projektu mającego na celu stworzenie narzędzia do łatwego przeliczania różnych wartości pomiędzy najpopularniejszymi systemami liczbowymi. Finalny szablon tego ćwiczenia będzie podstawą kolejnych zadań, takich jak dodanie funkcjonalności przyciskom (JavaScript), prowadzenia historii operacji (obecnie puste miejsce w szablonie) czy też dodanie opcji ustawień.

4. Poniższe podpunkty najlepiej jest wykonywać sekwencyjnie (jeden po drugim). Oczywiście nie jest to obowiązkowe (można wykonywać wszystko wedle swojego rytmu pracy).

### **a) język HTML:**

- utworzenie podstawowego szkieletu HTML (HTML, HEAD, BODY, META, TITLE)
- utworzenie nadrzędnego elementu blokowego z id „okno”
- utworzenie wewnątrz #okno elementu blokowego z id „kalkulator”
- utworzenie wewnątrz #kalkulator elementu blokowego z id „wyswietlacz”
- utworzenie wewnątrz #wyswietlacz elementu paragrafu zawierającego dowolną cyfrę (np. 0); paragraf powinien pozwalać na edycję swojej zawartości (odpowiedni atrybut HTML)
- utworzenie wewnątrz #kalkulator elementu blokowego z id „przelicznik”
- utworzenie wewnątrz #przelicznik elementu blokowego z atrybutem title „Liczba szesnastkowa”; wewnątrz elementu powinny znaleźć się dwa elementy typu paragraf, z czego pierwszy paragraf powinien otrzymać zawartość „Hex”, drugi natomiast dowolną cyfrę (preferowana 0)
- powtórzenie poprzedniego kroku 3x, wpisując kolejno wartości Oct, Bin oraz Dec (odpowiednie należy także zmienić atrybut title na „Liczba ósemkowa”, „Liczba binarna”, „Liczba dziesiętna”
- utworzenie wewnątrz #kalkulator elementu blokowego z id „klawiatura”
- utworzenie wewnątrz #klawiatura elementu blokowego z klasą „przycisk”. Element powinien zawierać paragraf z wartościami klawiszy. Pierwszy klawisz powinien zawierać literę A
- powyższy krok należy powtórzyć 29x, wpisując dalej odpowiednio:  
litery od B do F (szesnastki),  
wartości OR, (, 7,4,1,+/-  
wartości XOR, ), 8, 5, 2, 0  
wartości NOT, C, 9, 6, 3, .  
wartości AND, /, \*, -, +, =

W razie wątpliwości proszę sugerować się zrzutami wyglądu kalkulatora (w dalszej części dokumentu)

- utworzenie wewnątrz #okno elementu blokowego z id „pamiec”
- wewnątrz #pamiec utworzenie listy nieuporządkowanej
- lista powinna zawierać 4 pozycje; każda z pozycji powinna zawierać następujące elementy:  
paragraf z dowolnym tekstem (np. „Zapisana wartość #”), element blokowy z tytułem „Dodaj

wartość do wyniku” oraz tekstem „M+”, element blokowy z tytułem „Odejmij wartość od wyniku” oraz tekstem „M-”, element blokowy z tytułem „Usuń wartość z pamięci” oraz tekstem „MC”

**b) język CSS (dodany w pliku html bądź w osobnym pliku dołączonym do HTML):**

- selektor **html** oraz **body** powinny posiadać marginesy (margin) oraz odbicia (padding) o wartości 0 (z dyrektywą wymuszenia tejże wartości)
- selektor **#okno**: szerokość 80%, wysokość 100vh, tło #071D70, lewy margines ustawiony automatycznie, prawy margines ustawiony na 0, przepełnienie obiektu ustawione na ukrycie (overflow), wyświetlanie (display) ustawione na flex
- selektor **#pamiec**: szerokość 30%, przepełnienie obiektu ustawione na ukrycie (overflow), przepełnienie obiektu w pionie ustawione na auto (overflow-y)
- element listy (**li**) w selektorze **#pamiec**: rozmiar fontu 20px, kolor tekstu #6E84D6 oraz pogrubienie fontu
- element blokowy znajdujący się w elemencie listy (**li**): wyświetlanie (display) ustawione na liniowo-blokowe (inline-block), wysokość na 25 pikseli, tło #1435AD, wyrównanie tekstu do środka, obramowanie (border) o szerokości 1 piksela, pełne (solid) z kolorem #6E84D6, lewy margines 5 pikseli, szerokość wyliczona na 33% z uwzględnieniem marginesów oraz szerokości obramowania, zmiana kursora na punktowy (pointer), zapobieżenie zaznaczania tekstu przez użytkownika (user-select wraz z kombinacjami/przedrostkami dla najpopularniejszych przeglądarek)
- selektor **#kalkulator**: margines lewy ustawiony na 0, margines prawy na 0, szerokość na 70%, wysokość 100%, przepełnienie obiektu ustawione na ukrycie (overflow)
- selektor **#przelicznik**: kolor tekstu #6E84D6, wysokość fontu 16 pikseli, font pogrubiony
- element blokowy wewnątrz **#przelicznik**: wysokość 30 pikseli, margines 0, przepełnienie obiektu ustawione na ukrycie (overflow)
- pierwszy element (**:first-child**) typu paragraf znajdującego się w elemencie blokowym wewnątrz **#przelicznik**: odbicie lewe (padding) na 10 pikseli, szerokość na 10%, zapobieżenie zaznaczania tekstu przez użytkownika (user-select wraz z kombinacjami/przedrostkami dla najpopularniejszych przeglądarek)
- paragraf znajdujący się w elemencie blokowym wewnątrz **#przelicznik**: margines 0, margines górny będący wyliczeniem połowy przestrzeni niezajmowanej przez tekst (wysokość elementu – wysokość tekstu przez połowę), margines prawy 20 pikseli
- selektor **#wyswietlacz**: tło #6E84D6, wysokość 100 pikseli, przepełnienie obiektu ustawione na ukrycie (overflow)
- paragraf wewnątrz **#wyswietlacz**: wysokość fontu 80 pikseli, kolor tekstu #071D70, margines 0, wyrównanie tekstu do prawej
- selektor **#klawiatura**: przepełnienie obiektu ustawione na ukrycie (overflow), wysokość wyliczona jako reszta po odjęciu wysokości wyświetlacza oraz wysokości przelicznika (należy pamiętać iż wysokość przelicznika to 4x wysokość każdego jego elementu), wyświetlenie flex, kierunek flex na kolumny (flex-direction), zawijanie flex (flex-wrap)
- selektor **.przycisk**: przepełnienie obiektu ustawione na ukrycie (overflow), szerokość wyliczona na 20% (odjęcie szerokości obramowania), wysokość wyliczona na 16,6% (6 przycisków w kolumnie, dodatkowo należy odjąć szerokość obramowania), obramowanie o szerokości 1 piksel lite o kolorze #6E84D6, tło o kolorze #1435AD, zmiana kursora na punktowy (pointer)
- paragraf wewnątrz **.przycisk**: kolor tekstu biały, wyrównanie tekstu do środka, pogrubienie fontu, font o wysokości 18 pikseli, margines ustawiony na 0, górny margines będący połową przestrzeni przycisku niezajmowanej przez tekst (**UWAGA!** Należy ręcznie obliczyć procentową wysokość przycisku – inaczej obliczenia będą błędne!), zapobieżenie zaznaczania tekstu przez użytkownika (user-select wraz z kombinacjami/przedrostkami dla najpopularniejszych przeglądarek)

Dla każdego elementu listy nieuporządkowanej, elementu blokowego w **#przelicznik**, każdego selektora **.przycisk** oraz paragrafu w elemencie **.przycisk** należy dodać pseudoklasę hover, która będzie zmieniać kolor tła na #4867D6, a kolor tekstu na #071D70.

Dodatkowo trzeba uwzględnić dostosowanie strony do rozmiaru okna przeglądarki (ang. responsive):

a) dla każdego ekranu gdy szerokość jest nie większa niż 780 pikseli:

- selektor **#okno** z szerokością 100%

b) dla każdego ekranu gdy szerokość jest nie większa niż 615 pikseli:

- selektor **#pamiec** z szerokością 0

- selektor **#kalkulator** z szerokością 100%

c) dla każdego ekranu gdy wysokość jest nie większa niż 460 pikseli:

- selektor **#przelicznik** z wysokością 0

- selektor **#klawiatura** z wysokością uwzględniającą wysokość ekranu (wysokość kalkulatora minus wysokość ekranu)

- paragraf wewnątrz .przycisk z marginesem górnym uwzględniającym eliminację #przelicznik (należy ponownie przeliczyć pozostałą przestrzeń)

d) dla każdego ekranu gdy wysokość jest nie większa niż 280 pikseli:

- selektor **#wyswietlacz** z wysokością 40 pikseli

- paragraf w **#wyswietlacz** z wysokością fontu 35 pikseli

- selektor **#klawiatura** z wysokością uwzględniającą zmianę wysokości ekranu (ponowne przeliczenie)

e) paragraf wewnątrz .przycisk z marginesem górnym uwzględniającym pomniejszenie **#wyswietlacz** (należy ponownie przeliczyć pozostałą przestrzeń)

Wygląd kalkulatora w przypadku pełnej rozdzielczości (1920x1080):

Hex	0
Oct	0
Bin	0
Dec	0

A	OR	XOR	NOT	AND
B	(	)	C	/
C	7	8	9	*
D	4	5	6	-
E	1	2	3	+
F	+/-	0	,	=

0

- Zapisana wartość #1  
M+ M- MC
- Zapisana wartość #2  
M+ M- MC
- Zapisana wartość #3  
M+ M- MC
- Zapisana wartość #4  
M+ M- MC

Maksymalna szerokość 780 pikseli:

0

Hex 0  
Oct 0  
Bin 0  
Dec 0

A	OR	XOR	NOT	AND
B	(	)	C	/
C	7	8	9	*
D	4	5	6	-
E	1	2	3	+
F	+/-	0	,	=

• Zapisana wartość #1

M+ M- MC

• Zapisana wartość #2

M+ M- MC

• Zapisana wartość #3

M+ M- MC

• Zapisana wartość #4

M+ M- MC

Maksymalna szerokość 615 pikseli:

<b>0</b>				
<b>Hex</b>	<b>0</b>			
<b>Oct</b>	<b>0</b>			
<b>Bin</b>	<b>0</b>			
<b>Dec</b>	<b>0</b>			
<b>A</b>	<b>OR</b>	<b>XOR</b>	<b>NOT</b>	<b>AND</b>
<b>B</b>	<b>(</b>	<b>)</b>	<b>C</b>	<b>/</b>
<b>C</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>*</b>
<b>D</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>-</b>
<b>E</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>+</b>
<b>F</b>	<b>+/-</b>	<b>0</b>	<b>,</b>	<b>=</b>

Maksymalna wysokość 460 pikseli:

0				
A	OR	XOR	NOT	AND
B	(	)	C	/
C	7	8	9	*
D	4	5	6	-
E	1	2	3	+
F	+/-	0	,	=

- Zapisana wartość #1  
M+ M- MC
- Zapisana wartość #2  
M+ M- MC
- Zapisana wartość #3  
M+ M- MC
- Zapisana wartość #4  
M+ M- MC

Maksymalna wysokość 280 pikseli:

0				
A	OR	XOR	NOT	AND
B	(	)	C	/
C	7	8	9	*
D	4	5	6	-
E	1	2	3	+
F	+/-	0	,	=

- Zapisana wartość #1  
M+ M- MC
- Zapisana wartość #2  
M+ M- MC
- Zapisana wartość #3  
M+ M- MC

## 5. Zasady oceniania zadania

Każda klasa zostanie poinformowana o zadaniu 10.04.2018 roku. Od tego momentu wszyscy mają czas na wykonanie zadania w ciągu 3 najbliższych zajęć lekcyjnych (3 godziny lekcyjne). Zadanie będzie przyjmowane do godziny 23.59 dnia 12.04.2018. Zadanie należy przesłać na adres nauczyciela (podany na zajęciach). Zadanie może być przesłane w postaci odnośnika do repozytorium bądź w pliku ZIP.

Jeżeli zadanie nie zostanie dostarczone na czas to drużyna otrzyma 2 (słownie: dwie) oceny niedostateczne (za część HTML oraz część CSS).

Oddane projekty będą oceniane w następujący sposób:

- poprawność kodu HTML nie obniża oceny
- drobne błędy HTML (np. nieodpowiednie domknięcie elementu) obniża ocenę o 0,5

- poważne błędy składniowe (np. niewłaściwe wstawienie dokumentu, domykanie elementów niekonsekwentnie, błędnie zapisane nazwy elementów czy atrybutów itp.) obniża ocenę o 1
- stosowanie składni przestarzałej (innej niż HTML5.x) automatycznie obniża ocenę o 2
- poprawność kodu CSS nie obniża oceny
- drobne błędy CSS (nadmiarowe nawiasy, brak średnika itp.) obniża ocenę o 1
- poważne błędy CSS (błędne nazwy właściwości, zakleszczanie właściwości itp.) obniża ocenę o 2

Dodatkowo nie zastosowanie w pracy repozytorium obniża ocenę o 1

**NIE SPAKOWANIE PRACY DO PLIKU ZIP/7Z DODATKOWO OBNIŻA OCENĘ O 1 (DOTYCZY PRZYPADKU NIE KORZYSTANIA Z REPOZYTORIUM).**

Oceną wyjściową (maksymalną) jest bardzo dobry, zaś ostateczną (minimalną) niedostateczny. Ocena celująca nie jest przewidziana (chyba, że drużyna podłączy do przycisków reakcje na zdarzenia w JavaScript).

W następnym tygodniu każda z drużyn będzie musiała dodatkowo obronić swój projekt. Każdy uczeń otrzyma maksymalnie 3 pytania, na które będzie musiał udzielić odpowiedzi. Nie udzielenie odpowiedzi na zadane pytanie powoduje obniżenie oceny o 1, odpowiedź niepełna obniża ocenę o 0,5. Udzielenie jednej niepełnej odpowiedzi gwarantuje ocenę dopuszczającą!